Machine Learning-Based Payload Allocation in a Service Provided Area to Maximize the Efficiency of the Network

B. Suresh(✉) and K. L. V. Sai Prakash Sakuru

Department of Electronics and Communication Engineering, National Institute of Technology, Warangal-506004, Telangana, India
bsec21330@student.nitw.ac.in

Abstract. Resource block allocation in cellular networks plays a vital role in defining the efficient use of the spectrum and maximizing user density. The resource blocks, also referred to as "payload," is a wagon carrying the actual user data, and this study uses machine learning to forecast the needed payload for cellular consumers. They are payload as a target feature from the simulated large dataset with different frequency bands of arbitrary service provider cell sites. XGBOOST Regression ML model is used to optimize the payload allocation to various cell sites. The complete design is implemented in Google Colaboratory (Colab). It is an open-source cloud platform.

Keywords: XGBOOST · Machine Learning · Colab · Payload · Resource block · Allocation

1 Introduction

Experiencing the rise in economic growth and information and communication technology, the way the communication systems are evolving to handle the voluminous data is the research's outcome. Providing high-quality services by the service providers is a Hercules task. On the other hand, the increase of mobile users and providing them with resource blocks, thereby increasing the throughput efficiency of the network, is making the networks engineers night mere. Based on the available resource blocks, they can decide when to proceed with the call. The channels are allocated so that the interference level is minimal, as shown in Fig. 1. A single base station is expected to serve several hundred IoT-type devices and thousands of mobile devices. Hundreds of network parameters must be configured to optimize network parameters to accommodate the dense traffic. This drives the mobile networks to self-optimization from the enormous data collected from the network and processing it.

Mobile network usage patterns trends, such as higher traffic density in downlink compared to uplink changing rapidly. As billions of IoT devices exchange small junk of data on uplink and downlink networks, the LTE-A performance degrades. The network utilization on both links will increase significantly, and allocating small payload

B. Raj et al. (Eds.): ICETE 2023, AER 223, pp. 494–504, 2023.
https://doi.org/10.2991/978-94-6463-252-1_52

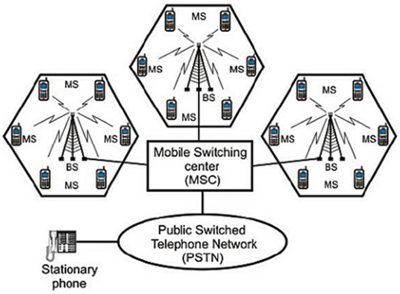


Fig. 1. Basic Cellular System

resources to the vast number of densely spread IoT devices is a Hercules task. The allocation of small payloads efficiently is the prime area of research by many designers to pack multiple data from various devices into a single sub-frame by handling collisions intelligently. By means of allocating large buffers, the transmission overhead can be managed effectively and send the frames orderly. Alternatively, clustering the nearby devices and integrating their payload into a single structure with less overhead can also reduce the complexity of the allocation problem.

Channel allocation is traditionally classified as a static and dynamic problem. The static channel allocation procedure allocates the predefined channels to each cell site based on the channel interference technique. In a non-uniform user distribution among the cell sites, the network utilization is inefficient and results in revenue loss. Signal interference to noise ratio-based channel allocation, referred to as dynamic channel allocation schemes, provides relatively good network utilization efficiency. Hybrid Channel Allocation (HCA) schemes are developed based on the advantages of both static and dynamic channel allocation. When a user makes a call, the fixed set of channels allocated is first utilized. If the fixed sets are completely utilized, then dynamic sets are used.

As the user data has undergone tremendous changes based on different applications, social networks, IoT device data, and machine-to-machine communication, resource allocation under variable packet sizes and channel conditions using static, dynamic, or hybrid allocation schemes are no more efficient. Mobile networks integrated with ML algorithms help to moment-by-moment traffic management without more human interactions and make the network more reliable without congestion.

This paper is organized as follows: Sect. 2 briefly describes the related work of network traffic congestion and channel assignment approaches with ML. Section 3 explained about design model with the regression method. Section 4 is the complete implementation with results and discussion, and Sect. 5 is the conclusion.

2 Related Work

Every user in a cellular network is provided with a resource block to have fruitful communication. As the number of resource blocks is limited, much work is carried out to allocate among the users to meet user demand efficiently. In the case of OFDM, each resource block has 12 sub-carriers with 6/7 OFDM symbols. That means each resource block of a total of 72/84 resource elements that can support variable bandwidths such as 1.4MHz, 3MHz, 5MHz, 10MHz, 15MHz, and 20MHz. Broadly payload allocation strategies [1] are classified as Fixed Channel (FCA), Dynamic Channel (DCA), Borrowing Channel (BCA), and Hybrid Channel (HCA). Mobile traffic is constantly changing; the DCA is the most effective strategy [2].

Resource allocation can be expressed mathematically as a relation to the influence of channels on input signals and can enhance the performance of various operations between the transmitter and receiver. The channel assignment problem is solved in this letter [3] using a convex optimization-based approach and machine learning techniques. Due to the higher cost of acquiring new customers than keeping existing ones, client churn is challenging for telecom businesses. This research [4] forecasts client turnover based on machine learning models.

Machine learning (ML) can solve complex problems without explicit programming [5]. The use of ML in wireless communication achieves a marginal increase in terms of increased user connectivity, enhanced data rates, and coverage. Real-time traffic classifiers must overcome the problems of the actual world. Researchers [6] put forth an innovative approach in which ML classifiers are trained using statistical characteristics computed across several brief sub-flows derived from entire flows produced by the target application.

Automatic feature learning is a capability of deep learning, and researchers have attempted to use it and found improved accuracy. Researchers [7] described popular deep-learning (DL) techniques and how they are used to traffic categorization challenges. DL, a subset of artificial intelligence (AI), is used to optimize the parameters under study by continuously training and learning as proposed in [8]. Reviews of the modern research trends in DL as intelligent communication is applied to different frameworks proposed in 5G technologies to cater to low latency and more services. They include wireless optical communication, Cognitive Radio, Channel estimation, Edge/Fog Computing, and end-to-end encoder/Decoder.

The volume of traffic data using big data technologies and machine learning techniques is analyzed in [9]. Designers look into ML-based classification systems based on a statistically independent payload with random variables such as arrival time and packet length [10]. ML-based I-FOREST and LOF models [11] were considered in the study to find anomalies in large datasets to detect smoke and gas leaks in gated communities.

The fifth generation of mobile communication accommodates three application scenarios: eMBB, uRLLC, and nmMTC. Wireless communication [12] systems must continue evolving with artificial intelligence (AI) development to satisfy the criteria. Machine learning is intended to optimize wireless networks by solving challenging tasks. The IoT Internet of Things is made up of equipment and gadgets with a variety of sensors connected via wireless networks. Proposed [13] Machine learning (ML) can be used to design, develop and optimize wireless channel encoders.

This paper proposes the XGBOOST regression machine learning model to optimize resource block allocation of payload in cellular networks. Colab was used to simulate the environment, and allocation was done with an accuracy of 99.8467%.

3 Design Model

The Design model is represented in Fig. 2, and the functions of various sub-blocks are explained below:

Dataset: Various data samples make up a dataset. This collection is often displayed in a tabular format [14]. Each column details a distinct aspect. And each row represents a certain data set participant. The values of each feature, such as the cell id, channel bandwidth, cell utilization, average throughput, payload, or random number values, are used as described in data sets.

Data pre-processing: The step involves observing the entire data and removing any null data if present and converting object data types into int or float form, and reshaping the data to further processing.

Parametric Separating: The variables that are fed into the machine learning models are referred to as features and represent every column in the dataset, and their selection has an effect on the final outcome.

Train-Test-Split: Data is divided into two parts: training data and testing data. More percentage of data is provided for understanding the data, and processing is called training data. The remaining data is used for testing, and the accuracy & predictions of the final outcome are identified from the testing data.

XG BOOST Regression: The mean learning model XGBoost, which entails training and merging many models to provide a single prediction, is used in this study. The payload and a utilization term are both objective function components.

Data Prediction: In ML, "prediction" refers to the outcome of an algorithm used to anticipate the likelihood of a specific result since the system is trained on the history dataset and applied to a new dataset.

Accuracy score: The reliable predictions of accuracy is a fundamental outcome and are mathematically modeled by dividing the total number of predictions by the number of predictions that were accurate.

$$\text{Accuracy} = (\text{Number of correct predictions})/(\text{Total number of predictions}) \tag{1}$$

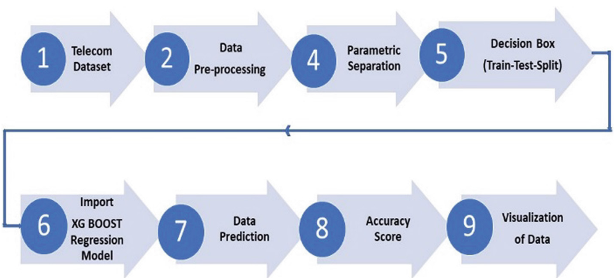


Fig. 2. Design model block diagram

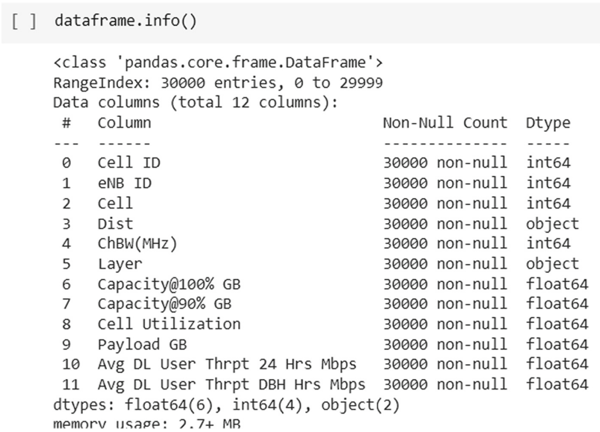
4 Results & Discussion

The dataset considered for evaluating the performance has features such as cell ID, eNB-ID, channel bandwidth, Layers, Cell utilization, Capacity, and Payload. Figure 3 shows that each feature is characterized by this work. The channel bandwidths considered are 5, 10, 15, and 20 MHz, with various layers of L900, L1800, L2100, and mMIMO. A total of 12 features, as shown in Fig. 3, was considered, and the data size consists of 30,000 instances.

Seaborn Library was utilized for exploring & understanding the data. It gives a graphical representation of the selected features. It examined that a more significant number of cells contains 15MHz channel Bandwidth out of 5,10,15,20 MHz, as presented in Fig. 4.

The distribution plots for analyzing the payload and the cell utilization are obtained by contrasting the observed data, and the expected results are shown in Figs. 5 and 6.

The aim is to optimize the channel's payload by considering the target 'Payload in GB.' It is observed that the payload is linearly proportional to channel utilization, as shown in Fig. 7. It also observed the correlation factor between the target feature and



```
[ ]  dataframe.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 30000 entries, 0 to 29999
Data columns (total 12 columns):
 #   Column                        Non-Null Count  Dtype
---  ------                        --------------  -----
 0   Cell ID                       30000 non-null  int64
 1   eNB ID                        30000 non-null  int64
 2   Cell                          30000 non-null  int64
 3   Dist                          30000 non-null  object
 4   ChBW(MHz)                     30000 non-null  int64
 5   Layer                         30000 non-null  object
 6   Capacity@100% GB              30000 non-null  float64
 7   Capacity@90% GB               30000 non-null  float64
 8   Cell Utilization              30000 non-null  float64
 9   Payload GB                    30000 non-null  float64
 10  Avg DL User Thrpt 24 Hrs Mbps  30000 non-null  float64
 11  Avg DL User Thrpt DBH Hrs Mbps 30000 non-null  float64
dtypes: float64(6), int64(4), object(2)
memory usage: 2.7+ MB
```

Fig. 3. Data frame

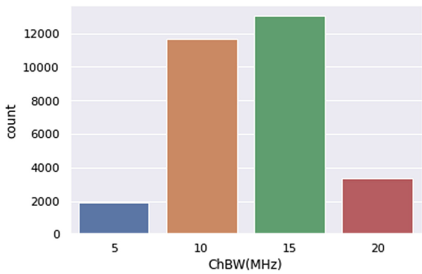


Fig. 4. Channel bandwidth statistical representation

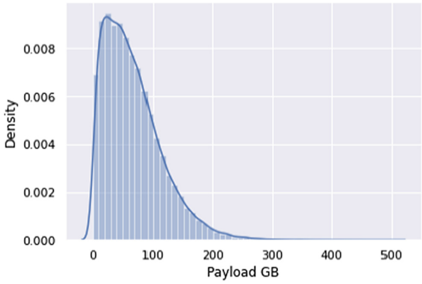


Fig. 5. Payload vs. Density

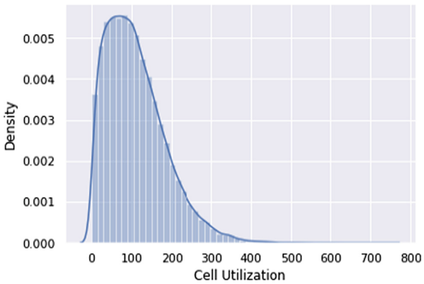


Fig. 6. Cell Utilization vs. Density

other features, as shown in Fig. 8, analyzed that 'Payload GB' has a 0.8 correlation factor out of 1 with 'Cell utilization.'

The data splitting is done into two parts: training and testing data, with training at 80% & testing data at 20% with the help of the 'train-test-split' command, as represented in Fig. 9.

Considered three different threshold levels in target (Payload GB) as:

i. <= 50 for LOW
ii. 51 to 200 for MEDIUM
iii. >200 for HIGH.

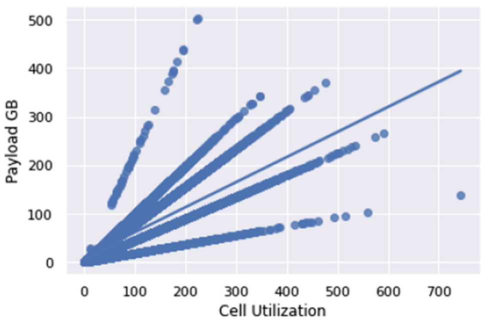


Fig. 7. Relation between payload vs. cell utilization

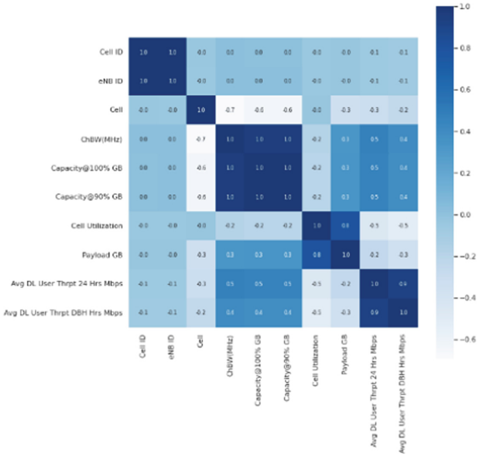


Fig. 8. Correlation Matrix

```
[92] X_train,X_test,Y_train,Y_test = train_test_split(X,Y,test_size=0.2,random_state=2)
```

```
[93] print(X.shape,X_train.shape,X_test.shape)

(30000, 10) (24000, 10) (6000, 10)
```

Fig. 9. Train-Test-Split

Obtained pictorial representation as illustrated in Fig. 10 & numerical count in Fig. 11.

XGBOOST Regression model for optimum evaluation was considered. The steps are evaluated internally when the model is deployed as given below:

The execution steps of hidden layers are represented below.

Step 1: Make an initial prediction and calculate Residuals.

$$\text{Residuals} = \text{Original samples} - \text{predicted samples} \tag{2}$$

Step 2: Develop the XGBOOST model tree

$$\text{Similarity score} = (\text{sum of Residuals})^2 / (\text{Number of Residuals} + \lambda) \tag{3}$$

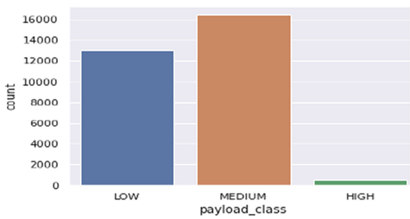


Fig. 10. Payload Classification

```
[98] dataframe['payload_class'].value_counts()

MEDIUM    16440
LOW       13082
HIGH        478
Name: payload_class, dtype: int64
```

Fig. 11. Payload Classification count

Where λ is the regularization parameter, and the default value is 1.

Step 3: Prune the tree.

Step 4: Estimating the output values of leaves

$$\text{Output value} = \text{Sum of Residuals}/(\text{Number of Residuals} + \lambda) \tag{4}$$

Step 5: Now make a new prediction.

Step 6: Calculating the new predictions using Residuals.

Step 7: Repeat steps 2 to 6.

After evaluating the model by training data, we need to predict the optimum payload instances using test data similar to Fig. 12.

Table 1 represents the comparison of the actual payload vs. the Optimum payload out of many instances picked and random samples representation.

$$\%\ \text{payload} = \frac{(Optimized\ payload) - (Actuat\ Payload)}{Actual\ Payload} \times (100) \tag{5}$$

From the table, visualize that for the same channel, we can enhance or decrement the payload without affecting the channel bandwidth and cell splitting. Then at the user

```
[41] test_data_prediction = model.predict(X_test)
     print(test_data_prediction)

[23.180405 20.331095 67.68988  ... 74.10544  21.16056  77.63958 ]
```

Fig. 12. Test Data Prediction

Table 1. Optimized payloads of channel

S.NO.	Actual payload	Optimized payload	% Of Improvement/ Reduction
1	19.69958	23.18041	17.66952123
2	20.84458	20.3311	-2.463412197
3	66.965	67.68988	1.08247592
4	18.9525	21.16056	11.65049466
5	79.22167	77.63958	-1.99703826

Test data Accuracy score

```
[44] model.score(X_test,Y_test)*100

99.84674071704767
```

Fig. 13. Accuracy Score

Fig. 14. Actual vs. Predictions Payload samples

end, customer services will be handled easily. The overall accuracy score obtained from the XGBOOST regression model is 99.8467%, as shown in Fig. 13.

The pictorial visualization of the payload samples in Fig. 14, it indicates the true and predicted instances of each sample lie at a smaller distance from the hyperplane, and the closure is proportional to accuracy.

5 Conclusion

This paper is developed in Colab, an open-source tool suitable for machine learning. XGBOOST regression model was considered for the implementation written in Python script. The feature considered was payload optimized against 12 features such as cell ID, eNB-ID, channel bandwidth, Layers, Cell utilization, Capacity and Payload, and channel bandwidths considered are 5, 10, 15, and 20 MHz, with various layers of L900, L1800, L2100, and mMIMO. The prediction using the XGBOOST regression algorithm for optimum ranking of payload achieved an accuracy score of 99.8467%. The future scope is to study the various model behavior with more datasets obtained from the service providers in predicting payload optimization.

References

1. T. K. Ramesh and Giriraja C V, "Study of reassignment strategy in Dynamic Channel Allocation scheme," 2016 3rd International Conference on Signal Processing and Integrated Networks (SPIN), 2016, pp. 731–734, DOI: https://doi.org/10.1109/SPIN.2016.7566794.

2. I. Katzela and M. Naghshineh, "Channel assignment schemes for cellular mobile telecommunication systems: a comprehensive survey," in IEEE Personal Communications, vol. 3, no. 3, pp. 10-31, June 1996, DOI: https://doi.org/10.1109/98.511762.
3. G. Jia, Z. Yang, H. -K. Lam, J. Shi, and M. Shikh-Bahaei, "Channel Assignment in Uplink Wireless Communication Using Machine Learning Approach," in IEEE Communications Letters, vol. 24, no. 4, pp. 787–791, April 2020, DOI: https://doi.org/10.1109/LCOMM.2020.2968902.
4. N. I. A. Razak and M. H. Wahid, "Telecommunication Customers Churn Prediction using Machine Learning," 2021 IEEE 15th Malaysia International Conference on Communication (MICC), 2021, pp. 81–85, DOI: https://doi.org/10.1109/MICC53484.2021.9642137.
5. Y. Sun, M. Peng, Y. Zhou, Y. Huang, and S. Mao, "Application of Machine Learning in Wireless Networks: Key Techniques and Open Issues," in IEEE Communications Surveys & Tutorials, vol. 21, no. 4, pp. 3072–3108, Fourth quarter 2019, DOI: https://doi.org/10.1109/COMST.2019.2924243.
6. T. T. t. Nguyen and G. Armitage, "Training on multiple sub-flows to optimize the use of Machine Learning classifiers in real-world IP networks," Proceedings. 2006 31st IEEE Conference on Local Computer Networks, 2006, pp. 369–376, DOI: https://doi.org/10.1109/LCN.2006.322122.
7. S. M. Rachmawati, D. S. Kim and J. M. Lee, "Machine Learning Algorithm in Network Traffic Classification," 2021 International Conference on Information and Communication Technology Convergence (ICTC), 2021, pp. 1010–1013, DOI: https://doi.org/10.1109/ICTC52510.2021.9620746.
8. J. Jiao, X. Sun, L. Fang and J. Lyu, "An overview of wireless communication technology using deep learning," in China Communications, vol. 18, no. 12, pp. 1–36, Dec. 2021, DOI: https://doi.org/10.23919/JCC.2021.12.001.
9. S. S. Hlaing, M. M. Tin, M. M. Khin, P. P. Wai, and G. R. Sinha, "Big Traffic Data Analytics For Smart Urban Intelligent Traffic System Using Machine Learning Techniques," 2020 IEEE 9th Global Conference on Consumer Electronics (GCCE), 2020, pp. 299–300, DOI: https://doi.org/10.1109/GCCE50665.2020.9291790.
10. Donghong Qin, Jiahai Yang, Jiamian Wang, and Bin Zhang, "IP traffic classification based on machine learning," 2011 IEEE 13th International Conference on Communication Technology, 2011, pp. 882–886, DOI: https://doi.org/10.1109/ICCT.2011.6158005.
11. D. S. Harsha, B. Suresh, and K. L. V. Sai Prakash Sakuru, "Machine Learning-based Oddity Detection of Smoke and Gas Sensor Data in a Large Gated Community," *2022 IEEE International Symposium on Smart Electronic Systems (iSES)*, Warangal, India, 2022, pp. 619–622, DOI: https://doi.org/10.1109/iSES54909.2022.00136.
12. J. Chen, Y. Gao, Y. Zhou, Z. Liu, D. p. Li, and M. Zhang, "Machine Learning Enabled Wireless Communication Network System," 2022 International Wireless Communications and Mobile Computing (IWCMC), 2022, pp. 1285–1289, DOI: https://doi.org/10.1109/IWCMC55113.2022.9824835.
13. S. Laha, N. Chowdhury and R. Karmakar, "How Can Machine Learning Impact on Wireless Network and IoT—A Survey," 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 2020, pp. 1–7, DOI: https://doi.org/10.1109/ICCCNT49239.2020.9225652.
14. Dataset link: https://doi.org/spreadsheets/d/1RCkh50RZWmXZ86HyUmLsRbJ6eNFf5A/edit?usp=share_link%26ouid=116345830286444700760&rtpof=true%26sd=true.