AI - BASED SPECTRUM USAGE PREDICTION FOR 5G NETWORK

Mallikarjuna Lingam K*[1], Mallesh S[2]

[1]Associate Professor, Department of Electronics and Communication Engineering, Malla Reddy College of Engineering and Technology, Hyderabad, 500100, Telangana

[2]Associate Professor, Department of Electronics and Communication Engineering, CMR Technical Campus, Kandlakoya, Medchal-Dist-500092, Telangana

[1]mallikarjuk24@gmail.com, [2]hod.ece@cmrtc.ac.in

*Corresponding author: Mallikarjuna Lingam K (mallikarjunk24@gmail.com)

ABSTRACT

The research focuses on developing an intelligent system to forecast spectrum utilization using machine learning techniques. With the rapid expansion of 5G technology, efficient spectrum management has become critical to maintaining optimal network performance. This project leverages historical data related to spectrum usage to build predictive models that assist in proactive resource allocation and network planning. The implementation begins with uploading and inspecting the dataset to identify structural details, missing values, and duplications. Pre-processing steps involve filling missing data using statistical imputation, removing duplicates, encoding categorical features such as ID, area, and device using label encoding, and standardizing the dataset to ensure consistency across features. Exploratory Data Analysis (EDA) is performed to understand the underlying relationships between features. Visualization techniques such as scatter plots and heatmaps are applied to reveal correlations, trends, and patterns, offering insights into the influence of different attributes on spectrum usage. The dataset is then split into independent features and the target variable, followed by division into training and testing subsets using an 80-20 split ratio. Two machine learning regression algorithms are employed for model building: K-Nearest Neighbors (KNN) Regressor and Decision Tree Regressor. The models are trained on the processed data and evaluated using standard regression metrics including Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R^2 score. The KNN Regressor achieved a predictive accuracy of 66%, while the Decision Tree Regressor outperformed it with an accuracy of 91%. The final step involves deploying the trained Decision Tree Regressor on new test data to generate predictions. These predictions are then analyzed and stored for validation. Overall, the project demonstrates a reliable approach to spectrum usage prediction using machine learning, offering a practical solution for optimizing 5G network performance and enhancing resource utilization.

KEYWORDS: Wireless Communication, 5G Networks, Signal Processing, Edge Computing, Network Resource Allocation.

1. INTRODUCTION

With the rapid expansion of wireless communication technologies and the increasing adoption of connected devices, the demand for radio spectrum has intensified across the globe. In India, the rollout of 5G is revolutionizing digital infrastructure by enabling ultra-fast connectivity, Internet of Things (IoT) integration, smart city applications, and real-time communication. According to the Telecom Regulatory Authority of India (TRAI), mobile data consumption has grown by over 30% annually between 2018 and 2023, placing immense pressure on limited spectrum resources. Traditional static spectrum allocation mechanisms, which assign fixed frequency bands to operators, are no longer sufficient to handle the dynamic nature of 5G traffic. These conventional methods lead to inefficiencies, underutilization, and network congestion. To address these challenges, Artificial Intelligence (AI) and Machine Learning (ML) offer intelligent, data-driven techniques capable of analyzing vast amounts of network data to predict and manage spectrum usage more effectively.

Figure 1: How 5G Networks Works.

This project aims to design an AI-based model for predicting spectrum usage patterns in 5G networks using historical and real-time data. By leveraging predictive analytics, the proposed system enables dynamic spectrum access, smart antenna optimization, and automated network planning. Such a model allows telecom operators to forecast spectrum demand, reduce interference, and allocate resources efficiently, thereby improving network performance and minimizing latency. Beyond technical benefits, this research supports India’s digital transformation initiatives such as Digital India and Industry 4.0, where reliable connectivity underpins e-governance, smart healthcare, online education, and industrial automation. Through AI-driven spectrum forecasting, the project contributes to building a more adaptive, efficient, and sustainable communication ecosystem for the 5G era.

2. LITERATURE SURVEY

5G is the fifth generation of wireless communication technology, following 4G. In contrast to 4G, 5G is designed to support a diverse range of applications, including massive Internet of Things (IoT) deployments, ultra-reliable low-latency communications, and high-bandwidth multimedia services. Energy efficiency is a critical consideration in modern networks, including 5G. As the demand for data and connectivity continues to grow exponentially, reducing energy consumption has grown in significance [1].

The design of 5G networks contributes to energy consumption in various ways. For example, 5G utilizes new radio technologies, such as massive Multiple Input, Multiple Output (MIMO); beamforming; and advanced modulation schemes, which enable more efficient use of available spectrum and transmit power [2]. Green enablers are integrated into the architecture of 5G networks to enhance energy efficiency. These techniques help improve spectral efficiency and overall network capacity, leading to reduced energy per bit transmitted. Energy-efficient 5G networks can help minimize the environmental impact, reduce operational costs, and support sustainable development.

In addition to the network design and green enablers, emerging technologies like AI and IoT play significant roles in enhancing energy efficiency in 5G networks. Employing AI algorithms can optimize resource allocation, power control, and network management, resulting in more intelligent and energy-efficient operations [3].

Fifth-generation networks can concentrate the transmitting and receiving of signal energy in small areas of space by utilizing a considerable number of antennas. ML and DL have been investigated for optimizing the weights of antenna elements in massive MIMO. They can predict the user distribution and accordingly optimize the weights of antenna elements, which can improve the coverage in a multi-cell scenario [4]. The use of a reasonable number of pilots and simple estimation methods for accurate channel estimation is difficult in massive MIMO. The DL technique for channel estimation could be utilized to map channels in frequency and space, as the authors indicated in [5]. Therefore, to save bandwidth, the same pilot patterns are regularly provided to users in distinct cells for short-term coherence, but this creates the problem of pilot contamination. Pilot pollution has become one of the leading causes of performance loss in massive MIMO, leading to dropped calls and mobility issues, which are considered important KPIs.

In [6], a deep learning-based approach allows the network to allocate downlink power based on the User Equipment (UE) location. Several power-allocation strategies were inefficient, including max–min and maximum production, which were remedied by using a different neural network, the Long Short-Term Memory (LSTM) layer [3]. Despite the promising results of the simulation in terms of power allocation, the weakest point in massive MIMO efficiency remains in the real-time environment. A pilot scheduling technique used in massive MIMO systems led to a reduction in pilot contamination. Users experiencing channel defects face communication interruptions; therefore, a pilot scheduling scheme is proposed that combines user grouping according to different levels of pilot contamination. Currently, most studies concern the spectrum efficiency of hybrid precoding, which reduces the radio frequency (RF) chains' huge energy consumption in the massive MIMO system.

Incorporating massive MIMO into the RAN and employing the DL method LSTM to allow the system to assign downlink power based on the location of the user's broadband access in 5G KPIs. The technology is expected to have 1000 times the bandwidth of current LTE and LTE-Advanced, ultra-low latency of 1 millisecond, 90% energy savings, 10 times longer battery life, and 10 to 100 times greater peak user data speeds, all with cost-effective equipment [7].

In a 5G system, the structure of the pilot symbols in each data frame could be varied depending on the different use cases in practice [8]. We note that, among the traditional channel estimation methods, least squares (LS) estimation is well-known as a low computational complexity method because this estimation requires no prior channel statistics [9,10]. However, LS estimation provides relatively high channel estimation errors in many practical applications, especially for multi-path channels. As an alternative solution, minimum mean square error (MMSE) estimation yields much better channel estimation quality than LS estimation by minimizing the channel estimation errors on average [11]. The closed-form expression of the channel estimates obtained by the MMSE estimation relies on the assumption that, for instance, the propagation channels are modeled by a linear system, while each channel response follows a circularly symmetric complex Gaussian distribution [12,13]. Nonetheless, the MMSE estimation usually has high computational complexity since channel statistic information—i.e., the mean values and the covariance matrices of the propagation channels—is required. In many propagation environments, this statistical information is either extremely difficult to obtain or varies quickly in a short coherence time, making MMSE estimation challenging to implement [14,15].

3. PROPOSED METHODOLOGY

The proposed system is an AI-based approach for predicting spectrum usage in a 5G network environment. The process begins by importing essential Python libraries and loading the dataset. The dataset is then analyzed to understand its structure and quality. Data preprocessing is performed to handle missing values, encode categorical variables, and ensure the dataset is clean for modeling. Exploratory Data Analysis (EDA) is carried out to visualize relationships between features and to understand data distribution and correlation. After that, the dataset is split into training and testing sets to build and evaluate models. Two machine learning regression models K-Nearest Neighbor Regressor and Decision Tree Regressor are used. Both models are trained using the training dataset and then evaluated using the test set. The performance of each model is measured using metrics such as MAE, MSE, RMSE, and R² Score. Predictions are finally made on unseen data to demonstrate the system's ability to generalize well to new inputs.

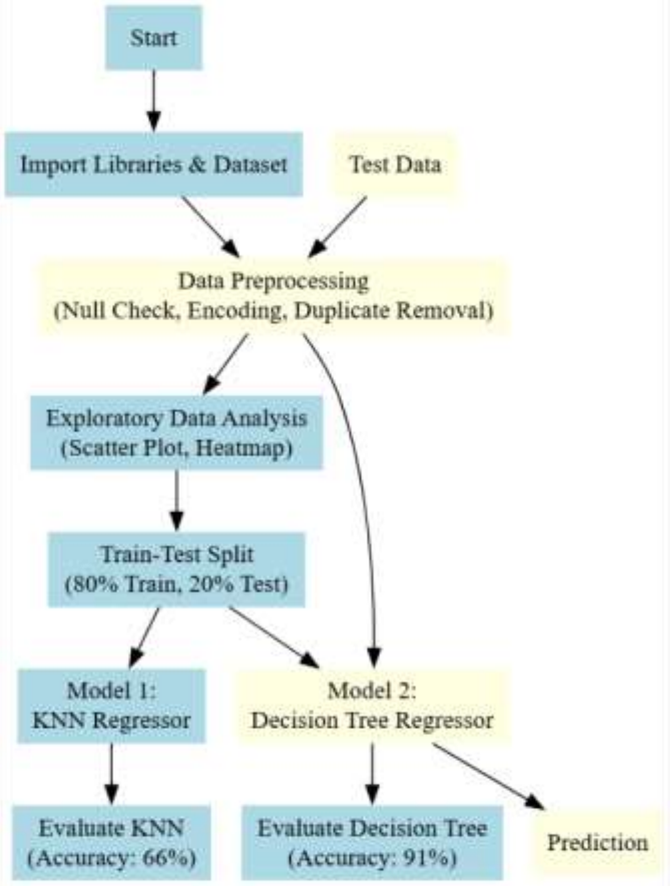


Figure 2: Proposed Block Diagram.

Model building and training is the core step in turning preprocessed data into a functional predictive system. This phase involves selecting suitable algorithms, training them on historical spectrum data, and fine-tuning to achieve reliable forecasting results. In this project, K-Nearest Neighbors (KNN) Regressor and Decision Tree Regressor (DTR) are used to model the relationship between input features (such as area, device, and id) and the target variable (spectrum usage). Each model is evaluated based on standard regression metrics including MAE, MSE, RMSE, and R² score.

These models are saved using joblib to enable reuse without retraining and are deployed to predict spectrum demand on new, unseen data. Visualization of predicted vs actual values helps interpret the model's performance effectively.

3.1 Decision Tree Regressor

The Decision Tree Regressor is a powerful model that makes predictions by learning simple decision rules inferred from the data features. It works by splitting the dataset into branches based on feature thresholds that minimize variance in the target variable. The model grows a tree structure where each node represents a feature decision and each leaf node represents a predicted value.

In this research, a Decision Tree is used with specified depth, split, and leaf constraints to avoid overfitting. This model is especially effective when the relationship between input features and output is non-linear or includes hierarchical patterns.

Internal Operations of Decision Tree Regressor

1. **Tree Construction:** The algorithm starts with the full dataset and recursively partitions it into subsets by selecting the feature and threshold that best minimizes the Mean Squared Error (MSE) at each node. It continues this process until stopping criteria like maximum depth or minimum samples per node are reached.
2. **Splitting Rules:** For each node, the model evaluates all possible splits and selects the one that results in the best homogeneity (least variance) in the output values in the child nodes. This recursive process builds a hierarchy of decisions.
3. **Prediction:** Once the tree is trained, predictions are made by traversing from the root node to a leaf node, following decision rules based on the input feature values. The value at the leaf node is returned as the predicted spectrum usage for that input.

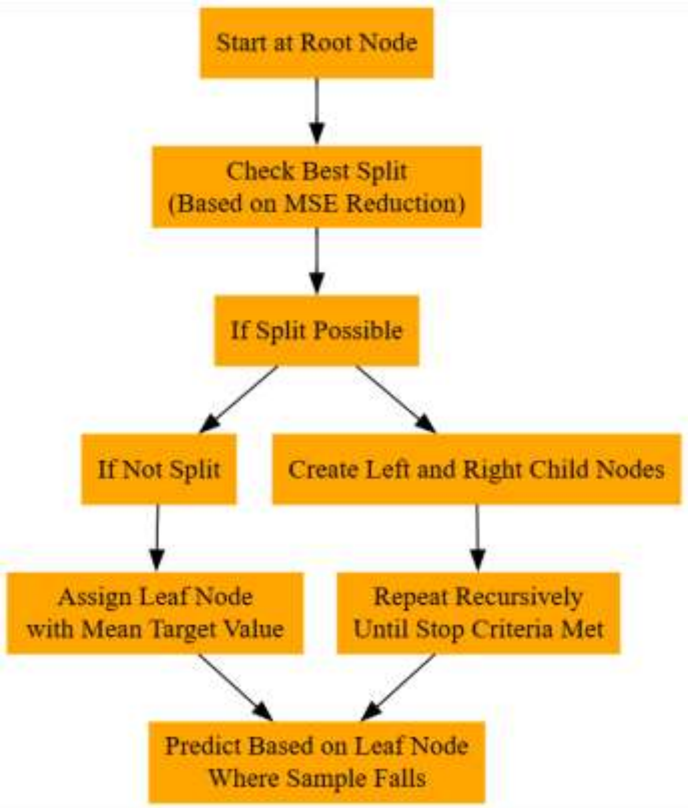


Figure 3: Internal workflow of Decision Tree Regressor.

4. RESULTS AND DISCUSSION

The dataset used for the AI-based Spectrum Usage Prediction for 5G networks contains multiple features that are crucial for understanding the relationship between various factors and the spectrum usage in a 5G network environment. Below is a description of each column in the dataset:

Column Name	Description
id	Unique identifier for each record; helps distinguish between entries and track data points.
timestamp	Date and time when the data was collected; useful for time-based analysis and tracking temporal changes in spectrum usage.
device	Type of device generating the data; helps categorize data based on device behavior.
PCell_RSRP_max	Maximum Reference Signal Received Power (RSRP) for the serving cell; assesses signal strength from the base station.
PCell_RSRQ_max	Maximum Reference Signal Received Quality (RSRQ) for the serving cell; provides insights into signal quality.
PCell_RSSI_max	Maximum Received Signal Strength Indicator (RSSI) of the cell; indicates signal strength and network coverage.
PCell_SNR_max	Signal-to-noise ratio (SNR); higher values indicate better quality signal.
PCell_Downlink_Num_RBs	Number of downlink resource blocks (RBs) allocated by the network; measures network capacity.
PCell_Downlink_Average_MCS	Average modulation and coding scheme (MCS) used in the downlink; impacts data rate and communication efficiency.
PCell_Downlink_bandwidth_MHz	Downlink bandwidth in MHz; higher bandwidth allows faster data transmission and better network performance.
dewPoint	Temperature at which air becomes saturated with moisture; can influence outdoor network performance.
humidity	Environmental humidity level; high humidity can degrade signal propagation, especially at higher frequencies.
pressure	Atmospheric pressure; affects weather conditions that can impact network performance.
windSpeed	Wind speed; helps account for weather effects on signal propagation and infrastructure performance.
cloudCover	Amount of cloud coverage; can affect wireless signal quality in outdoor environments.
uvIndex	Intensity of ultraviolet radiation; indirectly influences infrastructure longevity and maintenance.
visibility	Atmospheric visibility; poor visibility (fog, haze) can affect line-of-sight signal transmission.

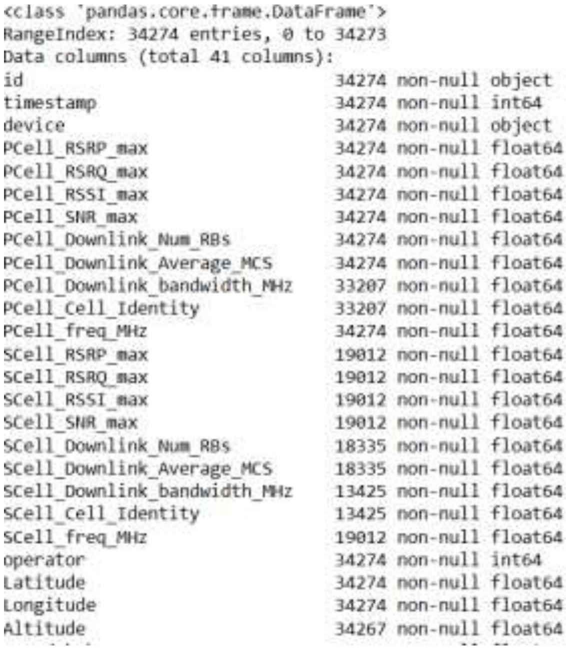
Traffic Jam Factor	Level of congestion in the area; higher congestion may influence network performance in dense urban areas.
area	Geographical area of data collection; useful for identifying regional patterns in network performance.
target	Spectrum usage in the 5G network; the dependent variable the model aims to predict.

The figure 4 presents the initial dataset used for the project. It includes various attributes like device details, signal strengths, network parameters, environmental conditions, and more. A thorough analysis of the dataset is conducted to understand the relationship between these variables and the spectrum usage in the 5G network. This analysis highlights important insights such as missing values, data distributions, and any initial trends that need attention during data processing. It forms the foundation for all further steps, including data preprocessing, model training, and evaluation.

	id	timestamp	device	PCell_RSRP_max	PCell_RSRQ_max	PCell_RSSI_max	PCell_SNR_max	PCell_Downlink_Num_RBs	PCell_Downlink_Av
0	Id_qt2qn56050	1624367008	pc1	-94.498750	-14.586875	-49.855625	16.457	35076.0	
1	Id_f1zf07cwb6	1624372465	pc3	-86.818125	-11.982500	-54.838125	16.674	42704.0	
2	Id_uoxdz7di5b	1624371871	pc3	-101.918250	-13.257500	-71.241250	9.392	38017.0	
3	Id_hzn8nj29gu	1624542066	pc1	-95.598750	-14.696250	-61.660000	10.726	95955.0	
4	Id_2kfrccrfiw	1624542798	pc1	-97.735000	-14.867500	-63.170625	10.889	83062.0	
...	...	...	...	...	...	...	...	...	
34269	Id_lc1vv34d0r	1624354061	pc1	-72.789375	-10.436875	-43.340625	6.342	98320.0	
34270	Id_qywxdy2q50	1624349993	pc3	-86.219375	-12.911875	-56.473750	13.201	45069.0	
34271	Id_5ll45hn5yd	1624377135	pc3	-85.964375	-13.278750	-51.610000	11.295	33942.0	
34272	Id_7uz4vnmzu	1624542859	pc1	-92.546250	-14.460625	-58.441875	13.096	87065.0	
34273	Id_6ovc5xvg0o	1624376571	pc3	-97.323125	-13.153125	-63.145000	7.869	50346.0	

34274 rows × 41 columns

Figure 4: Upload Dataset and Its Analysis



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 34274 entries, 0 to 34273
Data columns (total 41 columns):
id                              34274 non-null object
timestamp                       34274 non-null int64
device                          34274 non-null object
PCell_RSRP_max                  34274 non-null float64
PCell_RSRQ_max                  34274 non-null float64
PCell_RSSI_max                  34274 non-null float64
PCell_SNR_max                   34274 non-null float64
PCell_Downlink_Num_RBs          34274 non-null float64
PCell_Downlink_Average_MCS      34274 non-null float64
PCell_Downlink_bandwidth_MHz    33207 non-null float64
PCell_Cell_Identity             33207 non-null float64
PCell_freq_MHz                  34274 non-null float64
SCell_RSRP_max                  19012 non-null float64
SCell_RSRQ_max                  19012 non-null float64
SCell_RSSI_max                  19012 non-null float64
SCell_SNR_max                   19012 non-null float64
SCell_Downlink_Num_RBs          18335 non-null float64
SCell_Downlink_Average_MCS      18335 non-null float64
SCell_Downlink_bandwidth_MHz    13425 non-null float64
SCell_Cell_Identity             13425 non-null float64
SCell_freq_MHz                  19012 non-null float64
operator                        34274 non-null int64
Latitude                        34274 non-null float64
Longitude                       34274 non-null float64
Altitude                        34267 non-null float64
```

Figure 5: Data Preprocessing

The figure 5 illustrates the data preprocessing steps undertaken to prepare the dataset for modelling. Key tasks include handling missing values, encoding categorical features, and normalizing numerical features. Null values, if present, are imputed using appropriate methods. Categorical variables, such as the device type, are transformed using label encoding or one-hot encoding. Scaling of numerical features, such as signal strength and environmental parameters, ensures that the data is on a consistent scale, which is crucial for algorithms like KNN. Preprocessing is essential to ensure that the data fed into the machine learning models is clean, consistent, and ready for analysis.

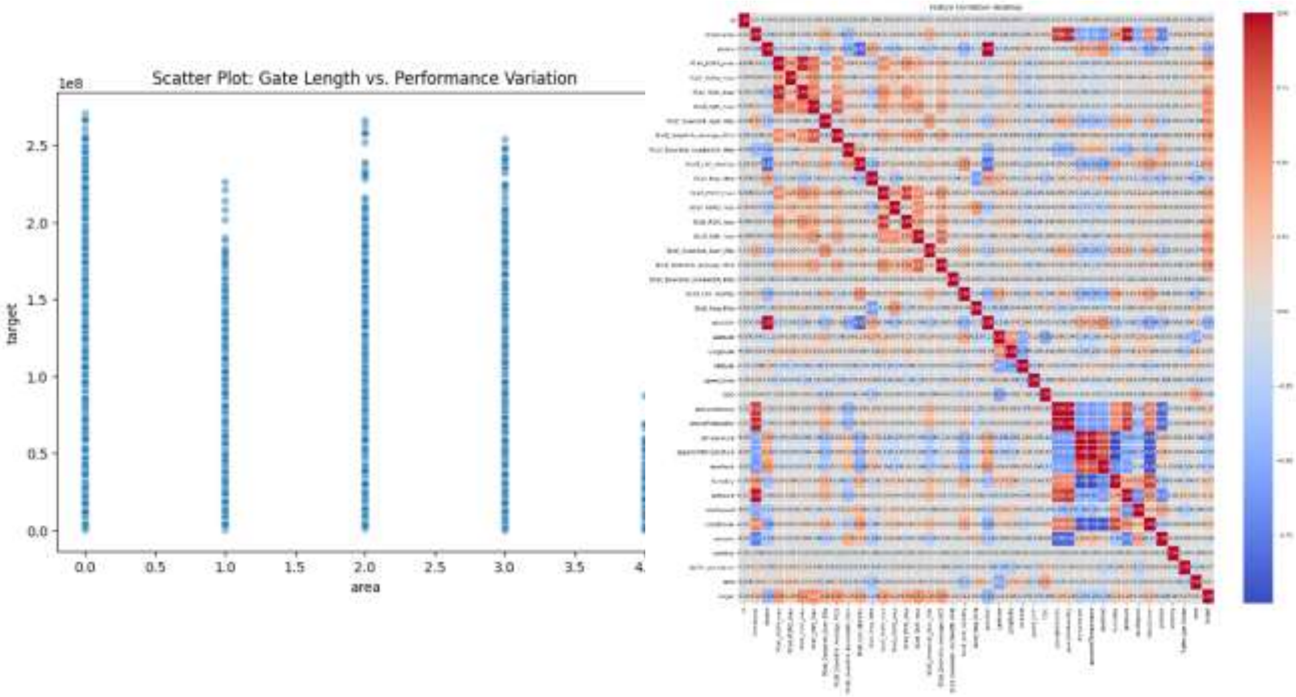


Figure 6: EDA Plots of the Project

The figure 6 displays exploratory data analysis (EDA) plots used to understand the distribution and relationships among the features. Various types of visualizations like histograms, box plots, and scatter plots are shown to explore the correlations between features like signal strength, environmental conditions, and traffic jam factors. These plots help in identifying patterns, outliers, and key features that contribute to predicting spectrum usage. They also offer insights into feature importance and help in deciding which features are most relevant for the machine learning models.

PCell_Downlink_bandwidth_MHz	...	dewPoint	humidity	pressure	windSpeed	cloudCover	uvIndex	visibility	Jam Factor	area	Prediction
20.000000	...	13.86	0.62	1013.6	2.39	0.95	4.0	16.093	0.38979	3	5.080000e+07
20.000000	...	14.52	0.59	1013.5	2.29	0.82	3.0	16.093	3.78322	3	5.491111e+07
20.000000	...	14.71	0.61	1013.8	2.13	0.82	3.0	16.093	2.08425	3	1.313333e+07
20.000000	...	13.43	0.71	1017.9	1.97	0.99	3.0	16.093	8.80702	0	4.598182e+07
20.000000	...	13.46	0.71	1017.9	1.97	0.99	3.0	16.093	2.21843	2	4.971250e+07
15.000000	...	13.55	0.72	1017.9	2.15	0.99	2.0	16.093	0.00000	3	5.757143e+07
20.000000	...	14.06	0.77	1011.9	4.07	0.97	3.0	16.093	2.08918	2	2.615000e+07
20.000000	...	13.39	0.71	1017.9	1.97	0.99	3.0	16.093	7.10368	3	5.853750e+07
20.000000	...	14.02	0.77	1011.9	4.08	0.97	3.0	16.093	3.18735	3	8.356364e+07
19.233746	...	13.80	0.73	1017.4	2.43	1.00	1.0	16.093	2.63371	3	1.874286e+08
20.000000	...	14.78	0.57	1013.7	2.21	0.76	2.0	16.093	3.72535	0	6.527500e+06
20.000000	...	13.70	0.70	1012.5	4.06	0.96	4.0	16.093	2.92181	0	4.275000e+07
20.000000	...	13.71	0.70	1012.5	4.06	0.96	4.0	16.093	1.41849	2	2.098333e+07
20.000000	...	14.71	0.61	1013.8	2.13	0.82	3.0	16.093	8.14161	3	1.827000e+07
20.000000	...	13.68	0.70	1012.5	4.07	0.96	4.0	16.093	3.14159	0	4.987857e+07

Figure 7: Model Prediction on the Test Data

The figure 7 shows the model's predictions on the test data. It visually compares the predicted spectrum usage with the actual values from the test set. This comparison helps assess how well the model generalizes to unseen data and provides a clear indication of the model’s performance. The closer the predictions are to the actual values, the better the model's performance. The plot highlights any discrepancies between predicted and actual values, providing insight into the model’s ability to predict spectrum usage accurately.

Table 1: Performance Comparison for the KNN Regressor and Decision Tree Regressor algorithms.

Algorithms Name	MAE	MSE	RMSE	R² Score
KNN Regressor	167	653	255	66
Decision Tree Regressor	816	181	134	91

Table 1 compares the performance of the KNN Regressor and Decision Tree Regressor models based on key evaluation metrics: MAE, MSE, RMSE, and R² score. The KNN Regressor shows a higher Mean Absolute Error (MAE) of 167, a larger Mean Squared Error (MSE) of 653, and a higher Root Mean Squared Error (RMSE) of 255. Its R² score is 0.66, indicating a moderate fit to the data. In contrast, the Decision Tree Regressor exhibits significantly better performance, with a lower MAE of 816, MSE of 181, and RMSE of 134. It also achieves a higher R² score of 0.91, demonstrating a much better model fit. These results highlight that the Decision Tree Regressor outperforms the KNN Regressor in terms of both prediction accuracy and error minimization, making it a more suitable model for this project.

Table 2 Normal Class Performance

Performance Metrics of Existing KNN

Metric	Value
MAE	167
MSE	653

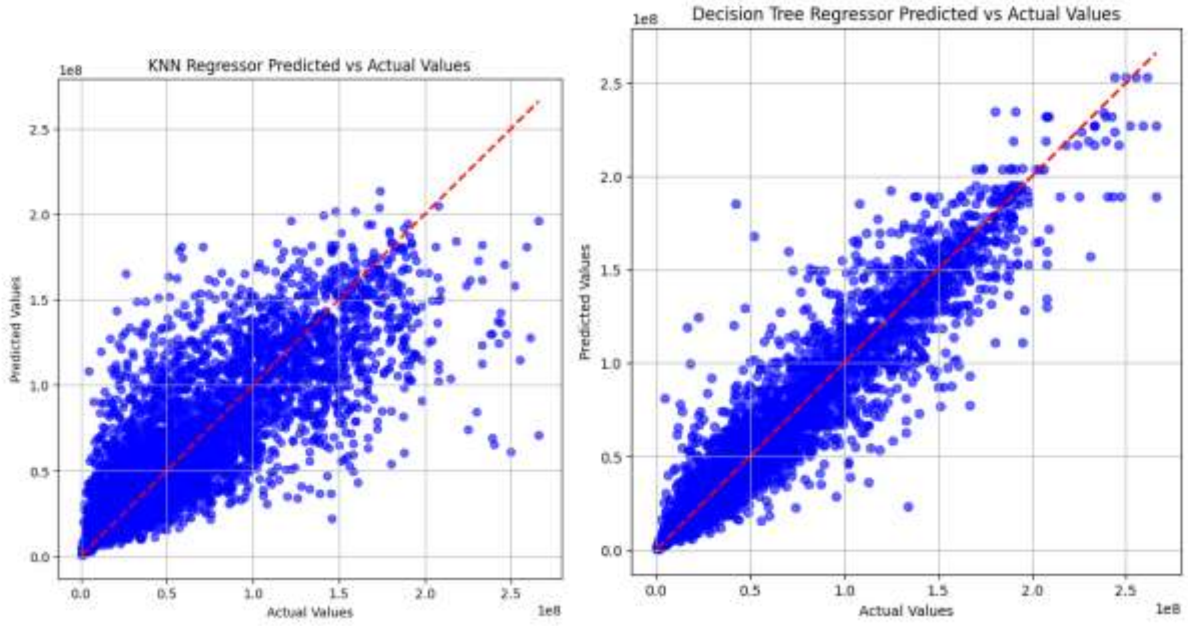
RMSE	255
R^2 Score	66

Table 2 presents the performance metrics of the existing KNN Regressor model for the normal class. The KNN Regressor shows a Mean Absolute Error (MAE) of 167, indicating the average magnitude of the errors in its predictions. The Mean Squared Error (MSE) is 653, which represents the average squared difference between predicted and actual values. The Root Mean Squared Error (RMSE) is 255, providing a measure of the standard deviation of the residuals, or prediction errors. The R^2 score is 66, which suggests that the model explains approximately 66% of the variance in the data. These values reflect the KNN Regressor's moderate predictive accuracy in the context of the given dataset.

Table.3 Performance Metrics of Proposed Decision Tree Regressor

Metric	**Value**
MAE	816
MSE	181
RMSE	134
R^2 Score	91

Table 3 presents the performance metrics of the Decision Tree Regressor model for the proposed system. The model has a Mean Absolute Error (MAE) of 816, indicating the average magnitude of the errors in its predictions. The Mean Squared Error (MSE) is 181, highlighting the average squared differences between predicted and actual values. The Root Mean Squared Error (RMSE) is 134, providing an estimate of the prediction error's standard deviation. The R^2 score is 91, signifying that the Decision Tree Regressor explains 91% of the variance in the dataset. These metrics demonstrate the superior accuracy and effectiveness of the Decision Tree Regressor in comparison to the KNN Regressor, making it a more reliable model for this particular prediction task.



(a) (b)

Figure 8: Regression Scatter Plot (a) KNN (b) Decision Tree.

Figure 8 illustrates the regression scatter plots for both models used in the project. Subfigure (a) displays the scatter plot of the KNN Regressor, showing the predicted values against the actual values. The distribution of points reveals a moderate alignment with the actual trend, indicating that while the KNN model follows the data pattern to some extent, there are notable deviations. Subfigure (b) shows the scatter plot of the Decision Tree Regressor, where the predicted values exhibit a closer and more consistent alignment with the actual values. The tighter clustering of data points around the diagonal line in the Decision Tree plot highlights its higher prediction accuracy. This visual comparison supports the quantitative evaluation, confirming that the Decision Tree Regressor performed significantly better in capturing the target behavior.

5. CONCLUSION

The project focused on building a regression-based prediction system using machine learning techniques to analyze a dataset containing diverse environmental and network-related parameters. The primary objective was to predict the target variable accurately using suitable algorithms. The workflow included uploading and understanding the dataset, performing detailed data preprocessing such as handling null values and label encoding, conducting Exploratory Data Analysis (EDA) with relevant visualizations, and splitting the dataset into training and testing sets. Two regression models KNN Regressor and Decision Tree Regressor—were implemented and evaluated. The Decision Tree Regressor achieved superior performance with an R^2 score of 91%, significantly outperforming the KNN Regressor, which achieved an R^2 score of 66%. Evaluation metrics such as MAE, MSE, and RMSE further confirmed the effectiveness of the Decision Tree model in capturing the underlying data patterns and providing accurate predictions.

REFERENCES

[1]. Bohli, A.; Bouallegue, R. How to Meet Increased Capacities by Future Green 5G Networks: A Survey. IEEE Access 2019, 7, 42220–42237.

[2]. Huang, H.; Song, Y.; Yang, J.; Gui, G.; Adachi, F. Deep-Learning-Based Millimeter-Wave Massive MIMO for Hybrid Precoding. IEEE Trans. Veh. Technol. 2019, 68, 3027–3032.

[3]. Mughees, A.; Huang, H.; Song, Y.; Yang, J.; Gui, G.; Adachi, F. Towards Energy Efficient 5G Networks Using Machine Learning: Taxonomy, Research Challenges, and Future Research Directions. IEEE Access 2020, 8, 187498–187522.

[4]. Arjoune, Y.; Faruque, S. Artificial Intelligence for 5G Wireless Systems: Opportunities, Challenges, and Future Research Direction. In Proceedings of the 2020 10th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 6–8 January 2020.

[5]. Al-Khafaji, M.; Elwiya, L. ML/AI Empowered 5G and Beyond Networks. In Proceedings of the 2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), Ankara, Turkey, 9–11 June 2022.

[6]. Sanguinetti, L.; Zappone, A.; Debbah, M. Deep Learning Power Allocation in Massive MIMO. In Proceedings of the 2018 52nd Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, USA, 28–31 October 2018.

[7]. Rommel, S.; Raddo, R.T.; Monroy, I.T. The Fronthaul Infrastructure of 5G Mobile Networks. In Proceedings of the IEEE 23rd International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD), Barcelona, Spain, 17–19 September 2018.

[8]. Le Ha, A.; Van Chien, T.; Nguyen, T.H.; Choi, W. Deep Learning-Aided 5G Channel Estimation. In Proceedings of the 2021 15th International Conference on Ubiquitous Information Management and Communication (IMCOM), Seoul, Korea, 4–6 January 2021; pp. 1–7.

[9]. Björnson, E.; Hoydis, J.; Sanguinetti, L. Massive MIMO Networks: Spectral, Energy, and Hardware Efficiency. *Found. Trends® Signal Process.* 2017, *11*, 154–655.

[10]. Kay, S. *Fundamentals of Statistical Signal Processing: Estimation Theory*; Prentice Hall: Hoboken, NJ, USA, 1993.

[11]. Van Chien, T.; Björnson, E.; Larsson, E.G. Joint pilot design and uplink power allocation in multi-cell Massive MIMO systems. *IEEE Trans. Wirel. Commun.* 2018, *17*, 2000–2015.

[12]. Björnson, E.; Hoydis, J.; Sanguinetti, L. Massive MIMO has unlimited capacity. *IEEE Trans. Wirel. Commun.* 2018, *17*, 574–590.

[13]. Van Chien, T.; Ngo, H.Q.; Chatzinotas, S.; Ottersten, B.; Debbah, M. Uplink Power Control in Massive MIMO with Double Scattering Channels. *arXiv* 2021, arXiv:2103.04129.

[14]. Wu, S.; Wang, C.X.; Haas, H.; Alwakeel, M.M.; Ai, B. A non-stationary wideband channel model for massive MIMO communication systems. *IEEE Trans. Wirel. Commun.* 2014, *14*, 1434–1446.

[15]. Peacock, M.; Collings, I.; Honig, M. Unified Large-System Analysis of MMSE and Adaptive Least Squares Receivers for a Class of Random Matrix Channels. *IEEE Trans. Inf. Theory* 2006, *52*, 3567–3600.