PREDICTION OF SPECTRUM OCCUPANCY USING MACHINE LEARNING ALGORITHMS: A CASE STUDY OF MINNA

BY

OYEWO, Temitayo Ayodeji

MENG/SEET/2018/7931

ELECTRICAL AND ELECTRONICS ENGINEERING DEPARTMENT,
SCHOOL OF ELECTRICAL ENGINEERING AND TECHNOLOGY,
FEDERAL UNIVERSITY OF TECHNOLOGY MINNA, NIGER STATE.

October, 2023

PREDICTION OF SPECTRUM OCCUPANCY USING MACHINE LEARNING ALGORITHMS: A CASE STUDY OF MINNA

BY

OYEWO, Temitayo Ayodeji

MENG/SEET/2018/7931

A THESIS SUBMITTED TO THE POSTGRADUATE SCHOOL FEDERAL UNIVERSITY OF TECHNOLOGY, MINNA, NIGERIA, IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF DEGREE OF MASTER OF ENGINEERING (M.ENG) IN ELECTRONICS ENGINEERING.

October, 2023

ABSTRACT

There is an alarming growth rate in spectrum usage, where some of the allocated spectra are fully engaged while others are sparsely utilized. The cognitive radio allows the primary users to use the available spectrum holes alongside the secondary users. The challenge of using cognitive radio technology is in the interference, which is a factor that causes a delay in the handoff time. This research developed a system that makes the cognitive radio operation more effective with little or no interference. Dataset were collected by scanning the spectrum between the frequency range of 80 MHz and 1 GHz using the Agilent N9342C Spectrum Analyzer (SA), which was connected to a personal computer and an antennae with a range of 47 MHZ to 1 GHz attached to the SA. The spectrum sensing exercise was carried out at Morris Fertilizer in Minna, Niger state, between 7:00 am-10:00 am (three hours). The method used in the sensing of the spectrum is Energy Detection. The dataset collected from the exercise was used to train and test different Machine Learning (ML) algorithms at a ratio of 7:3. The ML algorithms were used to predict the availability of the spectrum holes, that is, the frequency within the spectrum occupied or not occupied. The logistic Regression, Random Forest, Decision Tree, XGBoost and the K-Nearest Neighbour has training accuracy result of 94.84%, 99.93%, 99.93%, 99.86% and 98.19%, respectively and test accuracy result of 90.43%, 99.52%, 99.52%, 99.52%, and 97.61%, respectively. The test accuracy, precision, recall and F1-score are 90.43%, 90.40%, 93.39% and 91.43%, respectively was obtained with the application of logistic regression. Random forest results of accuracy, precision, recall and F1- score are 99.52%, 99.98%, 99.17% and 99.57%, respectively. For the Decision Tree, the test accuracy, precision, recall and f1-score are 99.52%, 99.99%, 99.17%, and 99.58%, respectively. The test accuracy, precision, recall and F1- score are 99.52%, 100.00%, 99.17% and 99.58%, respectively was obtained with the application of the XGBoost. Also, the test accuracy, precision, recall and f1-score are 97.61%, 100.00%, 95.87% and 97.89% respectively was obtained with the application of the KNN. From the result obtained, the XGBoost has the highest level of prediction accuracy. These results demonstrated the effectiveness of XGBoost when compared to other popular ML algorithms for spectrum occupancy prediction.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF PLATES

Plate **Page**

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
CR	Cognitive Radio
CRN	Cognitive Radio Network
CSV	Comma Separated Values
DSA	Dynamic Spectrum Access
DT	Decision Tree
ED	Energy Detection
FFT	Fast Fourier Transform
FPU	Floating Point Unit
GPS	Global Positioning System
GPU	Graphic Process Unit
IoT	Internet of Things
k-NN	k-Nearest Neighbours
ML	Machine Learning
NCC	Nigerian Communications Commission
PU	Primary User
QoS	Quality-of-Service
RF	Random Forest
SA	Spectrum Analyzer
SNR	Signal-To-Noise
SS	Spectrum Sensing
SVM	Support Vector Machine
SU	Secondary User
UHF	Ultra-High Frequency
VHF	Very High Frequency

WLAN	Wireless Local Area Networks
WSN	Wireless System Network
XGBoost	Extreme Gradient Boosting

CHAPTER ONE

1.0 INTRODUCTION

1.1 Background to the Study

The current exponential growth in technological advancements has led to an increased demand for wireless devices. This surge in demand, coupled with the static management of the radio spectrum, has resulted in a shortage of available spectrum. This shortage is due to the inefficiency of the static management of the spectrum, which is unable to accommodate the growing number of wireless devices.

Nasser *et al*. (2021) observed that most current wireless communication systems are based on the concept of fixed frequency allocation. This allocation has resulted in the overuse of certain portions of the radio spectrum and the underutilization of others, consequently leading to potential denial of service events. To address this scarcity of radio spectrum and to advance the evolution of devices, cognitive radio technology has been proposed as a solution. Therefore, further research is needed to assess the efficacy of this technology in mitigating the issue of radio spectrum scarcity.

Since its inception, there has been a significant amount of research conducted on Cognitive Radio Networks (CRN). Cognitive radio (CR) technology also helps to meet up with the quality-of-service (QoS) criteria of the radio spectrum while consuming less energy to carry out the task (Chen *et al*., 2018). Machine learning (ML) algorithms are used to mimic human intelligence and can make decisions without explicit programming (Goodfellow *et al.,* 2020). There are two categories of users in CRN namely principal or Primary Users (PU) and Secondary Users (SU) (Ding *et al*., 2018). Spectrum occupancy measurements are used by CR technology to comprehend how various spectrum bands are being used. These measurements can subsequently be used to create spectrum models that forecast upcoming usage trends (Bönsch and Kuhlen, 2020). Policymakers can use these models to help them make judgments about dynamic spectrum access (Wang and Liu, 2021).

CR technology adjusts its parameters to meet QoS requirements and conserve energy using ML algorithms. These methods select the spectrum occupancy measurements, which are subsequently applied to produce spectrum models (Al-Fuqaha *et al.*, 2015). These models guide decisions about dynamic spectrum access, which are crucial for the efficient utilization of the radio spectrum in CRNs. The use of ML algorithms in CR technology improves both the efficiency and effectiveness of spectrum utilization (Saber *et al*., 2020; Solanki *et al*., 2021).

To find patterns in data and base predictions or choices on those patterns, ML algorithms use statistical approaches. Examples of ML algorithms are Unsupervised, supervised, and reinforcement learning (Saber *et al.*, 2020). To evaluate and comprehend data without the use of labeled training examples, unsupervised learning methods are used. These algorithms are used to sort data into useful clusters or categories and to find patterns and relationships in data that might not be immediately obvious. Contrarily, supervised learning algorithms are trained on data that have labels, which means that the data contains both the input attributes and the intended output. This input-output mapping serves as the basis for the algorithm's predictions (Saber *et al.*, 2020; Solanki *et al*., 2021).

Examples of supervised learning include regression and classification tasks. Reinforcement learning algorithms are a type of ML algorithm that seeks to learn the best actions to take in a given environment to maximize a reward. These algorithms are used to train agents to make decisions in complex, dynamic environments. Thus, ML algorithms can be used to improve the performance of a cognitive radio network for spectrum sensing.

The whole process of spectrum sensing in collating data as a dataset and subject it to an ML algorithm to be able to build a model is carefully represented in Figure 1.1.

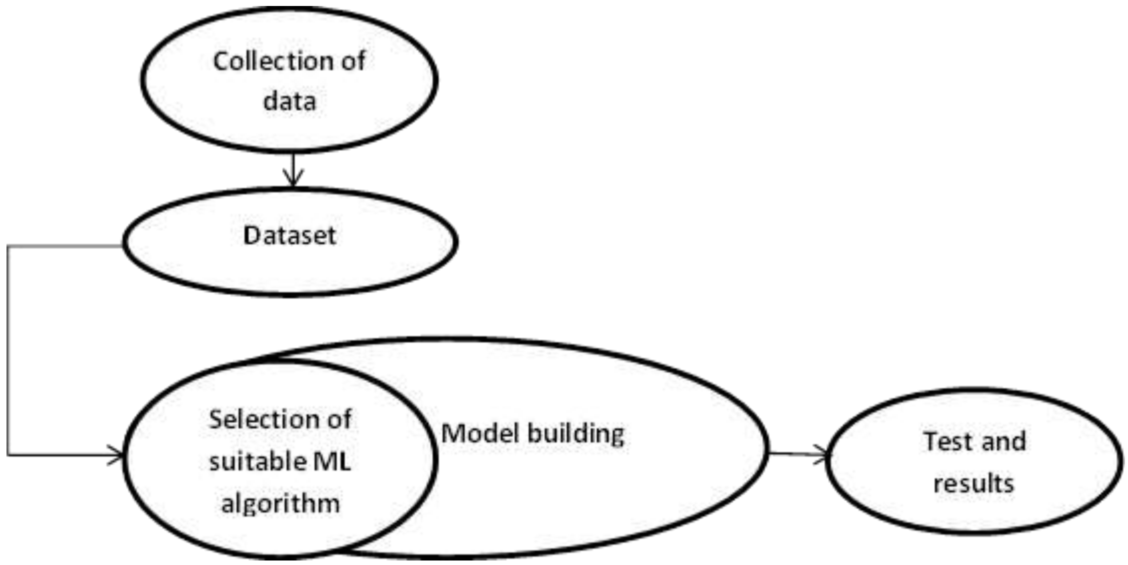

Figure 1.1: Workflow diagram of the entire process of prediction

1.2 Statement of the Research Problem

The Nigerian Communications Commission (NCC) has conducted relatively regular licensing rounds, showing that the allocated spectrum space is quickly diminishing, and the airwaves are becoming increasingly congested. The deployment of CR introduces the possibility of delays when transitioning between different frequency bands. This handoff delay can impact the overall performance of the CR system and affect the quality of service provided to PUs, and can lead to interference between the PU and the SU. Cognitive Radio technology is aware of its environment and can adjust its parameters to optimize the available spectrum (Wu *et al*., 2022). Cognitive radios enable increased spectral efficiency by sensing the environment and providing quality service to the PU, such as reducing interference levels and reducing handoff time. Additionally, CRs can use the discovered gaps in the unused licensed spectrum (white space or spectrum holes) for their transmissions, allowing secondary users to make use of the available spectrum.

Thus, this study determine the rate of spectrum utilization within the defined geographical location using Minna (Moris fertilizer) as a case study to sense and use the dataset to train different models of ML Algorithms to get the most efficient method to help improve the efficiency of the system while using the CR.

1.3 Aim and Objectives

The aim of the study is to predict spectrum occupancy using Machine Learning Algorithms. The objectives are to:

i. Scan the spectrum from the frequency range of 80 MHz to 1 GHz using a spectrum Analyzer.
ii. Train and test ML algorithms on the dataset so as to predict the availability of spectrum holes with various ML classifiers.
iii. Evaluate the performance of the different ML algorithms by using the model with the highest prediction accuracy.

1.4 Justification for the Research

Due to the increasing need for wireless standards and bandwidth-intensive technologies, a perceived shortage of spectrum has been observed. To meet the escalating demand for spectrum, a shift in spectrum management policy is required. Despite the existing regulatory obstacles that prevent these services from being easily accessible, acquiring spectrum licenses is typically exceedingly expensive for entrepreneurs.

CR systems have been proposed as a potential solution to the issue of spectrum occupancy, allowing multiple parties to share a single spectrum space, provided there are sufficient gaps in the allocated portions of the spectrum. Furthermore, the ability to predict occupancy levels and rates can assist entrepreneurs and network providers in determining whether the available parameters are suitable for their use.

Thus, sensing the spectrum within a geographical location to know the occupancy level and configure an ML algorithm that can predict when the frequency band is occupied or not occupied to help reduce the rate interference between the PUs and the SUs.

1.5 Scope of the Research

The scope of this research study is to investigate the rate of spectrum occupancy within the range of 80 MHz to 1 GHz and the utilization within a predetermined geographical area of Morris Fertilizer, within Minna as a case study. Data was collected over three hours to sense the level of occupancy in the early hours of the day (7:00am-10:00am).

The XGBoost ML algorithm, alongside some ML algorithms such as Random Forest, Logistic Regression, Decision Trees, and K-Nearest Neighbour were used to train and test the data aquired. This is to detect the spectrum occupancy level or the spectrum holes. This was achieved by using the dataset collected from the spectrum sensing using Agilent SA.

CHAPTER TWO

2.0 LITERATURE REVIEW

2.1 Review of fundamental concept and similar works

This chapter presents the background and an overview of the techniques and technologies relevant to the development of this thesis. The chapter links the problems introduced in Chapter one and the method used to solve the problem in the next chapter, and to provide a review of related works from literature.

2.2 Review of fundamental concept

Cognitive radio technology has the potential to address the shortage of available radio spectrum by enabling dynamic spectrum access. Since its introduction, researchers have been working on enabling this innovative technology in managing the radio spectrum. As a result, this research field has been progressing at a rapid pace and significant advances have been made (Arjoune and Kaabouch, 2019).

2.2.1 Cognitive Radio (CR)

Cognitive radio (CR) is a groundbreaking technology in the field of wireless communication that addresses the growing demand for efficient and flexible spectrum utilization (Usman *et al*., 2022). The primary objective of CR is to increase spectrum efficiency by enabling radio devices to intelligently and adaptively choose the best available frequency bands in the present. Dynamic spectrum access is made possible by CR, which, in contrast to conventional static spectrum allocation techniques, allows devices to detect unused or underutilized frequency bands and opportunistically transmit on them without interfering with other users. Fundamental to CR, this idea of spectrum agility is realized by sophisticated signal processing methods, machine learning algorithms, and intelligent decision-making capabilities built into CR devices. Cognitive radio's main objective is to utilize data more effectively. The main objective of cognitive radio is to utilize the limited and congested radio spectrum more effectively, resulting in enhanced connection for diverse wireless applications, improved wireless communication performance, and increased spectrum utilization (Zehra *et al.*, 2022).

Additionally, cognitive radio integrates self-learning processes to continuously update its understanding of the spectrum and adapt to shifting environmental conditions. It is not just about spectrum sensing and adaption. To ensure optimal spectrum allocation, CR devices may evaluate channel quality, find interference, and even bargain with other CR devices. In order to enable the coexistence of diverse wireless technologies, reduce interference, and promote effective spectrum sharing, all of which are crucial in the Internet of Things (IoT) age and its ever-expanding use, cognitive radio must be dynamic and adaptive (Salameh *et al.*, 2018).

2.2.2 Spectrum sensing

An essential component of cognitive radio systems is Spectrum Sensing (SS). This enables the identification of open frequency bands for opportunistic wireless communication. This procedure is essential for reducing the issue of spectrum scarcity and making sure that spectrum is used effectively. Recent rapid technology breakthroughs have made customers' lives easier through the development of sophisticated, cutting-edge devices. No matter the time or place, consumers can access data at fast speeds at their discretion. Due to the growing advantages of various wireless devices and technologies operating on well-known and effective radio frequency spectrum, radio frequency spectrum availability has become increasingly scarce in recent years. . Cognitive radio networks (CRNs) have been discovered to be efficient and intelligent solutions that offer an ideal means of allocating spectrum to demanding users through a series of intelligent sensing, aggregation of sensed information, and decision-making (Arjoune & Kaabouch, 2019; Sivagurunathan *et al.*, 2021).

Various spectrum sensing techniques have been developed to detect and identify unused or underutilized portions of the radio frequency spectrum. Recent research in this field, such as the work by has focused on machine learning algorithms and artificial intelligence to improve the accuracy and reliability of spectrum sensing (Saber *et al.*, 2020).

Spectrum sensing methods can be broadly categorized into two classes: energy detection and feature-based sensing. Energy detection involves measuring the energy level in a particular frequency band and comparing it to a predetermined threshold to determine the presence of a primary user. Feature-based sensing, on the other hand, leverages statistical and signal processing techniques to detect specific characteristics or features in the received signal (Arjoune and Kaabouch, 2019; Wu *et al*., 2022).

2.2.3 Machine Learning

Machine learning (ML) algorithms are built upon a foundation of core concepts that underpin their operation. Data serves as the raw material for these algorithms, consisting of input features and corresponding output labels, and it plays a central role in training models. Feature extraction and engineering are essential steps in preparing the data, involving the selection and transformation of relevant features. The machine learning model itself, whether it's a simple linear regression or a complex neural network, learns from this data by adjusting its parameters during the training process.

Machine learning is the scientific study of algorithms and statistical models that computer systems use to perform a specific task without being explicitly programmed. Learning algorithms in many applications that’s we make use of daily. Every time a web search engine like Google is used to search the internet, one of the reasons that work so well is because a learning algorithm that has learned how to rank web pages. These algorithms are used for various purposes like data mining, image processing, predictive analytics. The main advantage of using machine learning is that, once an algorithm learns what to do with data, it can do its work automatically. This adjustment is achieved through optimization algorithms, like gradient descent, which minimize the difference between the model's predictions and the actual target values (Mishra and Chaudhary, 2023).

The performance of machine learning models is rigorously evaluated through testing and validation on unseen data. Overfitting, where a model fits the training data too closely and generalizes poorly, and underfitting, where a model is too simplistic to capture data patterns, are common challenges in model training. Achieving a balance between bias and variance is crucial to ensure a model generalizes well to new data. Additionally, feature scaling, normalization, and proper handling of hyperparameters are important aspects of model development and tuning. Cross-validation techniques are employed to assess a model's performance and generalization on various data subsets, providing a comprehensive view of its capabilities (Batta, 2018; Wang and Liu, 2021).

To gauge a model's effectiveness, a range of evaluation metrics, specific to the problem at hand, is used. These metrics include accuracy, precision, recall, F1-score, and mean squared error, among others. They help quantify how well a machine learning algorithm performs against predefined objectives and provide valuable insights for model selection and refinement. These fundamental concepts form the bedrock of machine learning, guiding practitioners in the creation, training, and evaluation of models for diverse applications across various domains (Wang and Liu, 2021; Wu *et al*., 2022).

2.3 Review of Related Work on Cognitive Radio (CR)

The rapid increase in wireless devices and the limited availability of spectrum have raised concerns among researchers due to the growing demand for wireless connectivity. The significant growth in mobile subscribers and the need for multimedia access have led to an examination of the effectiveness of current radio frequency management. The traditional command and control approach to spectrum management has proven to be ineffective. Research on spectrum management has revealed that a large portion of the assigned radio frequency bands remains unused (Chukwuchekwa *et al*., 2021).

2.3.1 Cognitive radio

While some parts of the radio spectrum are often used, others are unused or only sometimes used. It was found that licensed users do not always make use of their spectrum resources. On the spectra of authorized users, there exist spectrum holes that can be accessed using dynamic spectrum access (DSA) technology (Bani, 2022).

Using Cognitive Radio technology, which has been thoroughly studied by the research community for more than two decades, is one way to overcome these and other difficulties. Wireless devices can sense the radio spectrum, make decisions about the condition of the frequency channels, and change their communication parameters to fulfill quality-of-service needs while consuming the least amount of energy possible through the use of cognitive radio technology (Arjoune and Kaabouch, 2019).

Using CR technology, which has undergone substantial research, is one way to overcome these and other difficulties. The capacity of cognitive radio technology to adjust to the radio environment is another crucial feature. Dynamic resource allocation techniques, which are created to optimize the utilization of available spectrum resources, are used to achieve this. These algorithms can also be used to modify the cognitive radio's broadcast settings, which will enhance its functionality and lessen the possibility of interference from other users (Arjoune and Kaabouch 2019; Bani 2022; Nasser *et al.*, 2021).

Cognitive Radio has been introduced as a potential candidate to perform complete Dynamic Spectrum Allocation (DSA) by exploiting the free frequency bands that are also called "spectrum holes" or "white spaces" (Hassan, *et al*., 2021).

The CR can be classified into two categories namely the Primary Users (PUs), and Unlicensed User which can also be referred to as Secondary Users (SUs). While PUs can access the spectrum whenever they want, SUs are restricted by the activities of PUs. In other words, SUs should respect the PUs' Quality of Service (QoS), and harmful interference coming from SUs to PUs transmission is prohibited.

In clear terms, CR helps the secondary user to use the spectrum without harming the (QoS) of the primary user. Interference with PUs is one of the biggest problems CR has to deal with. When the CR system tries to access a frequency band that is already in use by PUs, this may happen. Implementing interference-aware resource allocation algorithms that dynamically modify the CR system's transmission power to reduce interference with PUs is one suggested approach. Also, because the availability of frequency bands might change quickly, dynamic spectrum access presents another issue for CR. Implementing dynamic spectrum access algorithms that can quickly recognize and access accessible frequency bands is one suggested option. Users of CR systems must receive a specific level of QoS. Implementing QoS-aware resource allocation algorithms that may dynamically distribute resources to satisfy the QoS needs of various apps and users is one suggested solution (Zhang *et al*., 2023).

Numerous wireless communication systems, including cellular networks, wireless local area networks (WLANs), and wireless sensor networks (WSNs), have used cognitive radio technology. Cognitive radios can be used in cellular networks to increase the effectiveness of spectrum use by dynamically distributing resources to users following their needs. Also, by dynamically altering transmission parameters to prevent interference from other users and by optimizing the use of available spectrum resources, Cognitive radio has the potential to increase the effectiveness and performance of wireless communication networks.

2.3.2 Spectrum sensing

In cognitive radio, sensing techniques can be arranged into two fundamental classes: Narrowband and wideband. Energy detection (Nasser *et al*., 2021; Ranjan *et al*., 2016), cyclostationary detection (Arjoune and Kaabouch 2019), matched filter sensing, covariance-based detection, and machine learning-based sensing are examples of narrowband sensing approaches. Comprehensive wideband sensing and Nyquist-based wideband sensing are examples of wideband sensing techniques (Lu *et al*., 2017). The classification of these methods can be seen in Figure 2.1.

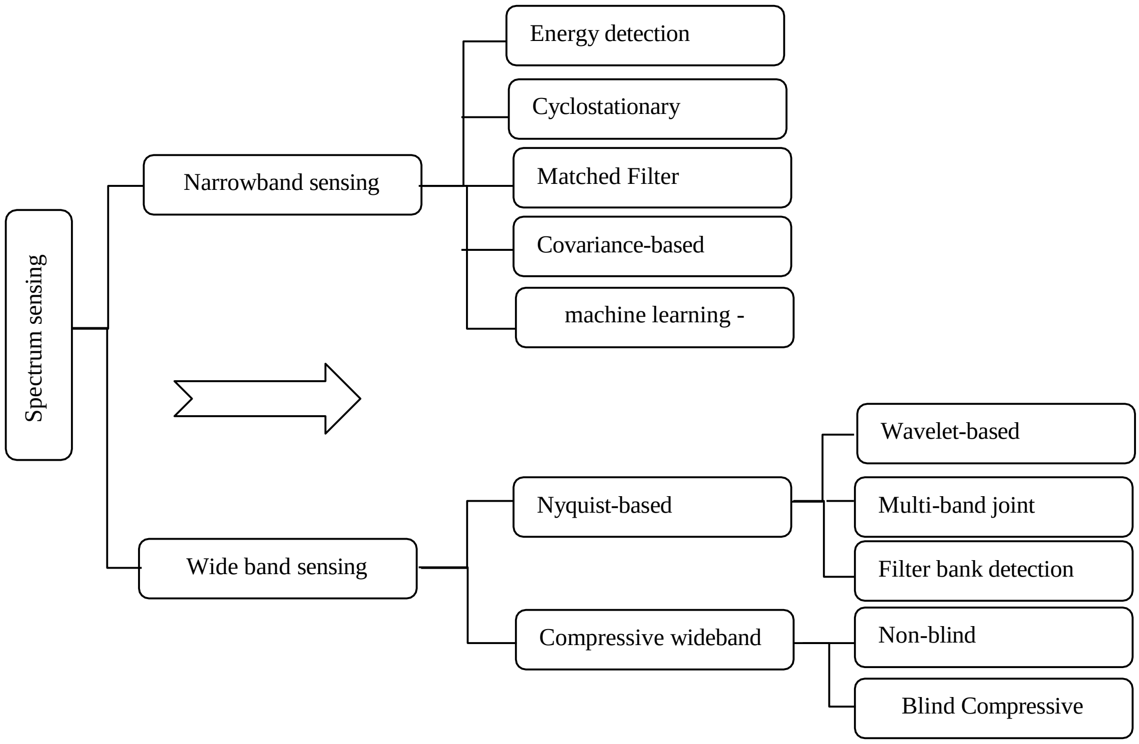

Figure 2.1: Classification of the spectrum sensing technique (Arjoune and Kaabouch, 2019)

2.3.2.1 The narrowband spectrum sensing technique

Narrowband spectrum sensing techniques is a type of technology that makes secondary users decide on the activity of the PU over a frequency channel of interest. Whether presently engaging the channel or not. Putting this in perspective, let us assume that H_0 denotes that the primary user signal is not present on the channel and H_1 denotes that the primary user signal is present. This simple illustration of the received signal under these two assumptions, H_0 and H_1, can be expressed as (Arjoune and Kaabouch, 2019):

$$H_0 : y(n) = \eta(n) \tag{2.1}$$

and:

$$H_1 : y(n) = s(n) + \eta(n) \tag{2.2}$$

where y(n) represents the received signal, η(n) represents a Gaussian white noise, s(n) is the transmitted signal, and n denotes the sensing time (Arjoune and Kaabouch, 2019).

State H_0 represents the primary user absence and state H_1 represents the primary user presence. For the sensing decision, a few of the recently referenced range-detecting methods can be utilized, including energy detection (Arjoune and Kaabouch, 2019; Arshid *et al*., 2022; Onumanyi *et al*., 2013), Cyclostationary detection, matched filter detection, covariance-based detection, and machine-learning-based detection which are examined underneath (Arjoune and Kaabouch, 2019; Nasser *et al*., 2021).

2.3.2.2 Energy detection

Energy detection (ED) computes the energy of the samples and compares it to a threshold (Arjoune and Kaabouch, 2019; Arshid *et al*., 2022). When the spectrum is observed, and the energy sampled seem higher than the threshold, this means the primary user signal is assumed present and if the signal is not above the threshold the primary user is considered absent. The concept calculates the energy of the samples as the squared magnitude of the Fast Fourier Transform (FFT) averaged over the number of samples N. This is given by (Arjoune and Kaabouch, 2019) :

$$T_{ED} = 1/N \sum_{N=1}^{N} \left(\mathrm{Y}[\mathrm{n}]\right)^2 \tag{2.3}$$

where N denotes the total number of received samples, and Y[n] denotes the nth received sample.

If that energy is above the threshold, the primary user is considered present; otherwise, the primary user is considered absent. This is expressed mathematically (Arjoune and Kaabouch, 2019):

$T_{ED} < \lambda_{ED}$ Primary User absent, (2.4)

and:

$T_{ED} < \lambda_{ED}$ Primary User present (2.5)

where λ_{ED} denotes the threshold that depends on the noise variance. The selection of the threshold, which can be static or dynamic, dramatically affects the detection performance. ED is a reasonably simple technique that does not require any prior knowledge of the signal characteristics. it has a low detection performance for low signal-to-noise (SNR) values.

In Table 2.1, the existing Spectrum Sensing Techniques for the Handoff Delay, Energy Efficiency, and Throughput and the comparison of the narrowband spectrum sensing methods are shown respectively (Arshid *et al*., 2022).

Table 2.1: Comparison of some of the existing spectrum sensing techniques for the handoff delay, energy efficiency, and throughput

Spectrum Sensing Technique	**Methods Used for Sensing**	**Throughput**	**Energy Efficiency**	**Handoff Delay**	**Merit**	**Limitations**
Cooperative Spectrum Sensing Technique	Cooperation between Multiple SPUs	Average	Average	Maximum	Reduction in Threshold. Sensitivity and Requirements.	Sometime wide channels need to be scanned. Increased Data Overhead.
Energy Detection	Sensed energy	Average	Average	Average	Easy to Implement. Do not Require Previous Information of FPU's.	High Sensing Times. Uncertainty of Noise Power. Need Tight Synchronization.
Matched filter detetion	Previous information of FPU	Average	Minimum	Maximum	Less detection time. Noise detection is optimal	Requires FPU's Previous Information. Need a Dedicated Receiver.
Cyclo-stationary Feature detection	Periodicity of received signals	Average	Minimum	Average	Robust to Noise. Improves SPU Throughput.	Long Sensing Time. High Computation Complexity.
Proposed Energy Efficiency in CRN	Energy detection on target channel	Improved	Improved	Minimized	Energy Efficiency Easy to Implement. Fewer Sensing Time. FPU's Previous Information not Required	

In Table 2.2, some sensing techniques are compared alongside energy detection showing the advantages and the disadvantages

Table 2.2: Advantages and disadvantages of the energy detection method and other three narrowband spectrum sensing methods

Sensing Technique	Advantages	Disadvantages
Energy detection (Alom *et al*., 2017; Arjoune *et al*., 2018.)	- Easy to implement - No prior knowledge of the primary signal characteristics is required	- High false alarm rate - Unreliable at low SNR values - Sensitive to noise uncertainty
Cyclo-stationary feature detection (Cohen and Eldar, 2017)	- Robust against noise uncertainty - Distinguish between signal and noise - The decreased probability of false alarms at low SNR	- Large sensing time to achieve a good performance - High energy consumption when the size of the samples is large
Matched Filter based detection (Saalahdine *et al*., 2016; Xinzhi *et al*., 2014)	- Better detection at low SNR region - Optimal sensing	- Prior knowledge of the primary user signal is required - Impractical since prior knowledge about the signal is not always available
Covariance-based detection (Zeng and Liang, 2007)	- No prior knowledge of the primary user signal and noise is required - Blindly detection - Good computational complexity coming	Good computational complexity coming

Sensing Technique	Advantages	Disadvantages
Machine learning based spectrum sensing(Y. Lu *et al*., 2016; Mikaeil *et al*., 2014)	- Machine learning can detect if trained correctly can be a good approach - Minimize the delay of the detection - Use the complex models in an easy manner	- Complex techniques - Has to be adapted to learning in very fast-changing environments - Features selection affects the detection rate and adds complexity - High dataset has to be built

ED is considered due to its simplicity, dependability, and sensitivity and is preferred over other techniques because of some reasons. The ED method is one of the most commonly used signal-sensing methods in spectrum sensing due to its low implementation complexity. ED can achieve good detection performance when the noise variance is known. However, in most cases, the noise variance is estimated, which may result in uncertainty in noise variance. In the presence of noise variance uncertainty, the detection performance of the ED method may degrade significantly. To reduce the impact of uncertainty in noise variance, an ED-based sensing method is proposed (Luo *et al.*, 2022). In energy detection, the received signal's energy is computed and a threshold is set up. The signal is regarded as present if the energy of the received signal exceeds the threshold; otherwise, it is regarded as absent. The signal is detected by comparing the output of the energy detector with the threshold which depends on the noise floor (Abdulsattar and Hussein, 2012).

The challenges faced by the energy detection method mentioned in Table 2.2 can be tackled using threshold selection and machine learning approaches to analyze and categorize received signals thereby reducing false alarms in energy detection and increase detection precision.

2.3.2.3 Threshold selection

In ED, threshold selection is a typical technique for reducing false alarms. A threshold is used in ED to assess if the incoming signal contains energy or is simply noise. The noise floor, or the amount of background noise present in the signal, is often used to determine the threshold.

The effectiveness of ED depends on the threshold choice. The threshold is set as low as possible to detect weak signals while maintaining the false alarm likelihood below a specific level (Yucek and Arslan, 2009). The detection threshold can be established in several ways, such as using a fixed threshold, an adaptive threshold based on the anticipated noise floor, or a dynamic threshold that changes depending on the received signal's properties.

Fixed thresholds are values that have been predetermined and are used to divide data into various groups or levels. These may not always be the best options because they are frequently selected based on prior knowledge or presumptions about the data. Fixed thresholds, however, have the benefit of being straightforward to use and understand (Yucek and Arslan, 2009).

Contrarily, adaptive thresholds are determined using the qualities of the data themselves, such as mean, variance, or other statistical properties. When the data is noisy or the background changes, adaptive thresholds can be more useful than fixed thresholds because they can adapt to the changes in the data and reduce the number of false positives or false negatives. Adaptive thresholds' key benefit is their capacity to change with the environment and offer greater accuracy in complicated or dynamic contexts. Adaptive thresholding techniques, on the other hand, can need more complex hardware or algorithms and be more computationally demanding. In general, the decision between fixed and adaptive thresholds is based on the particular application and the trade-offs between accuracy and simplicity (Yucek and Arslan, 2009).

To achieve this, one can analyze the noise floor and adjust the threshold as necessary. The Machine Learning Algorithm is quite helpful by analyzing and categorizing received signals based on their characteristics, machine learning techniques enable more precise and reliable detection. This can enhance overall sensor performance and lower false alarm rates.

2.3.3 Machine Learning Algorithm

ML algorithm is computer software made to automatically learn from data and enhance its performance at a task without being explicitly coded. It analyzes data patterns using statistical and mathematical models and then makes predictions or judgments based on the findings. ML is the scientific study of algorithms and statistical models that computer systems use to perform a specific task without being explicitly programmed (Batta, 2018). There are different kinds of ML algorithms, such as reinforcement learning, semi-supervised learning, unsupervised learning, and supervised learning. Popular ML techniques include k-nearest neighbors, neural networks, decision trees, logistic regression, random forests, and linear regression.

To improve the quality of service in the cognitive radio, the machine learning ML algorithm is added to the components. This means with the feature of this aspect of Artificial Intelligence (AI) can help with the prediction of the availability or unavailability of spectrum holes thereby reducing the probability of possible interference of the PU with SU and the handoff timing.

Wang *et al*. (2023) make the case that conventional machine learning methods that call for batch processing of historical data may not be successful in DSA contexts, because the occupancy of the spectrum bands varies quickly and unpredictably. They suggest a Bayesian online learning strategy instead, which can gradually learn from fresh data as it becomes available. To model the occupancy probability distribution of each frequency band, a Gaussian process, and a Dirichlet process are combined. Additionally, they include a Bayesian online learning algorithm that can instantly update the model parameters in response to fresh data. Using a dataset of spectrum occupancy measurements from a cognitive radio network, the authors assess their strategy. A brand-new Bayesian online learning-based method for predicting spectrum occupancy in cognitive radio networks. The method performs better than previous machine learning algorithms in terms of prediction accuracy and processing speed, and it is shown to help manage the dynamic and unpredictable nature of spectrum occupancy in cognitive radio networks.

Ajiboye *et al.* (2021) suggested utilizing the k-nearest neighbor (k-NN) algorithm a machine learning-based method for forecasting spectrum occupancy. The k-NN model is trained and tested using a dataset of measurements of spectrum occupancy. Many performance metrics, such as prediction accuracy, precision, recall, and F1 score, are used to assess the model. Also, the author contrasts the performance of the k-NN model with that of other machine learning techniques, such as decision trees and support vector machines (SVM). According to the findings, the k-NN model can accurately estimate spectrum occupancy with a prediction accuracy of above 90%. The SVM and decision trees, two additional machine learning algorithms, are outperformed by the k-NN model in terms of prediction accuracy and processing speed.

The paper concludes with a machine learning-based k-nearest neighbor algorithm as a method for cognitive radio systems' spectrum occupancy prediction. The strategy is demonstrated to be successful at forecasting spectrum occupancy, outperforming other machine learning methods, and achieving high prediction accuracy.

Arivudainambi *et al*. (2022) suggested a two-step method for spectrum prediction, wherein a feature selection algorithm is used in the first stage to choose the most pertinent features from the input data, and a machine learning algorithm is utilized in the second stage to predict spectrum occupancy. Using a dataset of measurements of spectrum occupancy, the author compares the performance of different machine learning algorithms, such as decision trees, k-nearest neighbors, and support vector machines (SVM). Also, the author contrasts the effectiveness of these algorithms with that of a conventional threshold-based strategy.

The outcomes demonstrate that in terms of prediction accuracy, the machine learning-based technique outperforms the conventional threshold-based approach. For spectrum prediction, the SVM method is demonstrated to be the most successful machine learning technique, obtaining over 90% predictive accuracy. The necessity for a lot of training data and the possibility of overfitting are just two of the approach's drawbacks that the author mentions. The author offers cross-validation and feature selection strategies as potential remedies for these constraints (Batta, 2018; Mohammed *et al.*, 2016).

The research concludes by suggesting a machine learning-based strategy for spectrum forecasting in cognitive radio networks. The method is demonstrated to be successful in enhancing spectrum utilization and decreasing interference, and it is demonstrated that the SVM algorithm is the most successful machine-learning technique for spectrum prediction. The shortcomings of the approach are highlighted in the study, along with potential fixes.

ML systems can enhance the functionality of CR networks by predicting the presence or absence of the main users of the radio spectrum. The ML algorithm can also be used to avoid potential sources of interference in the radio frequency, by taking into consideration the current demand for various frequencies, ML algorithms can be used to optimally allocate radio spectrum to various users. By monitoring and controlling the CR network, ML algorithms may spot problems like congestion, interference, and security risks and take appropriate action. Spectrum handoff is a crucial function of CR which is the change of operating frequency. The main problem in spectrum handoff is the time taken in the searching, selection, and switching to a new available channel which can cause a significant amount of delay during spectrum handoff. This research aims to minimize the delay that occurs during spectrum handoff amongst others such as knowing the current level of spectrum occupancy in the location (Alozie *et al.*, 2022)

To increase the time and energy efficacy of the Spectrum Sensing (SS) process, spectrum prediction is a crucial area of study for CR. Large sensing time and significant energy would be required for SS with a large number of PU channels. By selecting channels that have a high likelihood of being empty during the next time slot, CRs can reduce the number of channels they use for sensing with spectrum prediction (Unadhye *et al*., 2021).

2.3.3.1 Logistic regression

Logistic Regression is a statistical method used for modeling the relationship between a binary dependent variable (that is, one that takes on values of 0 or 1) and one or more independent variables (that is, variables that may be used to predict the dependent variable). The logistic regression model assumes that the relationship between the independent variables and the dependent variable is linear on the log-odds scale. The log-odds (or logit) function is defined as the natural logarithm of the odds, which is the ratio of the probability of the event occurring to the probability of it not occurring (Song *et al.*, 2021).

In the case of binary logistic regression, the dependent variable is binary, and the model estimates the probability of the dependent variable taking the value of 1 (as opposed to 0) as a function of the independent variables. Logistic regression is commonly used in various fields such as medicine, economics, marketing, and social sciences to predict the probability of an event occurring based on one or more predictor variables. It can also be extended to multiclass classification problems using multinomial logistic regression. A Logistic Regression plot in Figure 2.2 is shown and the sigmoid function that can be represented by equation (2.6).

$$h(x) = \frac{1}{1+e^{x}} \tag{2.6}$$

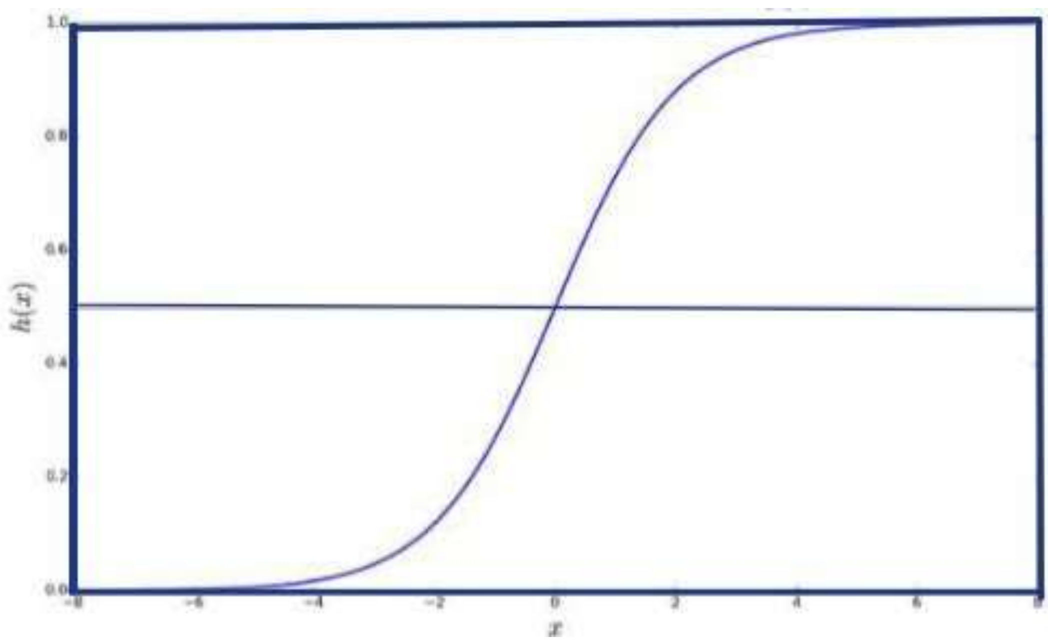

Figure 2.2: Logistic Regression plot

2.3.3.2 Random forest

Random forest is a machine learning algorithm that is used for both regression and classification problems. It is an ensemble learning method that works by constructing a multitude of decision trees during training and outputs the mode or mean prediction of the individual trees. In Random Forest, each decision tree is built using a subset of the training data and a subset of the available features. The tree is constructed by recursively splitting the data into smaller and smaller subsets based on the values of the selected features until the subsets are as homogeneous as possible. The splitting process continues until a stopping criterion is met, such as a maximum depth or a minimum number of samples required splitting a node. During prediction, the random forest algorithm aggregates the predictions of all the decision trees in the forest and returns the mode or mean of the individual predictions, depending on whether the problem is a classification or regression problem.

Random Forest has several advantages over other machine learning algorithms. It is resistant to overfitting and performs well on high-dimensional data. It is also able to handle missing data and can provide estimates of the importance of each feature in the prediction (Cebekhulu *et al.*, 2022). Random forest has been successfully used in various applications, including bioinformatics, remote sensing, and finance. However, it can be computationally expensive and may not be the best choice for very large datasets or real-time applications. Additionally, the interpretability of random forest models can be challenging due to the complexity of the model and the ensemble of decision trees. The Figure 2.3 shows the flowchart of Random Forest.

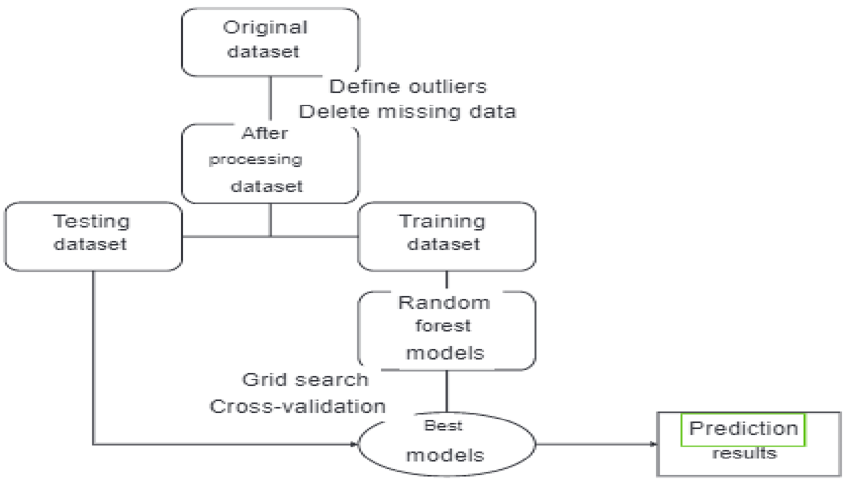

Figure 2.3: The Random Forest Flowchart

2.3.3.3 Decision tree

Decision Tree is a popular machine-learning algorithm used for both classification and regression tasks. It builds a tree-like model of decisions and their possible consequences, where each internal node represents a test on an attribute or feature, each branch represents the outcome of the test, and each leaf node represents a class label or a numeric value.

During training, the decision tree algorithm recursively partitions the training data into smaller and smaller subsets based on the values of the selected features until the subsets are as homogeneous as possible in terms of the class label or numeric value. The partitioning process continues until a stopping criterion is met, such as a maximum depth or a minimum number of samples required to split a node. During prediction, the decision tree algorithm starts at the root node and follows the path of the decision tree based on the values of the selected features, until it reaches a leaf node that represents the predicted class label or numeric value.

Decision trees have several advantages over other machine learning algorithms. They are easy to interpret and can be visualized, making them useful for explaining the reasoning behind the predictions. They can also handle both categorical and continuous data, and can perform feature selection by identifying the most important features in the data (Reddy, 2022).

However, decision trees can suffer from overfitting, which occurs when the tree is too complex and fits the training data too closely, resulting in poor performance on new data. This can be mitigated by using techniques such as pruning, which removes nodes that do not contribute significantly to the accuracy of the model. Decision trees have been successfully used in various applications, including medical diagnosis, customer segmentation, and fraud detection. However, their performance may be limited on large datasets with many features or noisy data, and they may not be the best choice for problems with complex interactions between features (Saber *et al.*, 2020). The Figure 2.4 is the representation of a Decision Tree flowchart.

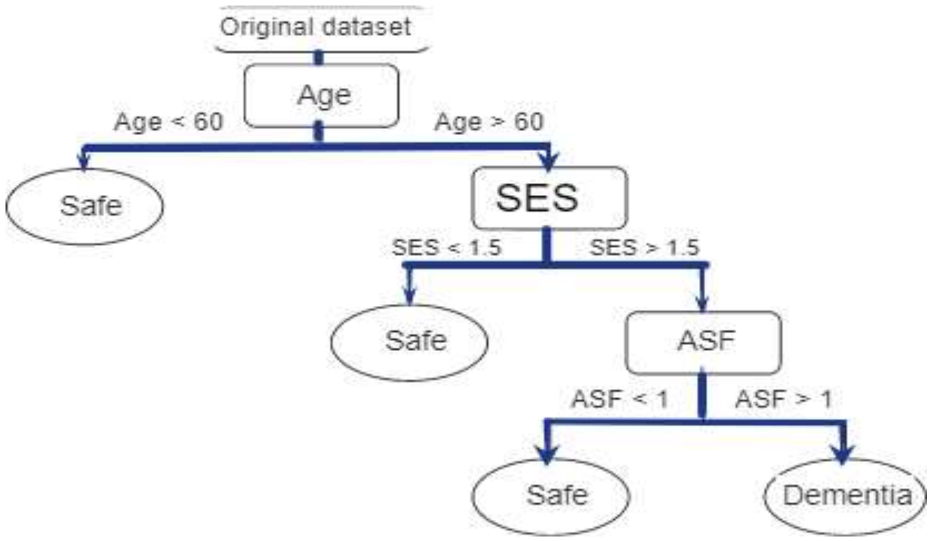

Figure 2.4: Flowchart of a Decision Tree

2.3.3.4 K-Nearest neighbours (KNN)

K-Nearest Neighbours (KNN) is a popular machine learning algorithm used for both classification and regression tasks. It is a non-parametric and instance-based algorithm, meaning that it does not assume any underlying probability distribution for the data, and the model is based on the entire training set.

During training, KNN stores all the training instances as points in a multi-dimensional space, where each feature represents a dimension. When a new instance is presented for prediction, KNN finds the K closest instances in the training set based on a distance metric, such as Euclidean distance, and assigns the class label or numeric value of the new instance as the majority class label or the mean of the K-nearest neighbors, respectively. The value of K, which is a hyperparameter of the algorithm, controls the level of complexity and smoothness of the decision boundary. A small value of K leads to a complex decision boundary that is sensitive to noise and outliers, while a large value of K leads to a smooth decision boundary that may oversimplify the problem (Saber *et al.*, 2020) . KNN has several advantages over other machine learning algorithms. It is simple and easy to implement and can handle both categorical and continuous data. It can also be used for semi-supervised learning, where only a small fraction of the data is labeled, by assigning the majority label of the K-nearest neighbors to the unlabeled instances.

However, KNN can be computationally expensive for large datasets, and the performance of the algorithm may degrade in high-dimensional data due to the curse of dimensionality. Additionally, KNN does not provide any information about the underlying structure of the data or the importance of each feature in the prediction. The Figure 2.5 shows K-Nearest Neighbor Representation and the Euclidean distance (d) is expressed in equation 2.7.

$$\text{Euclidean distance (d)} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \qquad (2.7)$$

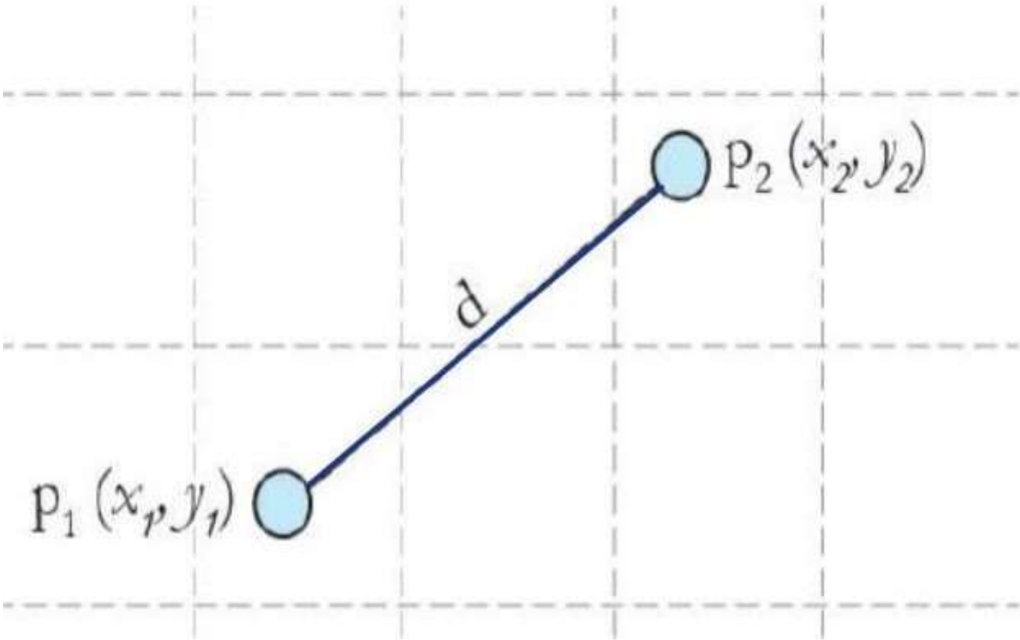

Figure 2.5: K-Nearest Neighbor Representation

2.3.3.5 The XGBoost

Among many ML algorithms or models built, the XGBoost (eXtreme Gradient Boosting) is a popular open-source implementation of gradient boosting, a powerful machine-learning technique used for both regression and classification problems. Tianqi Chen, a Ph.D. candidate at the University of Washington, created XGBoost, which was made available in 2014. The C++-written library offers bindings for several programming languages, such as Python, R, Java, and Julia. The XGBoost has some features that made it stand out from many models. Some of these are regularized learning, high performance, missing data handling, and tree-based learning cross-validation.

The main characteristics of XGBoost, such as regularized learning, tree-based learning, and cross-validation, are introduced and discussed in this study (Chen and Guestrin, 2016; Oyewo *et al.*, 2023) shows that XGBoost outperforms several other well-known machine learning methods on a range of datasets.

LightGBM is a gradient-boosting library that competes with XGBoost and has occasionally been demonstrated to be faster. The authors compare the two libraries on various benchmarks and demonstrate that, on some datasets, LightGBM is quicker than XGBoost (Bentéjac *et al.*, 2020; Ke *et al.*, 2017). In a study of XGBoost, covering its background, salient characteristics, and potential applications, the authors also describe XGBoost's limits and potential future research avenues, as well as the various additions and adjustments that have been made to it since its first release (Reddy, 2022).

The article outlines several XGBoost enhancements and changes that have been done since the software's initial release. These consist of (Graphic Process Unit) GPU acceleration, parallel processing, and distributed computing. The use of XGBoost for multi-class classification, ranking, and regression tasks is also covered by the authors. This work provides a thorough review of XGBoost's several applications, including time-series forecasting, natural language processing, and picture classification. The authors also go into how XGBoost has been used in various Kaggle tournaments and other real-world scenarios. The study examines XGBoost's restrictions as well as possible directions for development. These include the requirement for more sophisticated methods to handle imbalanced datasets and the need for more effective categorical feature-handling algorithms. The possibility of combining XGBoost with other machine learning methods to enhance performance is also covered by the authors (Reddy, 2022).

The functionality of cognitive radio networks has been proven to be improved by ML algorithms which have gained popularity in recent years. These methods can be used to forecast spectrum occupancy, maximize the distribution of radio spectrum, and keep away from probable interference sources. To detect and address issues like congestion, interference, and security concerns in the CR network, ML algorithms may also monitor and control it. Figure 2.6 shows the Flowchart of the XGBoost.

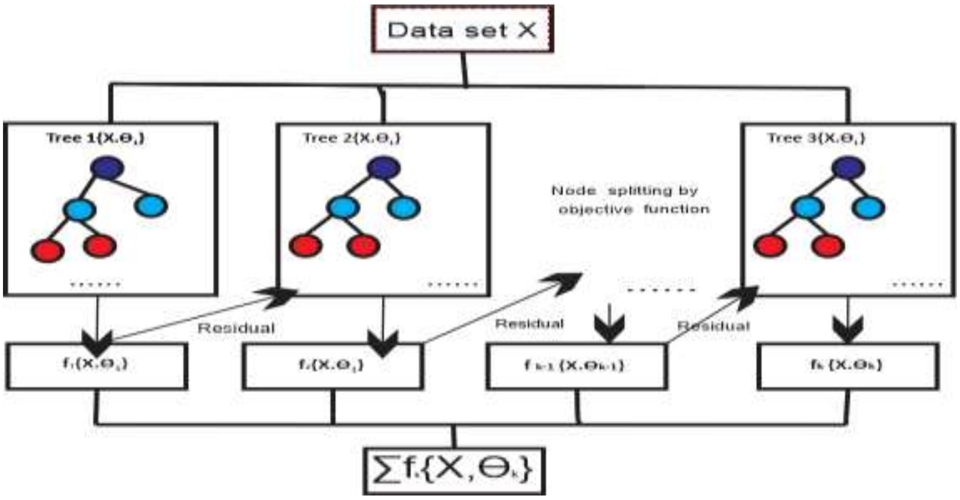

Figure 2.6: Flowchart of a the XGBoost

Table 2.3 shows some of the mentioned ML algorithms' advantages and disadvantages.

Table 2.3: Advantages and disadvantages of ML algorithms

ML Algorithm	Advantages	Disadvantages
K-Nearest Neighbors	- Simple to understand and implement - No training is required, as it stores all training data - Can be used for classification and regression problems	- Computationally expensive for large datasets - Sensitive to the choice of K and the distance metric used - Cannot handle missing data or categorical features easily
Logistic Regression	- Simple and fast algorithm Interpretability: easy to understand how each variable affects the outcome Can be used for classification and regression problems	- Assumes a linear relationship between the features and the outcome - Cannot capture complex nonlinear relationships between features and outcome
Random Forest	- Can handle both categorical and numerical features - Robust to outliers and missing data - Can handle high-dimensional data - Low risk of overfitting	- Black box model, difficult to interpret - Can be computationally expensive for large datasets - Difficult to tune hyperparameters
Extreme Gradient Boosting	- Highly accurate and performs well on a wide range of problems - Can handle missing data and outliers - Can handle high-dimensional data - Fast and scalable	- Can be prone to overfitting if hyperparameters are not tuned correctly - Black box model, difficult to interpret - Requires more computational resources than other algorithms

ML Algorithm	Advantages	Disadvantages
Decision Trees	- Simple and easy to understand - Can handle both categorical and numerical features - Can capture complex nonlinear relationships between features and outcome	- Prone to overfitting, particularly when the tree is deep - Sensitive to small changes in the data - Can be unstable and produce different trees with different splits

CHAPTER THREE

3.0 RESEARCH METHODOLOGY

3.1 Introduction

The system design consists briefly of two major stages:

i. The spectrum sensing stage.

ii. The data analysis with the ML algorithm for the prediction of spectrum holes

Figure 3.1 is a block diagram that shows the steps or the process involved in the two stages

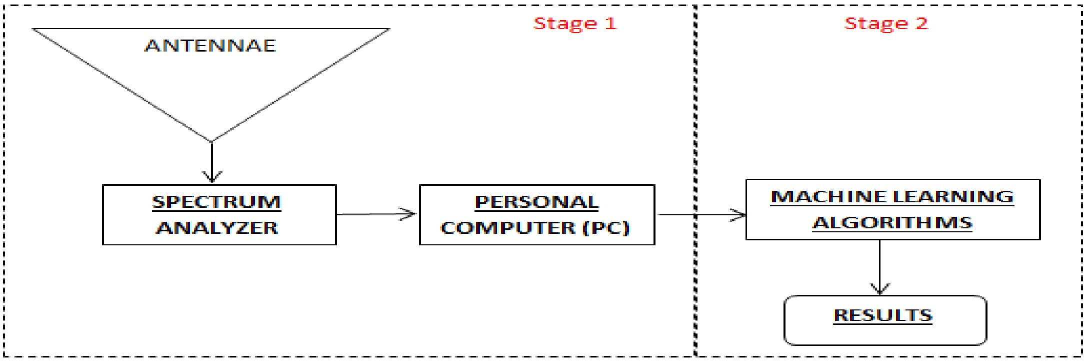

Figure 3.1: Methodology of System Architecture

3.2 Data Collection

In ML projects, data plays an important role. To collect our data to run analysis, a high-gain outdoor antenna with an acceptable capability frequency range of 47 MHz to 1 GHz was connected to the Agilent spectrum analyzer to capture electromagnetic signals. It is good to note that the data captured with the spectrum was an outdoor exercise carried out within three hours. The components used for the acquisition of data are as follows.

i. Gotv Antennae: In the VHF and UHF frequency bands, the typical frequency range of a GOtv antenna of frequency range of 47 MHz to 1 GHz. This antenna is connected to the spectrum analyzer. A personal computer (PC): laptop HP 250G was used to interface with the Agilent Spectrum analyzer with a USB cord. The PC is used to save the dataset after each sweep of the case study.

ii. A generator was available as a backup for an uninterrupted power supply.

iii. The Global Position System (GPS) is used to find a suitable location to carry out the outdoor survey.

iv. An Agilent Spectrum Analyzer N9342C has the following features as seen in Table 3.1.

Table 3.1: Features of Agilent N9342C Spectrum Analyzer

Features	**Value**
Frequency range	100 kHz to 7 GHz
Display resolution	640 x 480 pixels
RBW	1 Hz to 3 MHz
DANL	-155 dBm/Hz (@1 GHz, preamp off)
Phase noise	-100 dBc/Hz (@10 kHz offset)
Amplitude accuracy	±0.5 dB (@25°C ±5°C)
Maximum input power	+30 dBm (1 W)
Frequency range of study	80MHz – 1 GHz
Sweep time	413.5s
Trace Point	222

Driving around the town, using the GPS, a few locations were noticed but the MORRIS fertilizers (LAT N9^{o}35'37.374" and LONG E 6^{o} 32'12.858") was chosen based on their more elevated position and the region situated at a place where there is a good number of line of sight to base stations. This place is located in the city of Minna, Niger State in Nigeria. The spectrum analyzer is made to run for 3 hours and the CSV (comma-separated values) file is saved after each of the 13 sweeps. During the duration of the exercise, thirteen sweep was taken and eleven was used to ensure accuracy of result. Plate I and Plate II shows the set-up of the spectrum measurement campaign and the Back-up power system for the campaign.

Plate I: The connection of the Agilent spectrum analyzer and a PC

Plate II: Back-up Power Source

The CSV files were acquired in a format as the dataset to be used to train a model of the ML Algorithm. To have a proper record of the data captured, the spectrum analyzer is interfaced with a personal computer (laptop) to save captured data. Also, the laptop was configured to be able to control the Agilent spectrum analyzer interfacing it with a USB cord to the PC using Keysight HSA and BSA software alongside the Agilent Library IO office suite.

It is quite convenient to collect the data from the personal computer which will store all the information collected by the spectrum analyzer. This will serve as the source of the dataset used as the input for the ML algorithm. The dataset was cleaned. This involves removing errors, handling missing values, and addressing outliers to ensure that the data is accurate and reliable for training a model. The energy detector narrowband sensing technique was used since it has no need for prior information about the primary signal is required.

3.3 Determining Threshold

To carry out this research effectively, the dataset was gotten by sweeping across the frequency band in the spectrum. The threshold is determined using the energy detection method, by taking into cognizance the power sensed in the frequencies of the spectrum. Using equation 3.1, the effective threshold power can be calculated. For effective usage of the spectrum holes (unoccupied frequency), the threshold is set at the Power Threshold Calculation

Given:

Spectrum Analyzers' Noise floor power $P_{NF} = -120$ dBm

Required signal-to-noise ratio (SNR) = 20 dB

Bandwidth B = 1 MHz

To calculate the power threshold (P_{TH}) for energy detection, we use the formula:

$$P_{TH} = P_{NF} + SNR + 10log10(B) \tag{3.1}$$

Substituting the given values:

$$P_{TH} = -120 + 20 + 10log10(1) = -100dBm$$

Therefore, the power threshold for energy detection is −100 dBm. This is done so that the false alarm error is reduced to a minimal level.

3.4 Training and Prediction

The dataset was collected in CSV format. These files were analyzed using Python 3.0 in the Jupyter notebook. The data preprocessing was done to check if the data is clean. That is, to know if there are missing values or not. Also, the dataset was checked to know the type; whether it is a discreet or classification (yes or no) problem. The dataset from the spectrum sensing exercise is continuous.

The dataset was divided into training and testing datasets. To train the dataset, 70% of the dataset was used to train while the other 30% was used to test the functionality of the prediction exercise (Nagulapati *et al*., 2021). The dataset was grouped into the predictor and the target (what is being predicted).The target is the frequencies and the predictors are the powers. Using powers to predict frequencies occupied or not occupied.

The following ML algorithms Logistic regression, Random forest regression, Decision Tree, K- Nearest Neighbour, and the XGBoost were trained using the dataset gotten from the sensed spectrum. Each of the above-mentioned ML algorithms were trained and tested.

The mathematical representation of the XGBoost algorithm involves two main components: a loss function to be optimized and a regularized objective function to control model complexity. The objective function can be written as:

$$Obj = L\left(y_i, \hat{y}_i\right) + \Omega\left(f\right) \tag{3.2}$$

Where L is the loss function, y_i is the true label for the i-th observation, $\hat{y}_i$ is the predicted label, f is the ensemble of decision trees, and Ω(f) is a regularization term that penalizes complex models.

3.4.1 Trained model

The flowchart in Figure 3.2 fully describes the performance of the operation different stages. The model is retrained and tested, if the performance is not satisfactory.

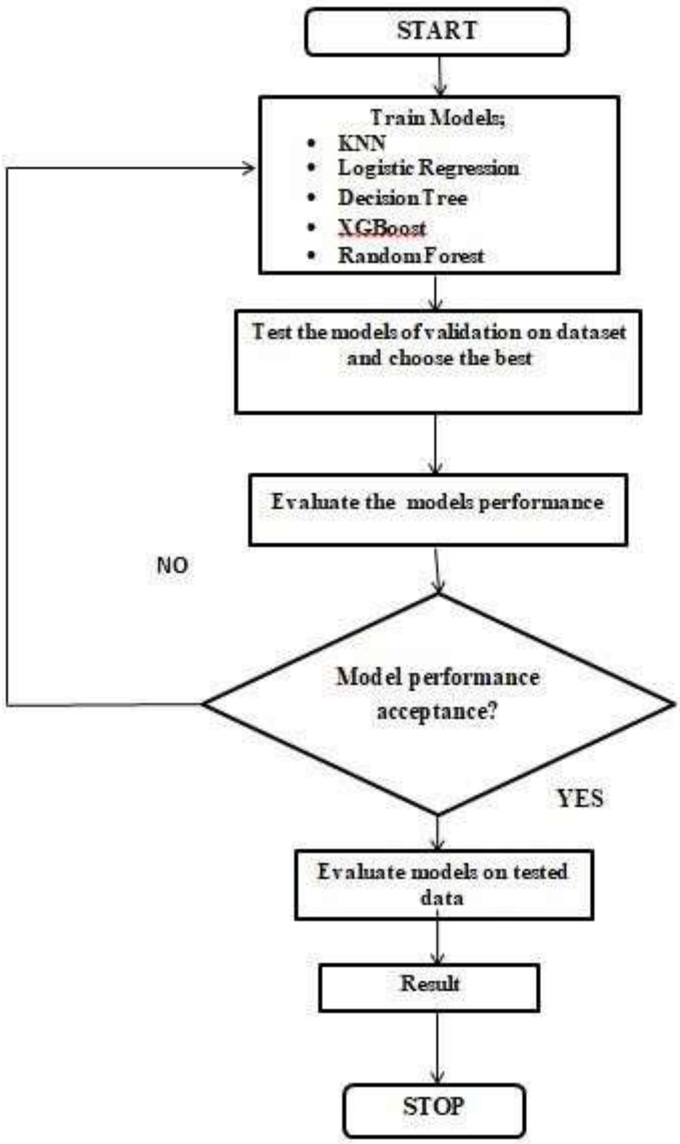

Figure 3.2: ML algorithm prediction flowchart

CHAPTER FOUR

4.0 RESULTS AND DISCUSSION

4.1 Introduction

The sensing and prediction of the spectrum were carried out in stages in which the first stage was the acquisition of the dataset in the field and stage two was to use the acquired dataset to train and to test different ML algorithms alongside the XGBoost to show its efficacy in the degree of accuracy.

4.2 Results

Table 4.1 shows the sample of the collected data from sensing the spectrum with the Agilent SA. A chunk of the dataset is presented in the Appendix A.

Table 4.1: Sample of cleaned dataset signal power for some trace point

Frequency (Hz)	8.00E+07	8.42E+07	8.83E+07	9.25E+07	9.67E+07	1.01E+08	1.05E+08	1.09E+08
	Data points 1	Data points 2	Data points 3	Data points 4	Data points 5	Data points 6	Data points 7	Data points 8
Sweep 1	-1.07E+02	-1.07E+02	-4.71E+01	-6.57E+01	-9.08E+01	-8.89E+01	-9.81E+01	-8.73E+01
Sweep 2	-1.06E+02	-1.06E+02	-5.39E+01	-7.76E+01	-8.59E+01	-9.13E+01	-9.86E+01	-9.25E+01
Sweep 3	-1.08E+02	-1.05E+02	-5.22E+01	-7.17E+01	-8.85E+01	-8.78E+01	-9.75E+01	-9.33E+01
Sweep 4	-1.09E+02	-1.07E+02	-3.98E+01	-7.38E+01	-8.85E+01	-8.78E+01	-9.75E+01	-9.33E+01
Sweep 5	-1.08E+02	-1.05E+02	-4.99E+01	-7.19E+01	-8.87E+01	-8.77E+01	-9.68E+01	-9.39E+01
Sweep 6	-1.04E+02	-1.14E+02	-4.81E+01	-6.79E+01	-9.87E+01	-8.88E+01	-1.01E+02	-8.38E+01
Sweep 7	-1.07E+02	-1.05E+02	-5.25E+01	-7.78E+01	-1.01E+02	-9.11E+01	-1.06E+02	-9.00E+01
Sweep 8	-1.06E+02	-1.08E+02	-5.34E+01	-6.67E+01	-9.69E+01	-9.48E+01	-1.06E+02	-9.34E+01
Sweep 9	-1.08E+02	-1.06E+02	-5.39E+01	-6.97E+01	-9.66E+01	-9.66E+01	-1.05E+02	-9.14E+01
Sweep 10	-1.04E+02	-1.08E+02	-5.14E+01	-6.59E+01	-9.63E+01	-9.56E+01	-1.02E+02	-9.47E+01
Sweep 11	-1.04E+02	-1.04E+02	-4.09E+01	-7.02E+01	-9.88E+01	-9.09E+01	-1.05E+02	-1.01E+02

4.2.1 Spectrum occupancy level of the dataset collected

The data point after each sweep across the spectrum from the 80 MHz to the 1 GHz is 222 data point. Applying the formula,

$$Spectrum\ Occupancy\ Level\ = \frac{Number\ of\ occupied\ data\ points}{Total\ number\ of\ data\ points} \times 100\% \qquad (4.1)$$

Given: Number of occupied data points = 141

Total number of data points = 222

$$Spectrum\ Occupancy\ Level\ = \frac{141}{222} \times 100\%$$

$$\approx 63.51\%.$$

The Table 4.2 shows the occupancy percentage after each sweep of the spectrum.

Table 4.2: Spectrum Occupancy Level

Sweep Number	Spectrum Occupancy (%)
1	63.51
2	62.16
3	61.26
4	60.81
5	60.36
6	59.91
7	59.46
8	59.01
9	58.56
10	58.11
11	57.66
12	57.21
13	56.76

The sample dataset for spectrum analysis was acquired at Morris Fertilizer and this collated data as seen in Table 4.1, was used as a dataset to train and test the following ML algorithm using the powers of each frequency as a predictor in Python 3 software.

4.2.2 Analysis of machine learning models

Various ML models were designed and results were obtained. The results obtained using the five selected classification algorithms (Logistic Regression, K-NN, Random Forest, XGBoost, and Decision Tree) are presented.

4.2.2.1 Logistic regression classifier

The test accuracy, precision, recall and f1-score are 90.43%, 90.40%, 93.39% and 91.43% respectively was obtained with the application of logistic regression as shown in Figure 4.1. The dataset was divided into two with 70% of the data used in training and 30% used in testing. Also, Figure 4.1 shows the confusion matrix of the output of the test carried out. This display that out of 240 frequency channels not occupied, the logistic Regression algorithm made the mistake of 8 predictions while when the channel was occupied it made a mistake of 24 predictions.

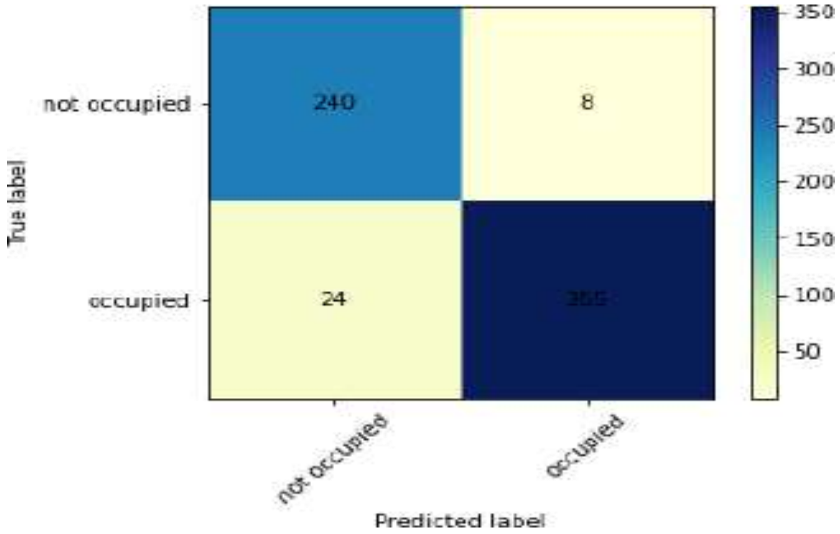

Figure 4.1: The Confusion Matrix for the Logistic Regression

4.2.2.2 Random forest classifier

The test accuracy, precision, recall and f1- score are 99.52%, 99.98%, 99.17% and 99.57% respectively was obtained with the application of the Random Forest as shown in Figure 4.2 as the confusion matrix is displayed. The dataset was divided into two with 70% of the data used for training and 30% used for testing. The Random Forest (RF) made a mistake of one channel as occupied when it was not occupied and it also made a mistake of seeing three channels as unoccupied when they were actually occupied.

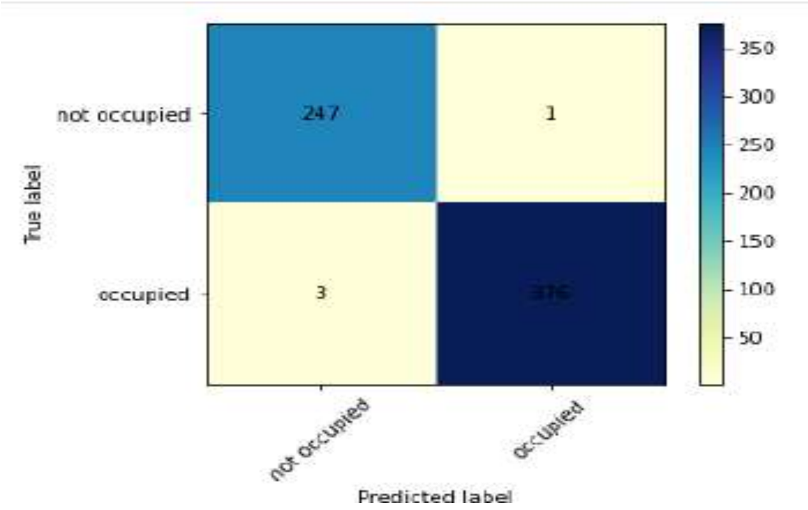

Figure 4.2: The Confusion Matrix for the Random Forest

4.2.2.3 Decision tree classifier

The test accuracy, precision, recall and f1- score are 99.52%, 99.99%, 99.17%, and 99.58% respectively was obtained with the application of the Decision Tree as shown in Figure 4.3. The dataset was divided into two with 70% of the data used for training and 30% used for testing. The decision Tree classifier gave similar results as the Random Forest, showing an equal number of prediction errors.

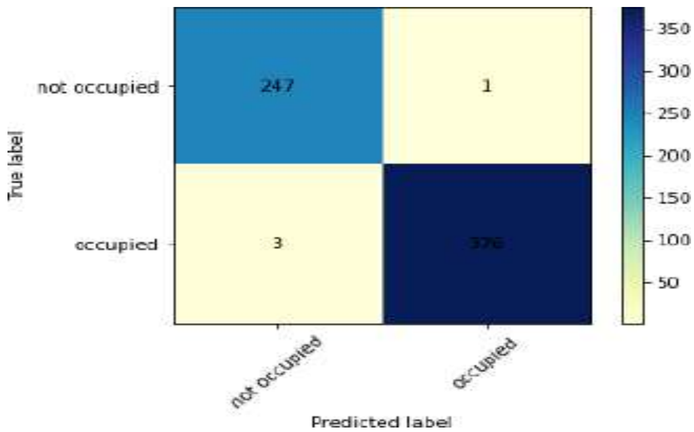

Figure 4.3: The Confusion Matrix for the Decision Tree

4.2.2.4 XGBoost classifier

The test accuracy, precision, recall and f1- score are 99.52%, 100.00%, 99.17% and 99.58% respectively was obtained with the application of the XGBoost as shown in Figure 4.4. The dataset was divided into two with 70% of the data used for training and 30% used for testing. After the test was carried out, it was discovered that the XGBoost made only three errors in prediction, seeing two channels that were not occupied as occupied and one channel that was occupied as occupied.

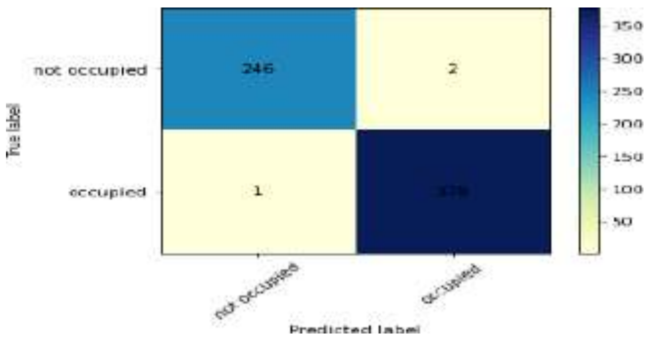

Figure 4.4: The Confusion Matrix for the XGBoost

4.2.2.5 K-Nearest neighbour classifier

The test accuracy, precision, recall and f1-score are 97.61%, 100.00%, 95.87% and 97.89% respectively was obtained with the application of the KNN as shown in Figure 4.5. The dataset was divided into two with 70% of the data used for training and 30% used for testing. In the K-nearest Neighbour, only four errors were made as the model predicted three occupied channels as not occupied and one channel that is not occupied as occupied.

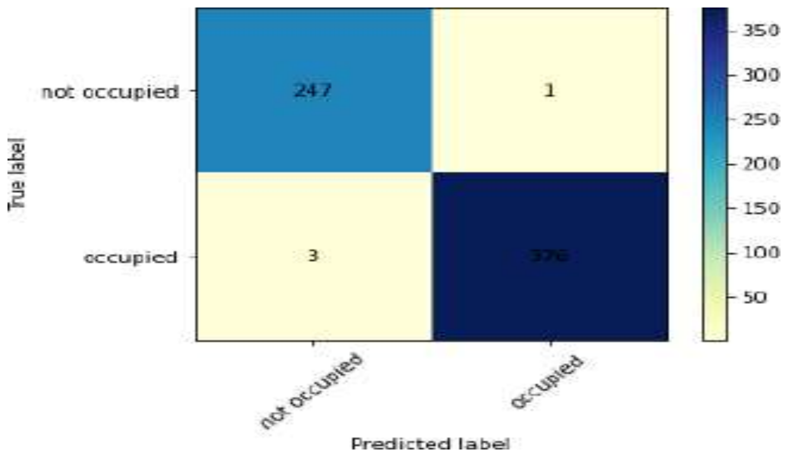

Figure 4.5: The Confusion Matrix for the K-Nearest Neighbour

Table 4.2 is the test score as well as the training score are presented in percentage.

Table 4.3: Tabular representation of results.

Serial Number	ML algorithm	Train Score %	Test Accuracy %	Precision %	Recall %	F1-score %
1	Logistic Regression	94.84	90.43	90.40	93.39	91.43
2	Random Forest	99.93	99.52	99.98	99.17	99.57
3	Decision Tree	99.93	99.52	99.99	99.17	99.58
4	XGBoost	99.86	99.52	100.00	99.17	99.58
5	K-Nearest Neighbour	98.19	97.61	100.00	95.87	97.89

4.3 Results and Plot

Figure 4.6 shows the graph of the evaluation plot displaying the trained model against their f1 scores in percentages. XGBoost with a lower training score than other models like Decision Tree and Random Forest gave a high performance evaluation.

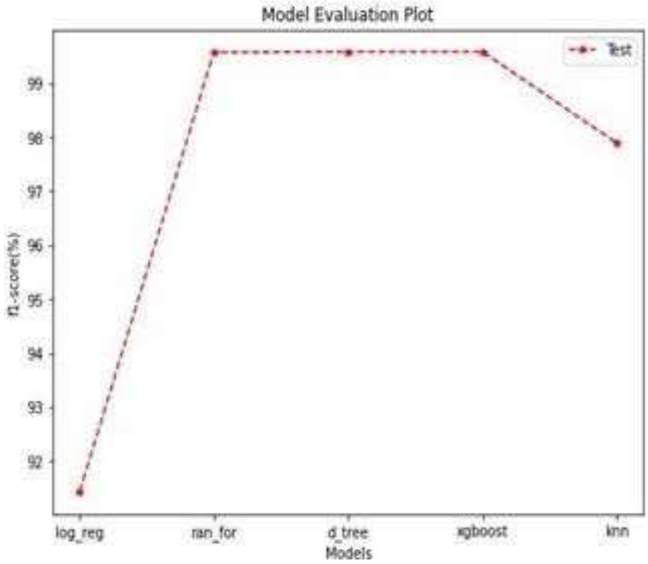

Figure 4.6 A graph of the train score and models

In Table 4.3, the XGBoost was used to test random signals in decibels, in which if the output displays “1”, the frequency is occupied and if the output displays “0”, this means the frequency is not occupied. It was proved that the degree of accuracy is exceptional.

The accuracy, the precision and the F1-score as presented in Table 4.4 was derived by the formulas:

Accuracy, describing the number of correct predictions over all predictions:

$$Accuracy = \frac{Numbers\ of\ correct\ predictions}{Numbers\ of\ \mathrm{all}\ predictions} \tag{4.2}$$

Precision is a measure of how many of the positive predictions made are correct (true positives).

$$\Pr ecision = \frac{True\ positives}{True\ positive\ and\ false\ positive} = \frac{Numbers\ of\ correct\ predicted\ positive\ instance}{Numbers\ of\ total\ positive\ predictions} \tag{4.3}$$

F1-Score is a measure combining both precision and recall while Recall is a measure of how many of the positive cases the classifier correctly predicted, over all the positive cases in the data. It is sometimes also referred to as Sensitivity.

$$F1-score = 2\times \frac{Precison \times \operatorname{Re}call}{Precison + \operatorname{Re}call} \tag{4.4}$$

Table 4.4: Testing the model with random power in (-dBm)

S/N	Power (-dBm)	ON / OFF	STATUS
1	69.10	1	OCCUPIED
2	70	0	UNOCCUPIED
3	100	0	UNOCCUPIED
4	100.1	1	OCCUPIED

4.4 Discussion of Result

As shown in Table 4.3, the degree of accuracy of the XGBoost, when tested, supersedes the other ML algorithm in performance. This is the main reason why the XGBoost algorithm was selected for the random prediction as seen in Table 4.4.

The retrained model of the XGBoost algorithm was used to test the random signal point of -69.10dbm. The algorithm sees the trace point as an occupied channel because any signal higher than the -70dBm as occupied. Also, the retained algorithm sees any frequency of the trace point below -100dBm as noise. That is, any signal below is seen as occupied which makes it unavailable for any secondary user.

According to Table 4.4, the trace point -100.1dBm was tested, which mathematically was approximately -100dBm. Although these trace point are in discretely close, the trained algorithm sees the -100.1 dBm as occupied because of the degree of accuracy. It is clear according to the result that the XGBoost will see in the spectrum the signal power of any trace point as unoccupied in an much if falls within the range of -100dBm (the threshold power signal) and the -70dBm , which signifies that the frequency of the trace point is presently free for the secondary user to engage.

This clearly signifies that the XGBoost take any signal that falls in between the range of -100dBm and -120dBm (the noise floor power signal) as occupied, since the weak signals can be disturbed by the noise, thereby making the quality of service poor for the CR user. Whenever the power signal trace point signals higher than -70dBm, such as -69.10dbm, the Xgboost sees it as occupied.

CHAPTER FIVE

5.0 CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

Sensing the spectrum to know the level of occupancy, for a cognitive radio user to engage the spectrum holes, can have certain setbacks such as the handoff time. This can lead to the interference of the secondary user with the primary user. This factor affects the quality of service of the spectra. Although ED is one of the simplest ways to know if a certain frequency within a spectrum is occupied, it has some disadvantages such as false alarms as well of its unreliability to detected properly at low SNR. To solve this challenge that follows the usage of the ED, the ML technique is used to produce an effective system to determine the occupied and unoccupied frequency within the spectrum sensed.

It was seen that out of the dataset that was trained and tested from the data collated from the sensed spectrum, the XGBoost gave a test accuracy of 99.52%. The logistic regression gave the least accuracy at 90.43%. The Decision Tree and the Random Forest, gave an accuracy of 99.52% each while the KNN gave the accuracy of 97.61%. Since the XGBoost gave the highest accuracy, it is used to train the dataset amongst the highlighted because it gave the highest prediction accuracy.

5.2 Recommendations

While XGBoost has been shown to provide excellent predictive performance, its models can be challenging to interpret. Research into methods for interpreting XGBoost models, such as feature importance (a technique used to determine the relative importance or contribution of different features which can also be called variables or predictors in a machine learning model ranking) or partial dependence plots (a visual representation that shows the relationship between a predictor and the predicted outcome of a machine learning model). These could improve the transparency and trustworthiness of XGBoost-based predictions.

Investigate the use of XGBoost in online learning scenarios. Online learning involves continually updating a model as new data becomes available. XGBoost may be able to adapt to new data more effectively than other algorithms, making it a promising choice for online learning applications.

5.3 Contribution to Knowledge

The research developed an improved predictive modeling techniques and demonstrating the effectiveness of XGBoost compared to other popular machine learning algorithms. As seen in Table 4.3, the test accuracy, precision, Recall and F1- score are 99.52%, 100.00%, 99.17% and 99.58%, respectively. This shows the best degree of accuracy compared to commonly used ML algorithms such as Random Forest, Logistic Regression, Decision Tree, and KNN. By leveraging on XGBoost, telecommunication companies can enhance predictive accuracy, leading to improved network performance, optimized resource allocation, and advanced customer analytics. Also, this will provide a competitive advantage, which will enable proactive network optimization, better resource management, personalized marketing campaigns, increased customer satisfaction, and ultimately, improved business outcomes.

REFERENCES

Abdulsattar, M. A., & Hussein, Z. A. (2012). *Energy detection technique for spectrum sensing in cognitive radio : a survey*. *4*(5), 223–242.

Ajiboye, J. A., Adegboye, B. A., Aibinu, A. M., Kolo, J. G., Ajiboye, M. A., & Usman, A. U. (2021). Ensemble Autoregressive Neural Network (ARNN) Model for Spectrum Occupancy Prediction. International Conference on Artificial Intelligence and Robotics. *Ensemble Autoregressive Neural Network (ARNN) Model for Spectrum Occupancy Prediction*, *1*(Icair), 21-22.

Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., & Ayyash, M. (2015). Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. *IEEE Communications Surveys & Tutorials*, *17*(4), 2347–2376. https://doi.org/10.1109/COMST.2015.2444095

Alom, Z., Godder, T., Morshed, M., & Maali, A. (2017). *Enhanced Spectrum Sensing Based on Energy Detection in Cognitive Radio Network using Adaptive Threshold*. https://doi.org/10.1109/NSysS.2017.7885815

Alozie, E., Faruk, N., Oloyede, A. A., Sowande, O. A., & Lucky, A. (2022). *Intelligent Process of Spectrum Handoff in Cognitive Radio Networks*. *4*(1), 205–218.

Arivudainambi, D., Mangairkarasi, S., & Varun Kumar, K. A. (2022). Spectrum prediction in cognitive radio network using machine learning techniques. *Intell. Autom. Soft Comput*, *32*(3), 1525-1540.

Arjoune, Y., & Kaabouch, N. (2019). A comprehensive survey on spectrum sensing in cognitive radio networks: Recent advances, new challenges, and future research directions. *Sensors*, *19*(1), 126.

Arjoune, Y., El Mrabet, Z., El Ghazi, H., & Tamtaoui, A. (2018, January). Spectrum sensing: Enhanced energy detection technique based on noise measurement. In *2018 IEEE 8th annual computing and communication workshop and conference (CCWC)* (pp. 828-834). IEEE.

Arshid, K., Jianbiao, Z., Hussain, I., Pathan, M. S., Yaqub, M., Jawad, A., Munir, R., & Ahmad, F. (2022). Energy efficiency in cognitive radio network using cooperative spectrum sensing based on hybrid spectrum handoff. *Egyptian Informatics Journal*, *23*(4), 77–88. https://doi.org/https://doi.org/10.1016/j.eij.2022.06.008

Bani, K., & Kulkarni, V. (2022). Hybrid Spectrum Sensing Using MD and ED for Cognitive Radio Networks. *Journal of Sensor and Actuator Networks*, *11*(3), 36–42.

Batta, M. (2020). Machine learning algorithms-a review. *International Journal of Science and Research (IJSR).[Internet]*, *9*(1), 381-386.

Bentéjac, C., Csörgo, A., & Martínez-Muñoz, G. (2020). A comparative analysis of gradient boosting algorithms. *Artificial Intelligence Review*, *54*, 1937–1967.

Bönsch, A., & Kuhlen, T. W. (2020, March). Joint Dual-Tasking in VR: Outlining the Behavioral Design of Interactive Human Companions WhoWalk and Talk with a User. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)* (pp. 167-168). IEEE.

Cebekhulu, E., Onumanyi, A. J., & Isaac, S. J. (2022). Performance analysis of machine learning algorithms for energy demand–supply prediction in smart grids. *Sustainability*, *14*(5), 2546-2552.

Chen, L. C., Zhu, Y., Papandreou, G., Schroff, F., & Adam, H. (2018). Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 801-818).

Chen, T., & Guestrin, C. (2016, August). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (pp. 785-794).

Chukwuchekwa, N., U, E. J., Ezema, L. S., & Agubor, C. K. (2021). *Investigation of radio frequency licensed spectrum utilization in Nigeria : a study of r umuokwuta , port harcourt , Nigeria*. *13*(5), 31–41. https://doi.org/10.5121/ijwmn.2021.13503

Cohen, D., & Eldar, Y. C. (2017). Compressed Cyclostationary Detection for Cognitive Radio. *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 3509–3513. https://doi.org/10.1109/ICASSP.2017.7952809

Ding, G., Jiao, Y., Wang, J., Zou, Y., Wu, Q., Yao, Y., & Hanzo, L. (2018). Spectrum Inference in Cognitive Radio Networks: Algorithms and Applications. *IEEE Communications Surveys & Tutorials*, *20*, 150–182.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2020). Generative adversarial networks. *Communications of the ACM*, *63*(11), 139-144.

Hussain, F., Hassan, S. A., Hussain, R., & Hossain, E. (2020). Machine learning for resource management in cellular and IoT networks: Potentials, current solutions, and open challenges. *IEEE communications surveys & tutorials*, *22*(2), 1251-1275.

Lu, Q., Yang, S., & Liu, F. (2017). Wideband Spectrum Sensing Based on Riemannian Distance for Cognitive Radio Networks. In *Sensors* (Vol. 17, Issue 4), 661-667. https://doi.org/10.3390/s17040661

Lu, Y., Zhu, P., Wang, D., & Fattouche, M. (2016). Machine learning techniques with probability vector for cooperative spectrum sensing in cognitive radio networks. *2016 IEEE Wireless Communications and Networking Conference*, 1–6.

Luo, J., Zhang, G., & Yan, C. (2022). An energy detection-based spectrum-sensing method for cognitive radio. *Wireless Communications and Mobile Computing*, *2022*,(pp. 3336-3342).

Mikaeil, A., Guo, B., & Wang, Z. (2014). Machine Learning to Data Fusion Approach for Cooperative Spectrum Sensing. In *Proceedings - 2014 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, CyberC 2014*. https://doi.org/10.1109/CyberC.2014.80

Mishra, Y., & Chaudhary, V. S. (2023). *Spectrum Sensing in Cognitive Radio for Internet of Things using Deep Learning Models*. *15*(1). https://doi.org/10.18090/samriddhi.v15i01.04

Mohammed, M. M. Z. E., Bashier, E., & Bashier, M. (2016). *Algorithms and Applications* (Issue December). https://doi.org/10.1201/978131537165

Nagulapati, V. M., Lee, H., Jung, D., Brigljevic, B., Choi, Y., & Lim, H. (2021). Capacity estimation of batteries: Influence of training dataset size and diversity on data driven prognostic models. *Reliability Engineering & System Safety*, *216*, 108048-108054.

Nasser, A., Al Haj Hassan, H., Abou Chaaya, J., Mansour, A., & Yao, K.-C. (2021). Spectrum Sensing for Cognitive Radio: Recent Advances and Future Challenge. *Sensors*, *21*(7), 2408. https://doi.org/10.3390/s21072408

Nasser, A., Hassan, H. A. H., Chaaya, J. A., Mansour, A., & Yao, K. C. (2021). Spectrum sensing for cognitive radio: Recent advances and future challenge. *Sensors*, *21*(7), 1–29. https://doi.org/10.3390/s21072408

Onumanyi, A. J., Onwuka, L., Adeiza, O., Onwuka, E. N., Ugweje, O., & Salami, M. J. E. (2013). *Spectroscopic Measurements and Analysis of Frequency Occupancy A Case Study of Minna, Niger State*. https://www.researchgate.net/publication/264635532

Oyewo, T. A., Dauda, U. S., & Mohammed, A. S. (2023). and Academic Research Spectrum Occupancy Prediction Using A Modified XGBoost. *2nd International Conference on Scientific and Academic Research*, *1*(ISBN: 978-605-72325-7-1), (pp. 232-237).

Ranjan, A., Anurag, & Singh, B. (2016). Design and analysis of spectrum sensing in cognitive radio based on energy detection. *2016 International Conference on Signal and Information Processing (IConSIP)*, 1–5. https://doi.org/10.1109/ICONSIP.2016.7857444

Reddy, B. S. K., & Deevi, N. (2022). Experimental Testing of Primary User Detection Using Decision Tree Algorithm With Software Defined Radio Testbed.

Saber, M., El Rharras, A., Saadane, R., Chehri, A., Hakem, N., & Kharraz, H. A. (2020). Spectrum sensing for smart embedded devices in cognitive networks using machine learning algorithms. *Procedia Computer Science*, *176*, 2404–2413. https://doi.org/10.1016/j.procs.2020.09.311

Saadallah, N. R., Alhafid, A. K., Al-Tayyar, H. A., & Ali, Q. I. (2021). A Survey of Different Security Methods in the Physical Layer of TCP/IP Model. *Eurasian Research Bulletin*, *2*, 11-27.

Salameh, H. A. B., Almajali, S., Ayyash, M., & Elgala, H. (2018). Spectrum Assignment in Cognitive Radio Networks for Internet-of-Things Delay-Sensitive Applications Under Jamming Attacks. *IEEE Internet of Things Journal*, *5*(3), 1904–1913. https://doi.org/10.1109/JIOT.2018.2817339

Sipper, M., & Moore, J. H. (2022). AddGBoost: A gradient boosting-style algorithm based on strong learners. *Machine Learning with Applications*, *7*, 100243-100247.

Sivagurunathan, P. T., Ramakrishnan, P., & Sathishkumar, N. (2021). Recent Paradigms for Efficient Spectrum Sensing in Cognitive Radio Networks: Issues and Challenges. *Journal of Physics: Conference Series*, *1717*(1), 12057. https://doi.org/10.1088/1742-6596/1717/1/012057

Solanki, S., Dehalwar, V., & Choudhary, J. (2021). Deep learning for spectrum sensing in cognitive radio. *Symmetry*, *13*(1), 1–15. https://doi.org/10.3390/sym13010147

Song, X., Liu, X., Liu, F., & Wang, C. (2021). Comparison of machine learning and logistic regression models in predicting acute kidney injury: A systematic review and meta-analysis. *International journal of medical informatics*, *151*, 104484-104488.

Upadhye, A., Saravanan, P., Chandra, S. S., & Gurugopinath, S. (2021, July). A survey on machine learning algorithms for applications in cognitive radio networks. In *2021 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)* (pp. 01-06). IEEE.

Usman, M. B., Singh, R. S., Mishra, S., & Rathee, D. S. (2022). Improving Spectrum Sensing for Cognitive Radio Network Using the Energy Detection with Entropy Method. *Journal of Electrical and Computer Engineering*,p 2-3, *2022*. https://doi.org/10.1155/2022/2656797

Wang, J., & Liu, B. (2021, April). A brief review of machine learning algorithms for cooperative spectrum sensing. In *Journal of Physics: Conference Series* (Vol. 1852, No. 4, p. 042094). IOP Publishing.

Wang, X., Chen, Q., & Yu, X. (2023). Research on Spectrum Prediction Technology Based on Back Propagation-Long short-term memory Time Forecasting. *Electronics*, *12*(1), 247.

Wu, Q., Ng, B. K., & Lam, C. T. (2022). Energy-Efficient Cooperative Spectrum Sensing Using Machine Learning Algorithm. *Sensors*, *22*(21),pp. 8230-8234.

Xinzhi Z. , Chai, R., & Gao, F. (2014, December). Matched filter based spectrum sensing and power level detection for cognitive radio network. In *2014 IEEE global conference on signal and information processing (GlobalSIP)* (pp. 1267-1270). IEEE.

Yucek, T., & Arslan, H. (2009). A survey of spectrum sensing algorithms for cognitive radio applications. *IEEE Communications Surveys & Tutorials*, *11*(1), 116–130. https://doi.org/10.1109/SURV.2009.090109

Zehra, S. S., Qureshi, R., Aamir, M., & Magarini, M. (2022, October). Utilizing Cognitive Radio and Reinforcement Learning for Multiple Preamble Detection in 5G-PRACH. In *2022 25th International Symposium on Wireless Personal Multimedia Communications (WPMC)* (pp. 70-74). IEEE.

Zeng, Y., & Liang, Y. C. (2007, September). Maximum-minimum eigenvalue detection for cognitive radio. In *2007 IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications* (pp. 1-5). IEEE.

Zhang, L., Wei, Z., Wang, L., Yuan, X., Wu, H., & Xu, W. (2023). *Spectrum Sharing in the Sky and Space : A Survey*. 1–33.

APPENDIX

APPENDIX A

SAMPLE OF POWER SPECTRUM DATA

Freq_in_MHz	Sweep 1	O/U	Sweep 2	O/U	Sweep 3	O/U	Sweep 4	O/U	Sweep 5	O/U
8.00E+07	-1.07E+02	1	-1.06E+02	1	-1.08E+02	1	-1.09E+02	1	-1.08E+02	1
8.42E+07	-1.07E+02	1	-1.06E+02	1	-1.05E+02	1	-1.07E+02	1	-1.05E+02	1
8.83E+07	-4.71E+01	1	-5.39E+01	1	-5.22E+01	1	-3.98E+01	1	-4.99E+01	1
9.25E+07	-6.57E+01	1	-7.76E+01	0	-7.17E+01	0	-7.38E+01	0	-7.19E+01	0
9.67E+07	-9.08E+01	0	-8.59E+01	0	-8.85E+01	0	-8.85E+01	0	-8.87E+01	0
1.01E+08	-8.89E+01	0	-9.13E+01	0	-8.78E+01	0	-8.78E+01	0	-8.77E+01	0
1.05E+08	-9.81E+01	0	-9.86E+01	0	-9.75E+01	0	-9.75E+01	0	-9.68E+01	0
1.09E+08	-8.73E+01	0	-9.25E+01	0	-9.33E+01	0	-9.33E+01	0	-9.39E+01	0
1.13E+08	-1.16E+02	1	-1.16E+02	1	-1.19E+02	1	-1.14E+02	1	-1.17E+02	1
1.18E+08	-1.12E+02	1	-1.10E+02	1	-1.14E+02	1	-1.13E+02	1	-1.13E+02	1
1.22E+08	-9.85E+01	0	-1.05E+02	1	-9.93E+01	0	-1.04E+02	1	-1.01E+02	1
1.26E+08	-1.15E+02	1	-1.08E+02	1	-1.05E+02	1	-1.12E+02	1	-1.10E+02	1
1.30E+08	-1.04E+02	1	-9.74E+01	0	-1.10E+02	1	-1.04E+02	1	-1.03E+02	1
1.34E+08	-1.07E+02	1	-1.05E+02	1	-1.07E+02	1	-1.05E+02	1	-1.11E+02	1
1.38E+08	-1.16E+02	1	-1.15E+02	1	-1.16E+02	1	-1.17E+02	1	-1.17E+02	1
1.43E+08	-1.13E+02	1	-1.13E+02	1	-1.12E+02	1	-1.14E+02	1	-1.13E+02	1
1.47E+08	-1.10E+02	1	-1.12E+02	1	-1.11E+02	1	-1.14E+02	1	-1.12E+02	1
1.51E+08	-9.68E+01	0	-9.39E+01	0	-9.46E+01	0	-1.08E+02	1	-1.15E+02	1
1.55E+08	-1.12E+02	1	-1.12E+02	1	-1.16E+02	1	-1.18E+02	1	-1.19E+02	1
1.59E+08	-1.12E+02	1	-1.14E+02	1	-1.14E+02	1	-1.14E+02	1	-1.16E+02	1
1.63E+08	-1.20E+02	1	-1.17E+02	1	-1.17E+02	1	-1.17E+02	1	-1.19E+02	1
1.68E+08	-1.15E+02	1	-1.12E+02	1	-1.12E+02	1	-1.07E+02	1	-1.09E+02	1
1.72E+08	-1.18E+02	1	-1.17E+02	1	-1.12E+02	1	-1.11E+02	1	-1.15E+02	1
1.76E+08	-1.02E+02	1	-1.03E+02	1	-1.05E+02	1	-9.78E+01	0	-1.01E+02	1
1.80E+08	-9.87E+01	0	-9.83E+01	0	-1.01E+02	1	-9.92E+01	0	-9.98E+01	0
1.84E+08	-1.16E+02	1	-1.10E+02	1	-1.17E+02	1	-1.16E+02	1	-1.15E+02	1
1.89E+08	-1.15E+02	1	-1.14E+02	1	-1.16E+02	1	-1.19E+02	1	-1.17E+02	1

Freq_in_MHz	Sweep 1	O/U	Sweep 2	O/U	Sweep 3	O/U	Sweep 4	O/U	Sweep 5	O/U
1.93E+08	-1.00E+02	0	-1.00E+02	0	-1.02E+02	1	-1.03E+02	1	-1.03E+02	1
1.97E+08	-1.17E+02	1	-1.15E+02	1	-1.14E+02	1	-1.19E+02	1	-1.18E+02	1
2.01E+08	-1.01E+02	1	-9.49E+01	0	-9.66E+01	0	-9.64E+01	0	-1.00E+02	1
2.05E+08	-1.06E+02	1	-1.05E+02	1	-1.05E+02	1	-1.02E+02	1	-1.02E+02	1
2.09E+08	-1.14E+02	1	-1.12E+02	1	-1.15E+02	1	-1.14E+02	1	-1.09E+02	1
2.14E+08	-1.14E+02	1	-1.13E+02	1	-1.15E+02	1	-1.10E+02	1	-1.14E+02	1
2.18E+08	-1.10E+02	1	-1.11E+02	1	-1.08E+02	1	-1.10E+02	1	-1.10E+02	1
2.22E+08	-1.19E+02	1	-1.17E+02	1	-1.12E+02	1	-1.14E+02	1	-1.16E+02	1
2.26E+08	-1.12E+02	1	-1.13E+02	1	-1.10E+02	1	-1.15E+02	1	-1.14E+02	1
2.30E+08	-1.18E+02	1	-1.19E+02	1	-1.19E+02	1	-1.17E+02	1	-1.19E+02	1
2.34E+08	-1.18E+02	1	-1.19E+02	1	-1.18E+02	1	-1.19E+02	1	-1.19E+02	1