ISSN No:-2456-2165

Network Congestion Control

Prof Aarti Sawant, Yash Kumar, Shreya Badia, Electronics and Tele-Communication Engineering Department, Bharati Vidyapeeth (Deemed to be University) College Of Engineering, Pune

Abstract:- An optimum machine learning model that used cell tower statistics such as usage, customer count, etc., was instructed to project the kind of congestion that might occur. The accuracy of the model was appreciable and proper measures were taken to make it robust. As per further analysis carried out with respect to all possible algorithms like linear regression, support vector machine and neural network its found that the major factors causing congestions were byte usage and subscribers. Hence, vendors should look for beefing up their hardware's to serve more subscribers at the same time with increased byte rate. Also, in case of congestion, they can come up with a scheme to prioritize network traffic i.e., giving critical bytes usage like communication more priority over less critical bytes usage. . .

Keywords:- *Linear Regression, Machine Learning Model Support Vector Machine ,Neural Network, Congestion.*

I. INTRODUCTION

In the context of telecommunications industry, one of the most important issues that industry faces is network congestion. It has been shown that congestion, even if for smaller durations, has a negative impact on customer loyalty, especially in price sensitive markets. To solve this problem effectively, it becomes imperative for firms to be able to forecast congestion in advance and take dynamic actions. In this competition, you are required to train machine learning models that use cell tower statistics such as usage, customer count, etc., . We are providing a subset of original dataset, while also randomizing/masking some values to avoid leakage of proprietary information. Hence, this dataset only has sample data for December, 2018 transactions. Some fields in the current dataset are anonymized to avoid data leakage; while the usage data has also been anonymously scaled and randomized by a single/constant factor. The given dataset comprised of 26 different bytes features which had a skewed distribution and thus, created inefficiency problems for the machine learning algorithms. Although, after taking log of these features, the skewness was removed and the new features were used to build the algorithm. As we can see, the correlation between the dependent variable and independent variables is very low in the initial dataset provided, but increases to a significant number after taking log of the features which we can see in this Figure 1.

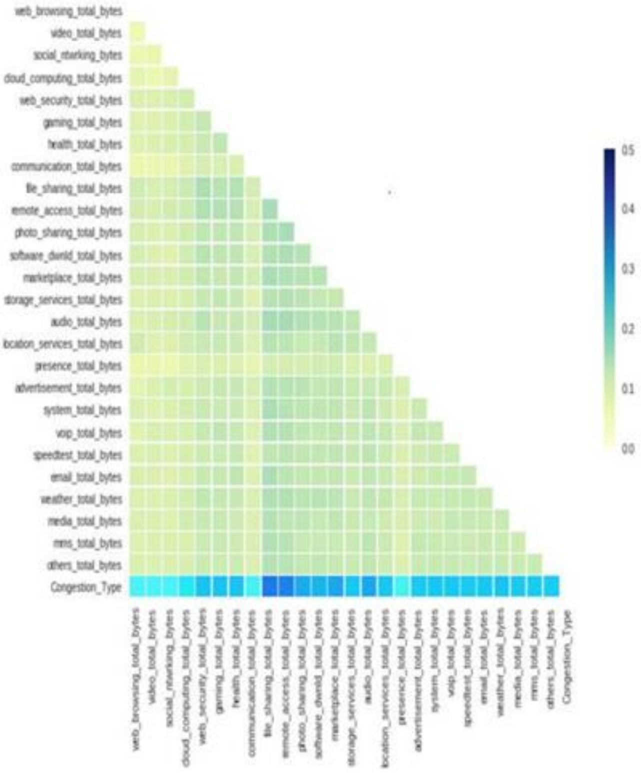

Fig1:Graph between congestion type and different bytes

When a linear fit of Congestion Type against Different Bytes was performed, an appropriate transformation that is log of different Bytes significantly improved the correlation because the original variables were positively skewed so taking **LOG** led to the Gaussian distribution and thus made them relatively more linear with Congestion Type. The nature of the Gaussian distribution often leads to extremely efficient gradient flow in the algorithms, which in turn leads to better optimization and performance. It can be seen that there is no significant correlation between different data bytes therefore none of the data features is dropped while training the model. The ran vendor column is a categorical feature therefore it is One-hot encoded to give better result. Binary "Weekend" feature is created from date feature to stimulate the model for detecting weekends,Par_hour is divided into 3 parts of day which is further One-hot encoded to "Day", "Night", "Work".

Models

The given problem statement is a multi-class classification problem and thus classification models such as Logistic Regression, Cart, Neural Networks and their ensemble forms were used while approaching the problem. Every model performed in a different way by correctly classifying certain sections of data while misclassifying the others. Thus, the final model was a maximum built voter classifier.

1.) Logistic Regression

The goal of logistic regression is to provide optimum estimates of Prob(Y=1|X). The mathematical equation can be described by Y= a + bx.

Test Accuracy: 0.7615

MCC: 0.6884

2.) Support Vector Machine

Support vector machines allow the use of kernels which can be engineered according to the problem. The regularization parameter helps in avoiding over fitting. Support Vector Machine with Linear kernel was found to be working poorly on the data. But after preprocessing the data, it was hypothesized that Radial basis kernel may give a better result, so it was employed and was proved to be a very efficient model

3.) CART

Cart implicitly performs variable screening and works well even in the presence of a nonlinear relationship. It uses information gained at each node to split up into further branches to predict the correct classes. However, for our dataset CART was giving a very large window of overfitting., Train Accuracy: 0.835

Test Accuracy: 0.5901

MCC: 0.4530

4.)Neural Network

Neural network is an algorithm which is very useful in inferring unseen relationships between the variables and doesn’t impose any restrictions on the input data type. A Neural network with two hidden layers was used while making the multiclass-classification model. But since making a deep neural network requires a lot of data and computation power, the approach was not restricted to it, yet it was not completely discarded. Therefore, a shallow neural network was employed achieving the required goal. (Input features, 200, 100, 4)

Neural network is a machine learning algorithm which is very useful in inferring unseen relationships between the variables and doesn’t impose any restrictions on the input data type. A Neural network with two hidden layers was used while making the multiclass-classification model. But since making a deep neural network requires a lot of data and computation power, the approach was not restricted to it, yet it was not completely discarded. Therefore, a shallow neural network was employed achieving the required goal. It is given by the formula: -

5.) Random forest

Random Forest is an organizable machine learning algorithm that develops a humongous number of irregular decision trees for scattering sets of variables. It adds additional arbitrary to the model while the trees kept growing. Instead of looking for the most valuable feature while scattering a node, it looks for the greatest feature among a unknown subset of features. Random forests are prone to over fitting, thus the parameter - number of trees, must be carefully tuned.

Train Accuracy: 0.866

Test Accuracy: 0.7831

MCC Metric: 0.711

6.) XGBOOST

XGBoost penalizes the complexity, which was not common for additive tree models and thus leads to optimal solution faster than other methods.

Train Accuracy: 0.82574

Test Accuracy: 0.8028

Rationale behind zeroing algorithm

The aforementioned reasons provide enough impetus to Ensemble the results of individual algorithms to give the final result.

Different combinations of individual algorithms were tried on the validation data and the set of algorithms (SVC(RBF) and XGBoost) giving best performance was chosen.

Feature Engineering

The following features were engineered to further enhance the productivity of the model.

Calculating area covered by a Transmitting tower using Tilt angle

Tilt was also an important feature which we got to know after calculating area from tilt angle.

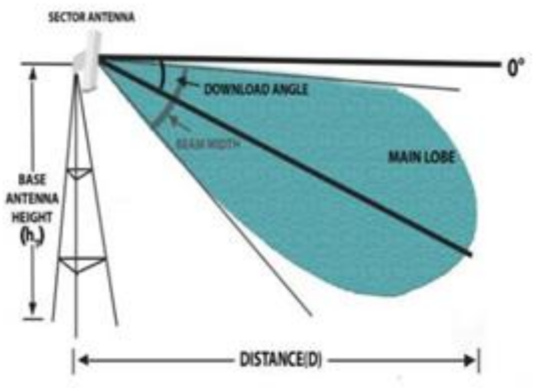

Fig 2:- Tilt angle of tower

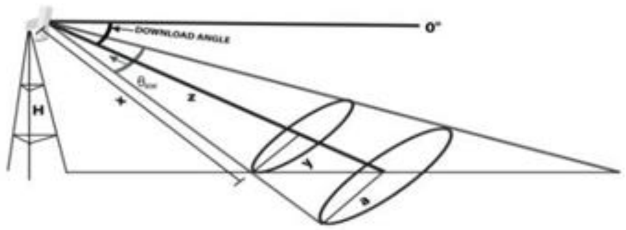

Fig 3:- Elliptical area covered by tower

$$Inner\ Radius = \frac{h_T}{\tan\left(A_{DT} + \frac{\theta_{bw}}{2}\right)}$$

Adt = Antenna down tilt in degrees

Ht = height of the transmitting antenna

Router = Outer radius of coverage

Rinner = Inner radius of coverage

BW = Beam Width

Using the given tilt feature and taking tower height to be 300ft, we found out inner radius and outer radius of area covered on the ground. The area covered on the ground is elliptical in shape and we found out area of that ellipse

III. CONCLUSION

The table given below as show the comparison between the models: -

Model	Test Accuracy	Train Accuracy	MCC score
Svc	80.64%	80.00%	0.7335
Rf	99.96%	79.10%	0.72
Xg boost	82.55%	80.665	0.74
ANN		79.12%	0.72
SVC(linear)	76.64	76.48	0.68

Table 1:-Table descripting the comparison of the models.

Important outcomes from the table were:-

1. **Rf model has the maximum test accuracy**.
2. **Xg boost has the highest train accuracy.**

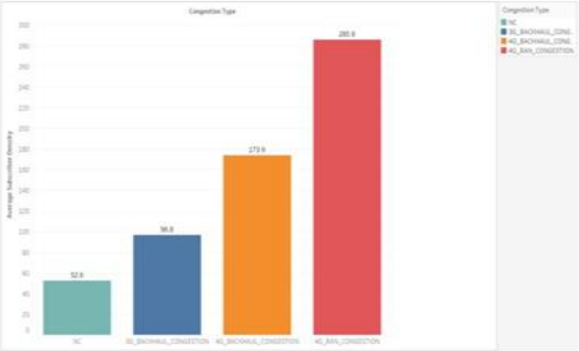

Fig 4:- Graph to find the relationship between types of congestion and the average subscriber density

As types of congestions were plotted against the average of their respective bytes used per day, a trend of increased byte usage for particular type of congestions was observed. While congestion was characterized by lowest level of average bytes usage (around 11,000 bytes), progressively higher level of average bytes usage was observed in 3G Backhaul (22,000 bytes), 4G Backhaul (40,000 bytes) and 4G RAN (70,000 bytes) congestions respectively. It means that no congestion will occur for lower byte usage and with further increment in byte usage, 3G Backhaul congestion will occur first, followed by 4G Backhaul congestion and then 4G RAN congestion no.

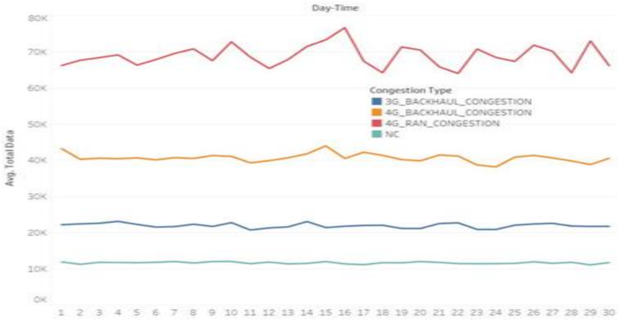

Fig 5:- graph showing the relationship between avg total data and no of days in the month

4. **A regression model was used to validate this observation**. The accuracy obtained was 77.25%.
5. **The probability of occurrence of congestions in a cell tower with respect to total bytes used via that cell tower was calculated**. As total bytes used via cell towers increased further, the probability of No Congestion and 3G Backhaul congestion decreased. With further increment in total bytes, probability of occurrence of 4G RAN congestion increased exponentially while the same for 3G and 4G Backhaul congestions decreased exponentially. As total usage exceeded 100,000 bytes, only 4G RAN and 4G Backhaul congestions were probable to occur with an initial probability of 0.77 and 0.23 respectively.

5.) Relationship between Congestion types

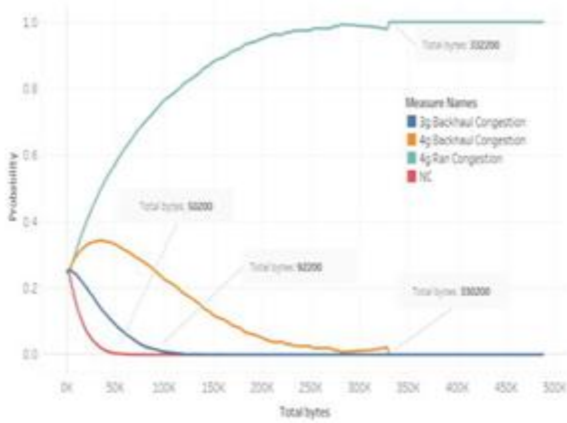

Fig 6:- Graph showing the dependency of probability vs total bytes

Assuming Uniform distribution of subscribers over the cell range, subscriber density was calculated by dividing number of subscribers by cell range. It was expected that higher subscriber density would lead to more frequent occurrences of congestions and it's evident from the average subscriber density for each congestion type. While lesser subscriber density helps in avoiding congestions, subsequent increment in subscriber density causes congestions.

6.) **The average byte usage over 30 days of the month has been plotted**.

While the byte usage didn't vary much for the first 11 days, it abruptly increased on Day 14 and 15. Afterwards, it went back to normal level only to see a steep rise on day

A final hike in byte consumption was seen on Day 25 and Day 26. These sudden hikes on particular days suggest the occurrence of events of greater importance on those specific days. These events can be festivals, sporting events, political events, ecommerce sales, releasing of new episodes of a TV series, viral news etc. The rise in byte consumption on Day 25 and Day 26 can be attributed to Christmas. On other days, it's due to other major happenings.

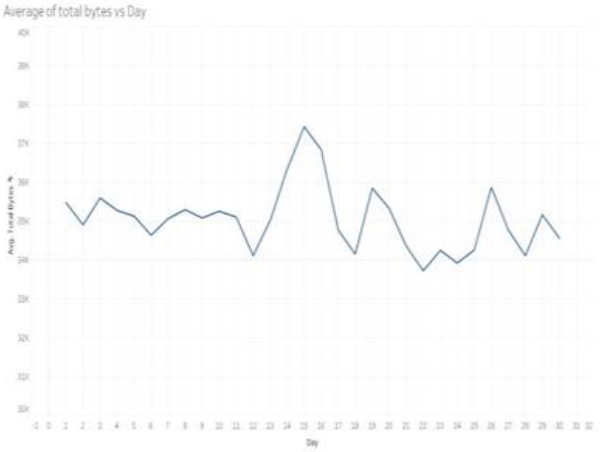

Figure 7:- Average of total bytes vs day

ACKNOWLEDGEMENT

This work is supported by the department of electronics and telecommunication Bharati Vidyapeeth (to be deemed University) College Of Engineering Pune

REFERENCES

[1]. https://www.computer.org/csdl/journal/sc/2019/05/07752911/13rRUwcS1wb
[2]. https://www.computer.org/csdl/journal/sc/2019/05/07
[3]. https://www.computer.org/csdl/magazine/ic/2019/05/08935577/1fPUeiCToMo
[4]. https://www.computer.org/csdl/proceedings-article/fpl/2018/851700a427/17D45WrVgc p
[5]. https://www.computer.org/csdl/proceedings-article/iciev%20&%20icivpr/2018/08640985/17PYEjTJ9C2
[6]. Intelligent Reinforcement-learning-based Network Management, draft-kim-nmrg-rl- 05. 2019. [Online]. Available: https://tools.ietf.org/html/draft-kim-nmrg-rl-05
[7]. Intelligent Reasoning on External Events for Network Management, draft-pedro-nmrg-intelligent-reasoning-00." 2019. [Online]. Available: https://tools.ietf.org/html/draft-pedro-nmrg-intelligent-reasoning-00
[8]. M. Wang, Y. Cui, X. Wang, S. Xiao, and J. Jiang, "Machine learning for networking: Workflow, advances and opportunities," *IEEE Netw.*, vol. 32, no. 2, pp. 92–99, Mar./Apr.2018.
[9]. K. Winstein and H. Balakrishnan, "Tcp ex machina: Computer-generated congestion control," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 43, no. 4, pp. 123–134, 2013.
[10]. F. Y. Yan, et al., "Pantheon: The training ground for internet congestion control research," in *Proc. USENIX Annu. Tech. Conf.*, 2018, pp. 731–743.
[11]. M. Dong, Q. Li, D. Zarchy, P. B. Godfrey, and M. Schapira, "PCC: Re-architecting congestion control for consistent high performance," in *Proc. 12th USENIX Conf. Networked Syst. Des. Implementation*, 2015.
[12]. M. Dong, et al., "PCC vivace: Online-learning congestion control," in *Proc. 15th USENIX Conf. Networked Syst. Des. Implementation*, 2018, pp. 343–356.
[13]. N. Jay, N. Rotman, B. Godfrey, M. Schapira, and A. Tamar, "A deep reinforcement learning perspective on Internet congestion control," in *Proc. 36th Int. Conf. Mach. Learn.*, 2019, vol. 97, pp. 3050–3059.
[14]. N. Jay, N. H. Rotman, P. Brighten Godfrey, M. Schapira, and A. Tamar, "Internet congestion control via deep reinforcement learning," in *Proc. Conf. Workshop Neural Inf. Process. Syst.*, 2018, pp. 1–10.
[15]. The network simulator—NS-3. 2001. [Online]. Available: https://www.nsnam.org/