Intelligent RAN Power Saving using Balanced Model Training in Cellular Networks

Vaibhav Singh, Maruti Gupta and Christian Maciocco
e-mail: {vaibhav1.singh, maruti.gupta, christian.maciocco}@intel.com
Intel Labs

***Abstract*—Optimizing power consumption of 5G systems and next generation technology deployments is a critical problem. It is essential that the solution for optimizing power consumption takes into account the tradeoffs with maintaining service level agreement (SLA). Mobile network operator (MNO) may have different priorities for the objectives of saving power and for maintaining SLA, which depends on factors, such as customer contract, location, time of day, type of traffic, etc. In this paper, we design an intelligent solution using switching cells on-off action to save power, using machine learning (ML)/ deep learning (DL) methods to forecast future traffic load. We firstly identify the problem of training imbalance in traffic load prediction due to data imbalance in real cellular networks, and MNO preferences for the competing objectives of saving power and SLA maintenance. We then propose a novel solution that incorporates *Balancing Loss Function*, which addresses the training imbalance problem. Compared with the performance of previous approaches such as Mean Square Error (MSE) minimization traffic forecast based methods, we demonstrate using network field data that our method is able to achieve upto 3X improvement in service quality outage, with fairly similar power savings.**

***Index Terms*—Power saving, SLA, Load Prediction, Machine Learning, RRM Optimization, Bias, Balanced Training**

I. Introduction

Power saving in RAN is critical for reducing operating cost and following environmental stringent requirements, while ensuring SLA in a cellular network [14], [15]. Different actions can be taken, as a function of future load, in a cellular network to save system level power. Consider an example scenario, where there are a couple of co-located cells, cell-A operates at a higher frequency carrier and cell-B operates at a lower frequency carrier. If we can predict that the resource utilization/load (both terms used interchangeably in the paper) in cell-A and cell-B will be low for a future time interval, then carrier corresponding to cell-A can be switched off, while offloading its UEs to co-located cell-B carrier in that time interval, thereby saving operating power by switching off carrier corresponding to cell-A. We refer to this carrier on-off as cell on-off in the rest of the paper. Cell switch-off allows an entire cell (i.e. a frequency layer) to be switched off, leading to most of its RF and Digital Front End components to be set to sleep mode. In addition to cell on-off, different energy saving alternative actions can be taken in cell-A to reduce power consumption, for example, reducing operating bandwidth, reducing/on-off MIMO channels, other advanced sleep modes etc.

Currently, the AI/ML based intelligent energy saving use case leveraging data from the RAN, is being discussed in 3GPP and O-RAN standards [21] [22]. The introduction of an intelligent controller leverages AI techniques to embed intelligence in RAN functionality. While messaging between intelligent controller and different RAN components; functionality and inputs of energy saving AI/ML models; are undergoing standardization discussions, the detailed energy saving AI/ML algorithms are out of scope of standardization. In this work, we identify certain critical aspects for designing an efficient energy saving solution, based on our field data study. Here we focus on cell on-off energy saving action, while presenting a general solution approach which can be applied to different energy saving action scenarios.

Estimating future load correctly is essential for designing cell on-off solution [11], [12]. Suppose in the earlier example scenario, the predicted load were lower than the actual load. Switching off cell-A, in this case may cause SLA outage as cell-A would be switched off while its actual future load were high. *SLA outage* here denotes the degradation of service quality due to incorrectly switching off a cell, and the terms are used interchangeably in the paper. In the case of load overprediction, cells may not be switched off when the actual future load is low, thereby, resulting in less power saving while not affecting the service quality. There is an innate tradeoff between the two metrics of power saving and service quality degradation, which is governed by the estimate of future load [9], [10]. Therefore, we aim to incorporate MNO's preferences in the future load estimate. MNOs may prefer to either prioritize maintaining service quality or high power saving in their network based on factors, such as customer contract, geographical location, time of day, type of traffic etc. For example, in downtown area during the day operators would place higher weight on avoiding underprediction events causing service quality degradation, while ensuring sufficient power saving; whereas at night time during less traffic hours, the operator may choose to place higher weight on power saving for these cells.

In the earlier discussed power saving scenario, the performance of the load prediction algorithm in cases of *high load* may be significantly more important than the performance in other load regimes. This is because, underprediction in high load scenario can cause service quality outage, whereas, prediction error in low load scenario could result in less power saving, which could be less catastrophic for the operators that

ISBN 978-3-903176-49-2 © 2022 IFIP

have more preference towards SLA maintenance. In addition, based on our analysis of the field data, we identify that for a particular cell, majority of load samples does not belong to high load regime, whereas the intended region of more importance could be high load regime which constitute minority number of load samples. Therefore, there is a mismatch in the importance of cell load regime and number of samples in the corresponding region and there is a bias in the data. Training on biased data can lead to incorrect predictions for minority class of samples [18]–[20]. This limited number of samples in the high load regime, and further the different MNO priorities for competing objectives of power saving and SLA maintenance leads to *imbalance in training*. Not accounting for the imbalance, leads to biased and incorrect load estimates and undesirable power saving/SLAs.

We further observe that each cell in a network may have different load distribution; different power saving priorities, and hence a different degree of imbalance. Therefore, we need to address the problem of training imbalance at a per cell level based on the corresponding data characteristics and MNO preference. Previous works [1]-[6], do not make the observation of mismatch of majority of load samples, and the intended range of cell load for the power saving at a per cell/s level. These solutions were designed under the assumption that the data is balanced and evenly distributed in different load regimes. They follow an approach to come up with sophisticated ML models to decrease mean error (for example, mean square error) for load prediction over the entire load range, but do not focus on performance of load prediction for power saving use case in load regime of importance. Recently training imbalance and fairness has received high attention in ML community applied to different fields for example computer vision etc [18]–[20]. However, it has not been clearly identified and addressed in intelligent RAN power saving design to the best of our knowledge.

While mean error metrics, for example MSE, is a commonly used notion of measuring the prediction accuracy and captures errors from an overall perpective, which means it calculates the loss by firstly sum up all the errors from the whole data set and then calculates the average value. This can capture the errors from the load in majority regime and minority regime equally when the load data sets are balanced. However, when the data set is imbalanced, the error from the majority regime contributes much more to the loss value than the error from the minority regime. In this way, MSE loss function is biased towards majority class and fails to capture the errors from two regimes equally. Further, prediction error in some regimes may be much more important than in other regimes making MSE an inadequate loss function. To address these drawbacks of MSE and previous load prediction models, we propose to use a load prediction solution, where the loss function is tuned to the imbalanced RAN load data and operator preferences.

The idea of intelligent RAN power saving using cell on-off action is not new, and has been explored earlier [9]–[12]. However, the earlier solutions used load prediction based upon MSE minimization, and did not balance the service quality outage and power saving based on MNO's preferences. In this work, we make the following contributions :

- Using real field data we identify the problem of imbalance in training for power saving due to load samples in more important load regime being a minority in number, and MNO preferences .
- We propose a novel *Balancing Loss Function (BLF)* to balance the training for load prediction. This approach provides us with parameters that act as handle to balance the power saving and SLA outage based on operator preferences. To the best of our knowledge, balanced training based load prediction has not been used earlier to balance SLA and the RAN power saving.
- We characterize the performance of load prediction into two types of errors, indicating SLA outage and power saving, rather than previously used single mean error metric
- We demonstrate a reduction in SLA outage by upto 3X using our proposed approach, as compared to baseline approaches, while having fairly similar power saving.

Next, we provide details on the field cellular dataset in section II, and corresponding characteristics that provide us the motivation for our solution. In section III, we introduce the problem formally and propose our solution in section IV. We evaluate the proposed solution with help of the dataset and compare the performance w.r.t. baseline algorithms in section V. And finally conclude in section VI

II. Network Dataset

We analyze open-source Telecom Italia [17], call detail records (CDRs) data, which contains two month records of Milan city cellular network. The dataset contains CDRs at a granularity of 10 minutes. CDRs are used as an estimate of data traffic load with appropriate assumptions listed in the next section. The city is divided into square grids (10,000 in number) and data traffic per grid across the city is studied. We observe temporal and spatial variations of the traffic volume, as shown in Fig.1, for 100 grids in 2 weeks interval (144 samples per day). In the figure, x-axis denotes the time samples and y-axis denotes the adjacent square grid number, and the brighter color corresponds to higher cellular data traffic. Overall, the traffic is observed to be higher during the day and lower at night. Also, some of the grids have much higher traffic volume as compared to others. Note the imbalance of darker and ligher regions in the plot, indicate bias towards low traffic volume in the RAN data.

Next, we quantify the imbalance by calculating the percentage of load samples lower than a particular load threshold in different grids. The threshold is set to be half of high percentile (95th) traffic across all grids over entire time duration (reason for factor half is that in next sections we will consider sum of traffic for pair of grids to make energy saving decisions). We observe, that in Fig.2 the imablance of load across different grids is variable. In the figure, y axis denote the percentage of load samples lower than threshold load and x axis denotes grid number. In some grids the imbalance is much more severe

than in other grids, and we need to take a solution approach that addresses the imbalance based on data characteristics of each grid.

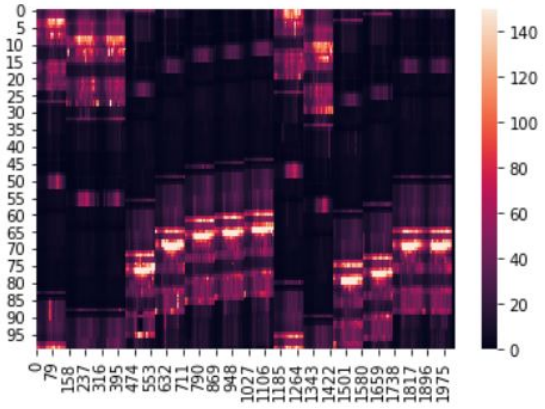

Figure 1: Cellular traffic (CDRs) heat map w.r.t. time (2weeks, x axis) and grids (100, y axis)

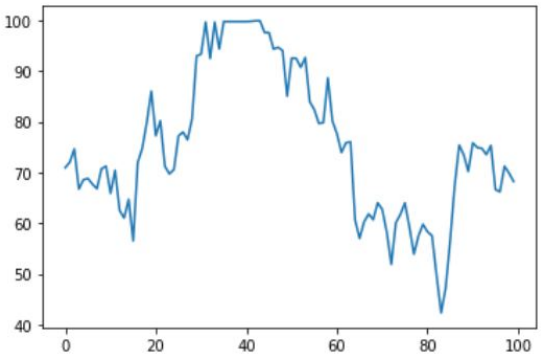

Figure 2: Percentage of samples (y axis) in different grids (x axis, 100 grids) that have traffic (CDRs) below a high traffic threshold .

We finally observe, that the traffic volume per cell at a granularity of 10 min. have sudden peaks and valleys, and does not follow a regular pattern as also earlier observed in [12], [13]. We observe a strong correlation of future load samples for a cell with immediate past load samples. Therefore, limited past samples can aid in prediction of future load samples. However, there can be sudden peaks which could be difficult to predict and can cause underprediction at relatively high traffic, thereby leading to SLA outage.

A. Mapping data to Power saving in cellular network

Next, we map the Telecom Italia dataset to the power saving scenario in cellular networks. Though the dataset is rich in spatial and temporal domains, we have to make appropriate assumptions to map the dataset to cellular power saving scenario. Firstly, we map the grids to cells, by assuming n and $n+1^{th}$ grid constitute a pair of co-located cells, where n corresponds to higher frequency cell and $n+1$ maps to lower frequency cell, $n \in 1, 3, 5.., ..$ Note that we do not have cells per grid information available in the dataset. Given that each grid is a square of side $\sim 200m$, there would be limited number of cells per grid in the actual deployment. Hence, the temporal characteristic of cellular traffic is mantained, by mapping each grid to a cell. Further, we observe that there is significant correlation of a grid traffic with adjacent grid, and therefore justifies our assumption of mapping two consecutive grids to two co-located cells given the limited information in dataset.

While there are sophisticated traffic models available, we assume a simple traffic model where each call duration consists of 2 MB data transfer. Therefore, we can estimate traffic intensity/data rate which is defined as amount of MBs of data incident on the cell per 10 min. time interval, in each grid using the CDR data. Although, each cell in the network can have a different maximum data rate based on factors such as bandwidth, transmission scheme etc., for simple mapping we assume the maximum data rate achievable for a cell is the top (for example 98th/99th) percentile data rate seen across all cells over the entire time duration. The estimate of data rate and maximum data rate for a cell helps us to approximate the load/resource utilization, and come up with power consumption using well known power consumption models [16].

III. System Model and Problem Statement

Let us say, given a cluster of cells (say Macro cells and small cells) has common service area A, with each cell i having n_i sleep/active states. We need to determine what sleep/active state each cell should be in, for minimizing power consumption and to mantain service quality. Suppose, we have $|B|$ cells in the cluster, and each cell has at max. K sleep states, then optimistically speaking there are $K^{|B|}$possible states for the cluster. With dense network scenarios, $|B|$ can be large causing the complexity of the problem to be very high. Note that each cell cannot take 'any' sleep state value, for example in a 2-cell co-located scenario, cell with higher center frequency will go into sleep state rather than lower frequency cell to guarantee coverage. Similarly, cells based on licensed frequency may have higher priority to be in sleep state given less traffic, as compared to unlicensed cells. We assume that this priority order of cells is known or learnt over time (longer time scale) and is denoted by C. Ties of cells with same priority are broken randomly.

Let Z denote a vector of length $|B|$ and $Z \in C$ is sleep state vector for a future time interval T, where each entry in Z corresponds to the sleep state for corresponding cell. Further, let X denotes future load for the group of cells for time interval T. The power saving problem can be stated as:

Power Saving Problem

$$\boxed{\begin{array}{l}\max_Z f(X,Z) - P(X,Z) \\ \text{subject to } Z \in \mathbf{C}\end{array}} \quad (1)$$

Where, $P(X,Z)$ is a function of system power consumed, for example $P(X,Z) = \alpha * P_{licensed}(X,Z) + \beta * P_{unlicensed}(X,Z)$, $P_{licensed}$ denotes the system power consumed on licensed spectrum which is a function of vector X , whereas $P_{unlicensed}$ corresponds to unlicensed spectrum.

$f(X, Z)$ is a measure of system service quality outage/ SLA outage, for example $100 - \mu(X, Z)$, where μ is the percentage of outage events in high load scenarios. α and β signify the relative importance/priority of power saving w.r.t. performance. The constants α and β can be set by network operator to prioritize power saving on licensed and un-licensed bands over the performance and vice versa. The value of α and β can change across different cluster of cells, and also with time within each cluster. The difference of the $f()$ and $P()$ in (1) denotes, that we aim to maximize the service quality and minimize power consumption.

Although the number of cells in cluster C is usually limited in a practical deployment limiting the complexity, the major challenge in solving the problem (1) is to obtain a reasonable estimate of X, and further there is no closed form expression available for the objective. Power consumed and SLA would depend on the combination of the following factors: (i) surrounding conditions for example, channel models etc.,(ii) configuration parameters for example, scheduler algorithm used etc., (iii) type of traffic, (iv) implementation details for example, implementation of non-standardized algorithms. Moreover, combination of parameters in these factors can grow exponentially. Thus, we abstain from theoritical modelling to keep the solution general across different surrounding conditions, configuration settings and implementation details, and do not use a closed form expression for the SLA and power consumption function. We limit the number of cells in a cluster to 2 from now onwards in the paper, so as to focus on the load prediction aspects of the problem in a practical deployment, though the proposed solution can be extended to multiple cells based on the priority order C.

We assume that a rich pipeline of RAN data comprising of history of load information for a cell and its neighbors, such as number of RRC connected/ active users, PRB usage in DL/UL are available, which can be leveraged to design intelligent power saving scheme. Further, data available to us include cell specific data which is more or less static with time. Metadata comprising of events like holidays, games in cell deployed in a stadium, and other events affecting traffic in the network is available to us as well. Finally, we also assume access to performance data for example cell throughput, SLA outage, power consumtion etc. The time granularity of the data is assumed to be non real time, that is order of seconds/minutes.

IV. Proposed Solution

We address the problem of intelligent power saving for RAN by first proposing a general solution framework. Next, we provide the load prediction solution tuned to balance the power saving and SLA.

A. Solution Framework

In Fig.3, we provide a solution framework for intelligent system level power saving problem. The traffic collection module leverages data from the RAN to come up with load trends per cell, so as to train the load prediction module. In our study, RAN data collection maps to leveraging the CDR

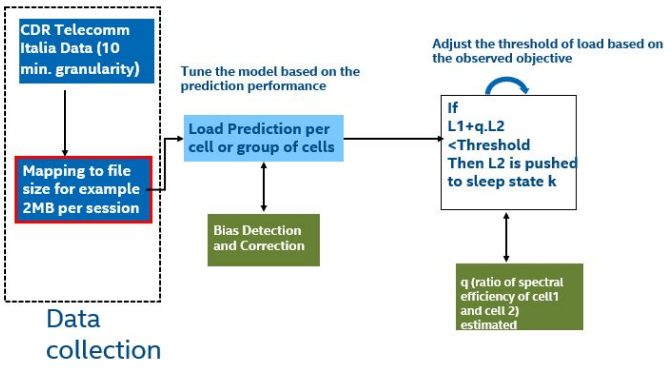

Figure 3: Solution framework for power saving

Telecom Italia dataset and estimating traffic volume based on assumptions mentioned in section II. The load prediction module predicts future load estimate using the relevant features from the RAN data. The load prediction should incorporate the bias in number of samples and importance for a particular operating range. Further, the total estimate of predicted load ($L1 + q.L2$): ($L1$ being load on lower frequency carrier, and $L2$ corresponding to higher frequency carrier) is compared against a threshold load to make the cell switch on-off decision. If the total load estimate is lower than the threshold, we take a decision to switch off cell operating at higher frequency. On the other hand, we make a decision of cell operating at higher frequency to be on if the estimate exceeds threshold. Further, the load estimate should incorporate the spectral efficiency ratio (q) of the cells to balance the cell/operating conditions in different cells. The prediction module is trained on the data, where the cell switch on-off is not enabled so that the future load is not affected by the on-off action. The algorithm can be extrapolated to determine on-off action when the number of cells is more than two, by considering the total estimate of predicted load and corresponding thresholds.

The load prediction model can be tuned based on prediction performance of the model. Moreover, the performance of the model may not be restricted to prediction error but can include maximizing the objective (1). Further, on-off threshold could be tuned so as to maximize the objective. Next, we choose the cost function best suited for training the model for load prediction. The choice of cost function is critical for balancing the SLA and power saving, as desired by problem objective (1).

B. Cost Function

As discussed earlier, there might not be sufficient number of training samples lying in the load regime of importance, thereby causing bias in trained model. Further, we need to prioritize power saving and SLA based on MNO preference. The Mean Square Error or L2 cost function used for training the model is the average of sum of squared difference between actual and predicted load. Similarly, L1 cost function used for training the model is the average of absolute value of difference between actual and predicted load. L1 loss function can be used to predict the median future load, while L2 loss

function can be used to predict the mean future load. Note that, in the L1 and L2 cost function all over-predictions or under-predictions errors have equal weight and do not account for either countering the bias or balancing the two metrics of SLA and power saving based on the operator. In the case of power saving, under-predictions and overpredictions need different weights naturally, based on balancing of SLA and power saving required by the network operator. The L1 or L2 cost minimization load prediction could be misleading in this case. Therefore, we design a tunable cost function.

We introduce two parameters $\nu \& w$, where ν could be used to tune the balance of underprediction and overprediction. Further, w is used to balance the number of samples in intended region of samples for the power saving use case. We propose a loss function based on $\nu \& w$, given by:

Balancing loss function (BLF)

$$L_{\nu,w}(X,\hat{X}) = \sum_{k:e_k>0, X_{t+k}\in S_L} w*\nu*(e_k) + \sum_{k:e_k<0, X_{t+k}\in S_L} w*(-1)*(e_k) + \sum_{k:e_k>0, X_{t+k}\notin S_L} \nu*(e_k) + \sum_{k:e_k<0, X_{t+k}\notin S_L} (-1)*(e_k)$$

where $w > 0 \& \nu > 0$. X_{t+k} denotes actual load value corresponding to future time $t+k$ and $\hat{X_{t+k}}$ is the predicted value of load corresponding to the time step $t+k$, where $1 =< k <= T-t$. e_k is the underestimation error, which is $X_{t+k} - \hat{X_{t+k}}$. S_L is the set of load corresponding to desired regime of importance for the power saving use case.

The first two terms of the loss function corresponds to the load samples in the desired regime of importance S_L, and weight w is used to balance the mismatch in the number of samples in the desired region. Thereby, multiplying by a weight w. The first term of summation in the above loss function, chooses training samples for which the actual load is greater than the predicted load ($e_k > 0$) and the second term of the summation corresponds to overprediction ($e_k < 0$) . In the first term corresponding to underprediction, the error is weighted by a factor ν as compared to second term corresponding to overprediction error. If we choose a high $\nu = 9$, then we favor overprediction, by weighing the overprediction error by 1 and underprediction error by 9. Favoring overprediction would lead to prioritizing mantaing the SLA as compared to power saving. The last two terms of the loss function corresponds to the load samples not in the desired S_L, while mantaining the underprediction and overprediction error ratio same as it is in the first two terms, are weighted w times less than the first two terms to balance the bias.

We can calculate appropriate value of w for a particular cell based on the load range of importance $[L_l, L_h]$. Appropriate w is estimated as the ratio of number of training samples not in the range $[L_l, L_h]$ to the number of samples inside the range. Therefore we amplify the error for samples in load range of importance, given the samples in intended region are a minority. The calculation of ν is challenging as the translation of overprediction and underprediction error to the SLA maintenance and power saving is complex and not known. The following procedure can be followed:

- The prediction model predicts load vector for P , where P contains ν elements such as [1,2,...,9,10] and the value of ν chosen based on load prediction performance in a particular use case. The value can also be chosen based on maximizing the objective (1) based on MNO preference.
- Alternatively, ν (or jointly with w) should be chosen as a hyperparameter and it could be chosen for power saving using **Bayesian optimization** or random search. The objective is based on observed value of the objective in (1).

Though we use v to balance SLA maintenance with power saving, it can also be used to balance performance metrics for example, cell throughput, latency etc. with power saving as well.

C. Prediction Model Solution

As pointed earlier, we observe that the on-field cell load time series has dynamic peaks and troughs, which are difficult to predict. Therefore, estimating point estimate value or conditional distribution of the future load is challenging. Function estimation using model parameter weights, forms a natural solution choice for load prediction. We can use the observed load samples and appropriate features for training the model weights. Neural network (NN) architectures combined with stochastic gradient descent algorithm are known to be very efficient predictors in different applications. Also, XG-boost ML models are known to perform very well in many application areas and form a natural choice for the function estimation. Various features along with the past load samples can be input into these standard ML/DL models for a particular cell c. We identify some features which form a reasonable correlation with cell load mentioned in Appendix A.

D. Evaluation criterion

The evaluation criterion for a load prediction should essentially be based on the RAN power saving switch on-off decision accuracy, instead of standard mean/median error metrics. Using the metrics for evaluation, which are based on mean error rather than assigning higher weight to a particular range of load (important to a use case) can lead to erroneous performance evaluation. For example, suppose objective in the power saving use case we need to detect if the load is overshooting a threshold (say 70%), average percentage error for a load prediction model (over all load samples) may show error $< 10\%$ but in the high load regime (where there are fewer samples, due to to innate bias in data) the average error might be much higher (say $> 40\%$), thus, leading to SLA outage. Therefore, mean error metrics over entire samples gives us a false sense of performance, due to the training imbalance observed from field data. This observation has not been considered in previous energy saving works, and mean error metrics have been used to evaluate the performance of load prediction.

We evaluate two types of error events indicative of RAN power saving performance for load prediction module at cell/-group of cell level:

- Type A- When sum of load of pair of cells is high and there is underprediction of more than a particular value, thereby indicating SLA outage
- Type B- When sum of load is low and there is overprediction by more than a particular threshold, thereby indicating less power saving

Finally, we also evaluate the performance of the solution in terms of SLA outage and Power consumed and demonstrate that the balancing cost function based load prediction helps to balance the two performance metrics much better than baseline predictions.

E. Model Architecture

We evaluate the Multi Layer Perceptron (MLP) NN based model for load prediction. NN architecture evaluated using tensor-flow for load prediction is the following: *(a) Model = Multi layer Perceptron model, (b) Number of hidden Layer = 1, (c) Number of neurons per hidden layer = 10, d) Epochs = 200, (e) Batch size = 4, (f) Learning rate = 0.003, (g) Optimization algorithm= Adam optimizer, (h) Cost function= Balancing loss function.*
We tune the NN architecture and obtain the best performance using a simple single layer NN and past five load samples with additional features as an input to the model. The features in addition to past load samples are the following: *(i) First order moment of load at same time across multiple days, (ii) second order moment of load at same time across multiple days and (iii) time of the day*. We further tune NN architecture with more layers, along with more complicated LSTM models and also with additional features like higher order moments, neighboring cell load etc. We also evaluated XG-boost ML algortihm. However, the different architectures showed no additional gains. Thus, to be concise, we present results based on MLP model in the paper. For simplicity, we use the same value of ν for all the cells, and different values of w per cell. Thus the cost function is based on data charactacteristics per cell and also MNO preference. We use different ν values and find the best one that obtains a reasonable balance between the SLA maintenance and power saving objective (1).

F. Power Saving model used

The power saving model used for pair of co-located cells is the following

- When both the cells are on power consumed is $334 + 2.73 * (x1 + x2)$ W, where x1 and x2 is percentage load for cell operating on higher and lower carrier frequency respectively.
- When the cell operating at higher frequency is switched off and the lower carrier frequency is on, power consumed for pair of cells is $167 + 2.73 * (x1 + x2)$ W.

The model is extrapolated from [16]. Power consumption is calculated and summed for 250 pair of cells (500 cells) across 2 days (144*2) test data. Finally, the calculated average power saving per cell per sample results are shown in section V.

V. Evaluation

We present an extensive evaluation of the proposed power saving solution on the cellular network field data described in section II. We firstly evaluate the load prediction solution based on the *Type A and Type B* error metrics (defined in section IV), while comparing with the baseline solutions. Further, we evaluate the SLA and power consumed balance while demonstrating the advantage of using our *balancing loss function* based load prediction approach. Finally, we provide an example scenario for choosing the right ν for optimizing the desired objective, while balancing SLA outage and power saving percentage.

A. Load Prediction evaluation

We aim to predict the next future sample value of traffic volume in a cell using the past traffic volume samples along with additional features. We evaluate the performance of load prediction on the field data at a granularity of 10 min. (averaged at 10 min. timescale).
The field data processed us is worth 14 days (14*144 samples). For a given cell, first 12 day consecutive time samples (12*144 samples) are used to train the model per cell and the remaining two day data is used to test the performance of model. The performance is evaluated for 500 cells with appropriate assumptions as mentioned in section II. Note that for each cell we train a separate model, and the performance is aggregated across test sets of different cells. The past load samples at a granularity of 10 min. are used to predict the next 10 min. load. We compare our solution with following baseline predictors:

– *Previous Sample Predictor*– The next load sample value is equal to the present load sample value.
– *Mean Square Error Predictor*– The cost function used for training the load predictor is the average of square of difference of predicted load value and actual future load value.
– *ARIMA predictor*– This is a well known classical time series predictor. Note that weight update is based on all the past load samples available rather than just training set. Model order used is (5,1,0) which fits best on the data.
– *Window Average*– Next sample is the average of last five load samples.

We evaluate performance of our proposed solution based on BLF for different v value for load prediction. For each cell, we calculate appropriate w for the loss function using training data based on section IV solution. We further, calculate the percentage of load samples higher than half (as we consider pair of co-located cells) of high load

threshold (95 percentile) and if it is $> 50\%$, we use BLF with corresponding v value to balance the training, and if the value is < 50 we use MSE loss function for the cell as we have sufficient samples for high importance load regime.

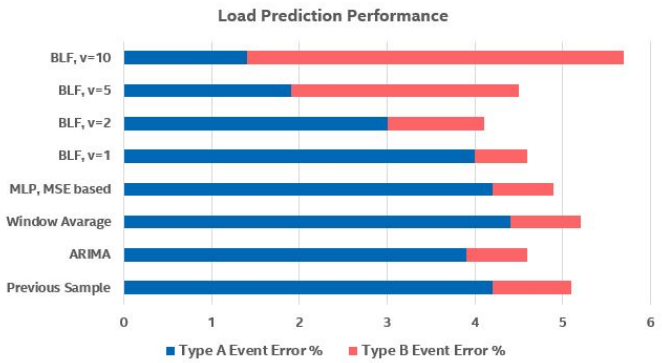

Figure 4: Load prediction evaluation based on Type A & Type B events

MNOs prefer Type A percentage to be as low as possible, while mantaining low Type B to be reasonably low. In Fig. 4 we evaluate the Type A and Type B errors per cell for different load prediction algorithms. For Type A events, we consider scenarios when sum of actual load of pair of co-located cells is greater than the threshold for high load which is 95 percentile (of traffic gathered from 500 cells), and Type A error is commited if the sum of predicted load of pair of co-located cells is lower than the high load by an amount greater than buffer of 5 MB. Similarly, for Type B events, we consider scenarios when actual load sum is lower than the threshold of low load affected by overprediction, which is 95 percentile minus a buffer of 5MB, and Type B error is commited if the prediction is higher than the threshold by an amount greater than buffer of 5 MB.

In Fig. 4 we observe that the BLF based MLP prediction outperforms the other baselines w.r.t. Type A events, which are very important for MNOs. For $\nu = 10$ the Type A events are reduced to 1.4%, while affecting the Type B accuracy by upto 3.6% w.r.t. MSE. Whereas, the MSE based prediction suffers from high 4.2% Type A errors leading to SLA outage. Moreover, BLF for $\nu = 5$ is able to balance the two event types. Therefore, BLF based approach can outperform the baseline solutions by efficiently balance the two types of error. Finally, comparing the Type A events value of 4% for BLF, $\nu = 1$ to percentage of 1.4% for BLF, $\nu = 10$ underscores the significance of the parameter ν in the proposed BLF.

B. Power saving and SLA outage evaluation

Next, we aim to evaluate the actual power saving and SLA outage using the solution in section IV. We make decision to switch off a high frequency cell, if the sum of predicted future load of the two cells is lower than 95 percentile (of traffic gathered from 500 cells). The power saving model used is extracted from [16] (more details in section IV F). In the below table, we evaluate the percentage of SLA outage events (translates to type A events) and power saving for different load prediction models. We observe that, baseline solutions provide at best an SLA outage event percentage of 3.9 %, while on average power utilization achieved is 144 W. However, the MNO would suppose want to further reduce the outage percentage to say $< 2\%$. Here we demonstrate that the BLF proposed for the RAN load prediction is critical in such a scenario. For example, BLF $\nu = 10$ for the MLP reduces the SLA outage to 1.4 % while marginally increasing power consumed. BLF based prediction is able to improve the SLA by 3X as compared to MSE cost function. Therefore, BLF based ML model are critical to balance both the outage and power consumption. Without much loss in power saving, we can balance the SLA outage which is not possible with other baseline load prediction methods. Note that, proposed approach obtains similar gains over baseline schemes by setting different on-off threshold values as well. Ideal prediction algorithm refers to a genie based scenario, where future load are known accurately beforehand. Ideal prediction algorithm obtains 32.5 % power saving over the case when there is no power saving (213.2 W power consumed in case of no power saving).

Prediction Algorithm	Outage Events	Power consumed per 10 min per cell
Previous Sample Prediction	4.2 %	143.9 W
ARIMA Prediction	3.9 %	144.6 W
Window Average	4.4 %	143.9 W
MLP MSE prediction	4.2%	143.8 W
BLF v=1 Prediction	4 %	143.7 W
BLF v=2 Prediction	3 %	144.6 W
BLF v=10 Prediction	1.4 %	147.6 W
Ideal	0%	143.9 W

Figure 5: Service quality outage and Power saving using different load prediction algorithms

C. Choosing right γ based on operator preference

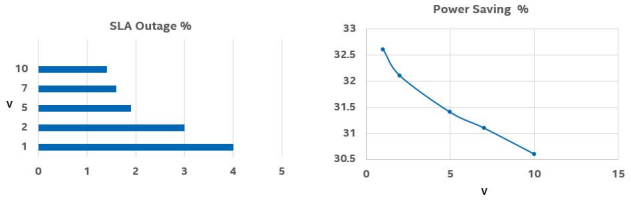

Figure 6: Service quality outage and Power saving tradeoff

Finally, we aim to chose the right ν based on MNO's preferred objective. In Fig. 6 we plot the SLA outage, and power saving advantage of using the corresponding load prediction algorithm w.r.t. always all cell on case. As can be seen from the figure, appropriate γ can be chosen so as to balance the SLA outage and power saving. For example, suppose the objective is $maximize_{\gamma} k * f(\gamma) + P(\gamma)$, where P denotes percentage gain in power saving w.r.t. previous sample predictor, f denotes the functional form of SLA outage and k denotes the weight of SLA maintenance over power saving. Let us suppose avoiding outage is 100% more important than saving power for an operator, k value is 2. Also , f is $\mathbf{1}_{(SLAoutage<2\%)}$. For $\nu = 7$ objective is 2*1 -1.5=0.5, $\nu = 5$ objective is 2*0-1.1= -1.1, we will choose $\nu = 7$ out of the two. Similarly, we will choose $\nu = 7$ over value of 10, based on the objective function. Thus, we can select the appropriate ν dynamically, from different values of ν to balance SLA and power saving based on MNO preference.

VI. Conclusion

We have proposed a balancing loss function based solution for the problem of balancing SLA outage and power saving in a cellular network. The proposed solution achieves superior performance on real field data as compared to the state of the art MSE cost based ML and other baseline solutions. The proposed technique will apply to a broad set of decision/selection to optimize energy and we're here showing a specific action space of cell on-off. There remain some open issues we have not explored here. For instance, in this work we have considered SLA outage as a performance metric, but not mapped our solution to performance metrics such as latency, throughput etc. We used ML models per cell but multiple cells could be grouped/clustered based on different criterions, with a common ML model so as to improve the performance of load prediction performance. Also, we have not explored the performance of the solution using online training [7]-[8] approaches such as reinforcement learning for the problem. We plan to expand our work incorporating these issues and more energy saving actions in the future.

References

[1] S. Di, D. Kondo, and W. Cirne. Host load prediction in a google compute cloud with a bayesian model. In *The International Conference for High Performance Computing, Networking, Storage, and Analysis*, 2012.

[2] W. Wang et.al. Cellular Traffic Load Prediction with LSTM and Gaussian Process Regression. *IEEE ICC*, June 2020.

[3] S. Jaffry et al. Cellular Traffic Prediction using Recurrent Neural Networks. In *IEEE ISTT*, 2020.

[4] J. Wang, J. Tang, Z. Xu, Y. Wang, G. Xue, X. Zhang, and D. Yang. Spatiotemporal modeling and prediction in cellular networks: A big data enabled deep learning approach. In *Proc. IEEE INFOCOM*, 2017.

[5] X. Wang, Z. Zhou, Y. Zheng, Y. Liu, and C. Peng. Spatio-temporal analysis and prediction of cellular traffic in metropolis. In *IEEE 25th International Conference on Network Protocols (ICNP)*, 2017.

[6] C. Zhang et al. Deep transfer learning for intelligent cellular traffic prediction based on cross-domain big data. *IEEE Journal on Selected Areas in Communication*, 37(6), June 2019.

[7] G. Ditzler and R. Polikar. Incremental learning of concept drift from streaming imbalance data. *IEEE Trans. on Knowledge and Data Engineering*, 25(10), Oct. 2013.

[8] S. Wang et al. A systematic study of online class imbalance learning with concept drift. *IEEE Transactions on Neural Networks and Learning SystemsW*, 29(10), Oct. 2018.

[9] G. Vallero et al. Greener RAN Operation Through Machine Learning. *IEEE Transactions On Network And Service Management*, 16(3), Sep. 2019.

[10] I. Donevski, G. Vallero and M. Marsan. Neural Networks for Cellular Base Station Switching. *IEEE INFOCOM WKSHPS*, 2019.

[11] Y. Gao, J. Chen, Z. Liu, B. Zhang, Y. Ke, and R. Liu. Machine Learning based Energy Saving Scheme in Wireless Access Networks. *IWCMC*, 2020.

[12] S. Zhang et al. Traffic Prediction Based Power Saving in Cellular Networks: A Machine Learning Method. *ACM SIGSPATIAL*, 2017.

[13] F. Xu et al. Big Data Driven Mobile Traffic Understanding and Forecasting: A Time Series Approach. *IEEE Transactions on Services Computing*, 9(5), Sep.2016.

[14] J. Perner et al. Network Energy Efficiency. *NGMN Alliance*, Oct.2021.

[15] J. Wu, Y. Zhang, M. Zukerman, and E. Yung. Energy-Efficient Base-Stations Sleep-Mode Techniques in Green Cellular Networks: A Survey. *IEEE Communication Surveys and Tutorials*, July 2015.

[16] B. Debaille et al. A Flexible and Future-Proof Power Model for Cellular Base Stations. *IEEE VTC*, May.2015.

[17] G. Barlacchi et al. A multi-source dataset of urban life in the city of Milan and the Province of Trentino. *Scientific Data*, Oct.2015.

[18] Z. Wang et al. Towards fairness in visual recognition: Effective strategies for bias mitigation. *IEEE CVPR*, 2020.

[19] T. Wang et al. Balanced datasets are not enough: Estimating and mitigating gender bias in deep image representations. *IEEE ICCV*, 2019.

[20] B. Wilson et al. Predictive Inequity in Object Detection. *arXiv*, 2019.

[21] 3GPP TR 28.813 . Study on new aspects of Energy Efficiency (EE) for 5G . *Technical Report*, 2021.

[22] https://orandownloadsweb.azurewebsites.net/specifications. *O-RAN Alliance Specifications*,2022.

Appendix A
Appendix: Model Features

The features are-

- Related load metrics known for the cell. For example, if we aim to predict the PRB usage, related load metrics include number of RRC connected users, number of active users etc.
- Metadata related to time and events, like whether a particular day is a holiday/special event, has a special weather pattern, time of the day or day of the week etc. Some of these features like time of the day along with past load or related metrics.
- First, second and even higher order moments of features (including max. and min. values) and load time series of the cell for example, mean, variance or higher order moments of the load could be used. The moments can be specific to particular times of the day of interest.
- Cell specific configuration information for cell, for example, bandwidth, center frequency, number of antennas etc.
- Finally, other similar load and feature time series of neighboring cells/group of cells in the network.