International Journal of Innovative Research in Engineering & Management (IJIREM)
ISSN: 2350-0557, Volume-9, Issue-2, April 2022
https://doi.org/10.55524/ijirem.2022.9.2.2
Article ID IJIR-1180, Pages 8-16
www.ijirem.org

A Recommendation System to Analyse SDCCH Using Data Munging for Communication Network

Shahnawaz Shah[1], Ravinder P. Singh[3], Dr. Monika Mehra[2] and Dr. Jasmeen Gill[4]

[1]M. Tech, Department of Electronics & Communication Engineering, RIMT University, Punjab, India
[2]Technical Head, Department of Research, Innovation & Incubation, RIMT University, Punjab, India
[3]Head of Department, Department of Electronics & Communication Engineering, RIMT University, Punjab, India
[4]Innovation Consultant, Department of Research, Innovation & Incubation, RIMT University, Punjab, India

Correspondence should be addressed to Shahnawaz Shah; shahnawazshah500@gmail.com

ABSTRACT– The recommendation system in communication network plays an important role to recommend the end user for their choices, while analysing the browsing patterns it forecasts user's preferences for the thing. To facilitate required useable contents and information from the web data and to increase the system performance in hilly and mountain terrain areas where there are lots of communication network challenges, the data munging based on collaborative filtering technique is employed. To provide easy operations on e-commerce sites, online shopping sites and online income sites; the proposed recommendation system delivers better results in term of accuracy while considering high network congestion, channel traffic rate, SDCCH (Standalone Dedicated Control Channel) block calls sites, cell use, SDCCH dropping percentage, TCH (Traffic Channel) drop, handover, total cell derivatives, total site calls and losing sites.

KEYWORDS– Recommendation System, Data Munging, Collaborative Filtering Algorithm, SDCCH, TCH

I. INTRODUCTION

Recommendation system (commonly called a recommender or a recommendation engine) plays a crucial role to predict the end user's preferences and browsing patterns in web services such as online marketing, job web sites, income web sites, e-commerce sites, online shopping, online entertainment and online study. Such recommendation engines offer a number of benefits in terms of increase in the rate of return, sales and income of commercial sites by providing the ease of access to personalized or desired contents to the end users thereby enhancing the satisfaction of both customer and service providers. The most popular applications of machine learning based recommendation systems include Google, Goodreads, Amazon, Netflix, to name a few.

Research and solutions for telecommunications network management focus for many years on automation of operations that maintain complex networks in operational conditions and at the same time making them lucrative for network operators. Given the continually greater magnitude and complexity of the issue field combined with unique limitations, it is nearly difficult to have a completely automated management method. The NOCs still supervise and manage network operations in order to react to the large numbers of messages, mistakes, warnings and faults continually streaming from a controlled network. The NOC operators have the obligation to control and manage the network. In the past, Network Operators used expert systems or were highly relied on their operational staff's expert knowledge. Taking into account current growth of the size and complexity of the managed telecommunication network, the existing manual and semi-automated procedures are generally recognized and accepted [3].

In this research work, the applicability of recommendation systems as a way of helping NOC operators react properly to issues in the network that they manage are examined and an efficient model for recommendation of communication networks has been proposed and validated through various performance metrics. As an attempt to investigate to what degree it is feasible, using implicit feedback and advanced recommendation algorithms, to profile the clients.

The remaining article is organized as follows: overview of recommendation systems and its types are briefed in Section 2. The details of materials and methodology used for comparative analyses are provided in Section 3, results and discussions are presented in Section 4, and finally, the conclusions are summarized in Section 5.

II. OVERVIEW

A. Recommendation System

A recommendation engine usually analyses the data using several algorithms and suggests consumers the most relevant stuff. The general operational flow of a recommendation system can be concluded through four main steps, namely, data collection, data storage, data filtering and data analysis. This is demonstrated in figure1where the typical recommendation system is shown.

- **Data Collection –** The first and most important stage towards the construction of the recommendation motor. There are two ways to acquire data: explicit and implicit [2]. Explicit data are information deliberately submitted by users, e.g. film ratings. Implicit data is information which is not deliberately supplied but is collected from data streams accessible such as search history, clicks, order history, etc.

- **Data Storage –** The data level determines how well the model suggestions can achieve. For example, the more ratings people make films, the better the suggestions for other users in a film recommendation system. The data type plays a crucial role in determining the storage type that must be employed. A normal SQL database may contain this storage [1].
- **Data Filtering –** After the data collection and storage it has to filter to get the important information necessary to build the final guidelines. Different algorithms assist us facilitate the filtration process.
- **Data Analysis –** At this step, analysis of data is performed so that association between the customer data be formed and mapping of the preferences be accomplished using appropriate algorithms such as machine learning algorithms or data mining techniques.

Taking in view the benefits of recommendation systems, many studies have been conducted in the past and hence, a number of recommendation techniques have been developed as well as utilized by various researchers as evident in figure 2. Recommendation systems are broadly classified into three types: Content-based filtering, Collaborative filtering and Hybrid filtering techniques. Content-based filtering algorithm is based on a description of the element and a profile of a user's preferences. These algorithms try to recommend items that are similar to those that a user has liked in the past. Collaborative filtering is based on collecting and analyzing a large amount of information about the behaviour, activities, and preferences of users and the predicting what a user likes based on the similarity of the user to other users. The hybrid recommendation algorithm combines collaborative filtering and content-based filtering. In some cases, hybrid approaches can be more effective.

User-based filtering is based on history of users and similarity between them from their purchase histories for example. But, Item-based recommendations are based on content based similarity. Like, "how many times few items are bought together". Next time, most frequent of these purchases will be recommended together. In case of collaborative filtering "User Behaviour" is cashed in for recommending items. These recommendations can be generated with user-user similarity or on the basis of item-item similarity. And on the basis of this similarity measure the suggestions are provided to user. However, major problem is how to recommend effectively in the absence of user data. The problem of generating recommendations gets transformed into a clustering like problem. Where the similarity measure is based on how close two items are, while generating recommendations.

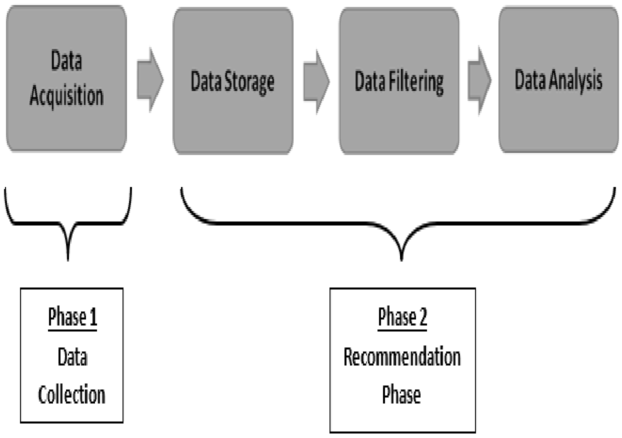

Figure 1: Flow Diagram of a Recommendation System

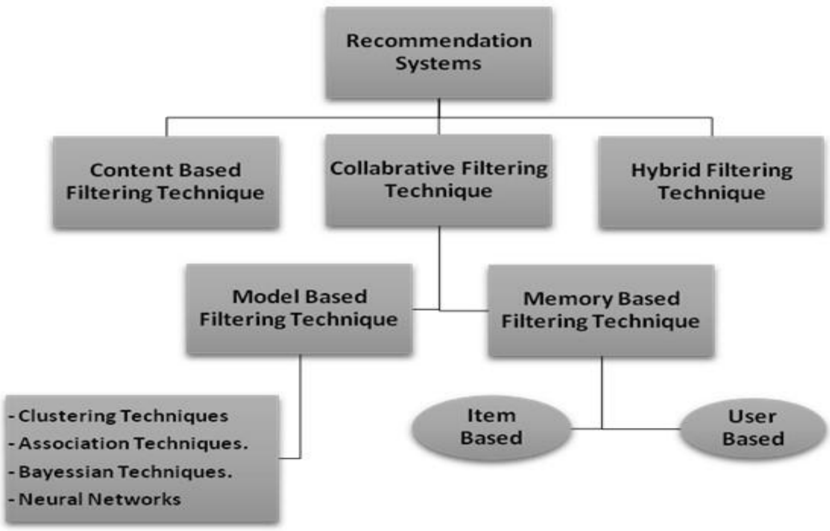

Figure 2: Types of Recommendation Systems

The measure for generating recommendation will be on the basis of similarity of two items like vector distance between these items. Hence, the popularity of cluster based collaborative filtering is justified from the aforementioned context. As a result, the proposed recommendation system for communication networks utilizes the cluster based collaborative filtering technique to resolve the issues of low bandwidth availability, high congestion or network down like situations.

III. MATERIALS AND METHODS

While designing the proposed recommendation engine for communication networks, there were a number of issues to be resolved. These are:

- Figure out which of the sites are important, which of the sites provide the highest revenue.
- Figure out the sites that usually remain down (non-working sites).
- Figure out which of the sites have faulty hardware.
- Determine the traffic sites and traffic rate per channel.
- Determine various site parameters, such as, SDCCH block calls, cell utilization, SDCCH drop percentage, total calls in site, TCH drop, total cell derivatives and reverse generating sites.

For resolving all these issues, the proposed system has been designed to work in two phases: data collection phase and recommendation phase. In the first phase, data in the form of customer's previous behaviour/preferences are collected and then these data are stored, filtered and analysed to suggest recommendations in the recommendation phase.

B. Data Acquisition

The proposed system utilized BSNL Telecommunications dataset for the month of Jan 2G Ericsson. The dataset was for one-month in the Excel sheet consisting initially 11564 rows and 36columns with information regarding different parameters such as Cell Peak hour, TCH Traffic, SDCCH Block calls etc., of 359 sites in Srinagar.
The dataset considered in this research work is from the perspective NOCs who are assigned the task of resolving the problems faced by the BSNL customers. Consequently, the whole purpose is to assist particularly the NOCs in making recommendations according to customers' queries.

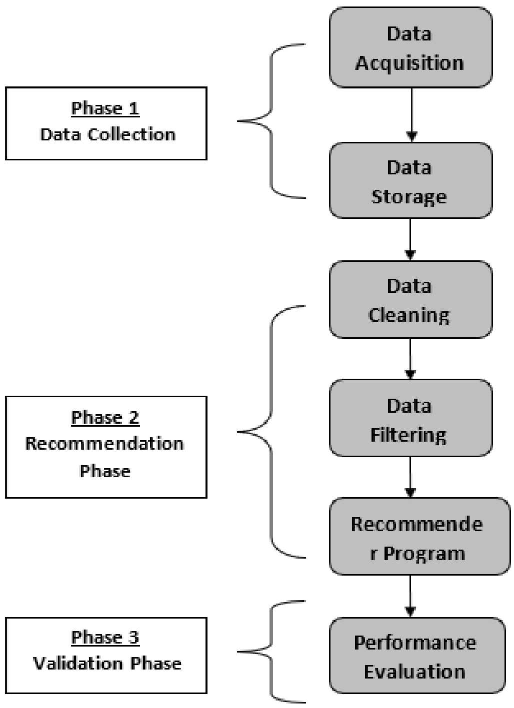

Figure 3: Collaborative Filtering based Proposed Recommendation System for Communication Networks

C. Data Cleaning

The data package used was initially in the raw form and thus, required to be 'mung' first so to utilize it further in the model and various techniques be applied thereafter. These are basically pre-processing tasks performed prior to finding any association between data. Following operations were performed for cleaning the data and hence making them desirable for the next steps:

- Extra rows in the datasets are removed containing irrelevant data.
- Extra columns are deleted that are of no use to reach any desirable relation between the data.
- Removing an unexpected value in rows and columns.
- Deleting duplicate data that is repeated in the middle of the dataset.
- Deleting the dummy data in the data frame.
- Grouping of the similar data.

The deletion in the rows and columns has been performed on the basis of any irrelevant data in the cells for instance, any natural language code which does not act as a candidate solution for the customer query. One such example is removal of the phrase,'ll' in I'll, we'll, he'll etc. In other words, only keywords in the data would be considered for the recommendation model.

D. Data Filtering

With the data grouping operation performed at data cleaning step, the whole data have been available in a single column of the customer query. Now the data are available in the desired format ready for extracting associations and mappings through the well known filtering method, collaborative filtering. This technique has been applied into three stages: creating a user model, finding the closest set of neighbours, and making recommendations.
In general purpose recommender systems, the user-member rating matrix contains the ratings of users for a set of items. However in the current scenario this communication networks based application is specifically for the NOCs (user) who would be recommended possible solutions on the basis of customer queries. Therefore, the matrix has been framed consisting of customer queries and the possible solutions of their problems while operating in the BSNL networks. When the target customer wishes to find out a solution from the recommendation system, neighbouring customers with similar issues as that of the target customer are determined. Based on the assessment of the previous ratings of the satisfactory level (customer feedback) of neighbouring customers, a solution that might be an optimal solution for the customer's query is predicted. In other words, a solution to be recommended by the NOC to a customer is rated based on the feedback of neighbouring customers with similar network problems to the target customer's.

E. Recommender Program

At this step, actual recommendations are made by finding the similarity between the neighbouring customers and the target customer. So, the computational similarity method, which allows inferring the similarity between an active customer and the available customers, plays an important role in the process of predicting the rating of a recommender system. When ratings are explicitly presented, similarity can be easily determined using the Pearson's correlation coefficient (PCC) or Pearson's similarity metric (PSim), given that similar customers tend to rate a solution with similar rating points. However, in case ratings are not explicitly available, these are to be first extracted. In the proposed model, k-Means Clustering has been employed to find the similarity. Figure 4 demonstrates the clustering algorithm.

K-MEANS CLUSTERING (Jane and Dubes 1988)

1. Arbitrarily select k objects as initial cluster centers (m_1, m_2 ... m_k);
2. Calculate the distance between each object X_i and each cluster centre, and then assign each object to the nearest cluster.
3. Calculate the mean of objects in each cluster as the new cluster centers.
4. Repeat 2 & 3 until the criterion function E converged, return the new cluster centers.

Figure 4: Algorithmic Steps for k-Means Clustering Technique

F. Performance Evaluation

After the recommendations being predicted, the model has been validated. To evaluate the model, performance potential of recommendations has been assessed quantitatively. For this purpose, different performance measures are computed. There are ten parameters that have been used in the present research, as mentioned in table 1. Also, the table shows expected value of the parameter and the description of the performance parameter. The ten performance metrics are- Traffic (Erlang), Traffic volume, Channel signals, SDCCH Channel, SDCCH Call Drop, TCH drop rate of 100%, Call Setup Success Rate, Handover, Drop Calls and Busy hours.

Table 12: Details of Performance Parameters

Sr. no.	Performance Parameter	Expected Value of Measure	Description
1.	Traffic (Erlang)	Low value	The volume of data travelling over the network at a particular point in time is network traffic or data traffic. Network data is usually contained in network packets that supply network loading.[4] Traffic is measured in units of Erlang hours by means of the measurement of total labor, generally in excess of 24 hours, per resource or facility. It also covers the interaction between traffic-sensitive devices and the rate at which calls are made. It is the product of the average intensity of traffic and the research time period.
2.	Traffic volume	Low value	The traffic industry has three main sub-sectors: telecommunications (the biggest), telecommunications (the second largest) and wireless communications. The main function is distance communication.
3.	Channel signals	---	The air interface signaling channels are used for call setup, paging, call maintenance, and sync, etc. There are three kinds of signals: Broadcast, Common Control and Dedicated Channels.[6]
4.	SDCCH Channel	---	SDCCH is used to exchange communication signals between the mobile GSM and network (base station). That is used to establish a traffic connection for pure signaling purposes. The RACH base station sent by the mobile station is allotted to the SDCCH response from AGCH.
5.	SDCCH Call Drop	Low value	The SDCCH Call Drop Rate specifies a call drop chance while the SDCCH occupies the MS. One of the accessibilities KPIs is the Call drop rate of SDCCH. The KPI shows signaling channels' seizure status. If the value of this KPI is high, it will severely effect user experience.[7] SDCCH call fall rate = (SDCCH Call drops + successful channel transfer SDCCH seizure) SDCCH call drop rate
6.	TCH drop rate of 100%	Low value	TCH drop (or drop-down call) can be widely categorized into subclasses: A. Link degradation (uplink and downlink) either deterioration of the signal force which fails at or below the sensitivity of the base station (about -110 dBm) or the sensitivity of the mobile station (about -104 dBm)

			B. Excess TA (TA>63 or high TA excess trajectory imbalance. The rate of drop-off from SDDCH: The SDCCH rate of call drop shows the likelihood of call drop while the MS occupies the SDCCH.
7.	Call Setup Success Rate	High value	The Call Setup Success Rate (CSSR) is the proportion of a call which makes a link to the numbers (due to various reasons not all call attempts end with a connection to the dialed number).
8.	Handover	Low value	Handover is a control mechanism started when mobile devices migrate from their current cell to the nearby cell. A mobile phone user is always moving. In such a case, especially if the user is using the phone, the mobile connection should also be maintained. These connection transfer from one cell to another should be rapid and so users do not know that a transfer has occurred.
9.	Drop Calls	Low value	The dropped rate is the percent of the telephone call that was stopped before the speakers ended their conversation tone for technical reasons and before one of them stopped (dropped call). Usually, this fraction is measured in a proportion of calls. A call attempt calls a call setup process that results in a connected call if successful. A related call may be discontinued because of a technical reason prior to the calling parties, which means telephone conversations before either of the parties has stopped. These calls are categorized as dropped calls.
10.	Busy hours	Low value	Busy hour as the peak of one hour on the day when the highest number of subscriber traffic is handled by leading outsourcing businesses (CPU, memory use, IOPS). The average intensity of traffic for this timeframe remains unbroken for 60 minutes. The busiest day in typical week is supposed to be, with the exception of holidays, weekends and events [8].

IV. RESULTS AND DISCUSSION

The tool used for the implementation of the proposed recommendation model was Python 3.7 and for GUI based data science application, Jupyter has been utilized. Following factors were identified and found while implementation-

- Sites in which voice traffic is more.
- Sites which are idol for a longtime.
- Sites with network problems.
- Sites with hardware problems.
- Sites with high data consumption.
- Sites with less data consumption i.e, sites where no data is used at all.
- Sites which generates maximum revenue.
- Sites at which hardware should be increased.

Since the primary task was to identify the largest revenue generating sites from the whole dataset as depicted in figure 1. Thus it gives a glimpse of a successful handover. 11564 rows and 36 columns were reduced to 11129 rows and 25 columns, RAW DATA: which was sorted using python into useful data, removed unnecessary data, dummy columns and errors.
RESULT 2: 10 Largest revenue generating sites which can be seen in Figure 2.

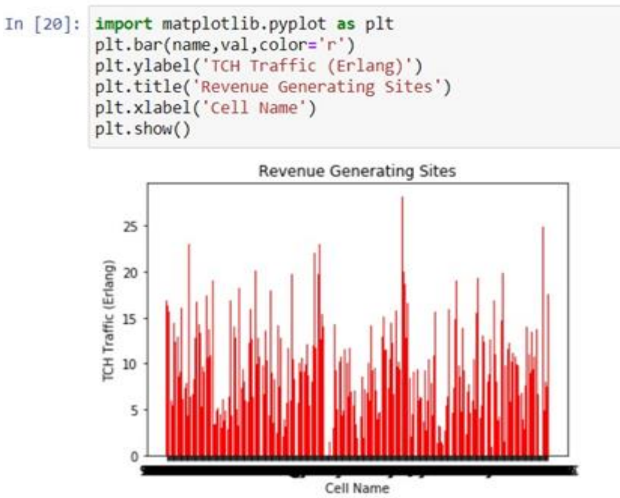

Figure 5: Largest revenue generating sites

RESULT 3: 5 Largest Sites with highest SDCCH Blockin3. This can be illustrated in figure.

```
In [25]: import matplotlib.pyplot as plt
         plt.bar(name,val,color='r')
         plt.ylabel('TCH Traffic (Erlang)')
         plt.title('Revenue Generating Sites')
         plt.xlabel('Cell Name')
         plt.show()
```

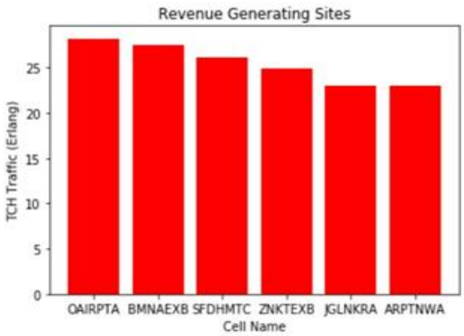

Figure 6: Largest Sites with highest SDCCH Blockin3

RESULT 3: The graph shows sites with highest TCH Call Drop Rate as shown in Figure 4. TCH Call Drop is related to retainability. It indicates the probability of call drops due to various reasons like hardware issue or handover issues.

```
In [51]: import matplotlib.pyplot as plt
         plt.bar(name,val,color='b')
         plt.ylabel('SDCCH Blocking')
         plt.title('Hardware Issue')
         plt.xlabel('Cell Name')
         plt.show()
```

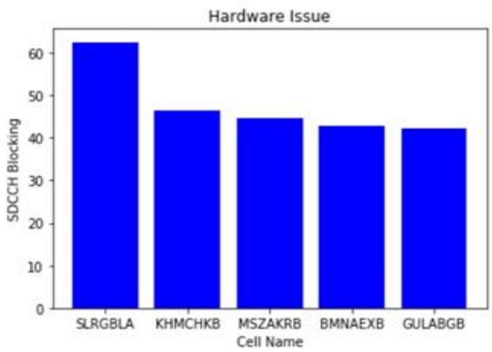

Figure 7: The graph shows sites with highest TCH Call Drop Rate

RESULT 4: The graph shows sites with highest TCH Call Drop Rate
TCH Call Drop is related to retain ability. It indicates the probability of call drops due to various reasons like hardware issue or handover issues.
Changing the data type to numeric.

```
In [45]: import matplotlib.pyplot as plt
         plt.bar(name,val,color='magenta')
         plt.ylabel('TCH Drop (%)')
         plt.title('TCH DROP')
         plt.xlabel('Cell Name')
         plt.show()
```

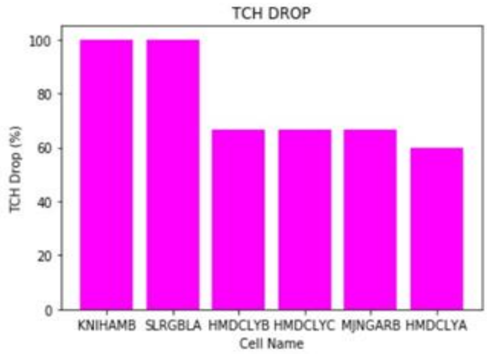

Figure 8: The graph shows sites with highest TCH Call Drop Rate-2

RESULT 5: The incoming HOSR (Hand Over Success Ratio) is the ratio of the number of successful handovers to the handover request as illustrated in Figure 5. The major purpose of handover is to guarantee call continuity, improve speech quality, reduce cross interference in the network and thus provide better services for mobile station subscribers.

```
In [61]: import matplotlib.pyplot as plt
         plt.bar(name,val,color='aqua')
         plt.ylabel('Incoming HO Success Rate')
         plt.title('Incoming HO Success Rate')
         plt.xlabel('Cell Name')
         plt.show()
```

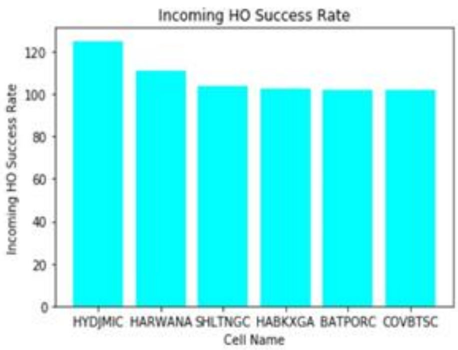

Figure 9: The incoming HOSR (Hand Over Success Ratio) is the ratio of the number of successful handovers to the handover request

RESULT 6: The OUTGOING HO SUCCESS RATES is the ratio of the number of successfully performed outgoing handover procedures to the number of attempted outgoing handover procedures to evaluate inter RAT outgoing performance request.

```
In [75]: import matplotlib.pyplot as plt
         plt.bar(name,val,color='brown')
         plt.ylabel('Outgoing HO Success Rate')
         plt.title('Outgoing HO Success Rate')
         plt.xlabel('Cell Name')
         plt.show()
```

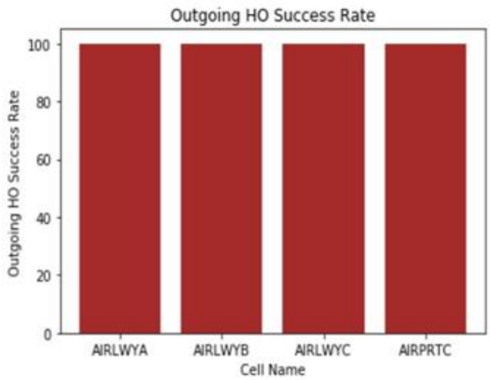

Figure 10: The Outgoing Ho Success Rates

RESULT 7: TCH Availability Rate is the ratio available TCH to the total available and unavailable TCH. This depicts rate of calls going under normal release that is not interrupted by SDCCH Drop and neither by Call Drop.

```
In [95]: import matplotlib.pyplot as plt
         plt.bar(name,val,color='yellow')
         plt.ylabel('TCH Availability Rate')
         plt.title('TCH Availability Rate')
         plt.xlabel('Cell Name')
         plt.show()
```

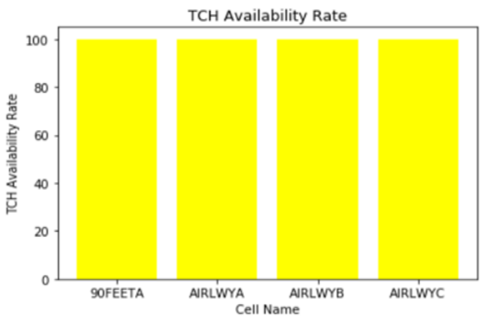

Figure 11: TCH Availability Rate is the ratio available TCH to the total available and unavailable TCH

RESULT 8: The sites where cell utilization is more will depict that all TCH channels are utilized.

```
In [116]: import matplotlib.pyplot as plt
          plt.bar(name,val,color='aqua')
          plt.ylabel('Cell Utilization')
          plt.title('Cell Utilization')
          plt.xlabel('Cell Name')
          plt.show()
```

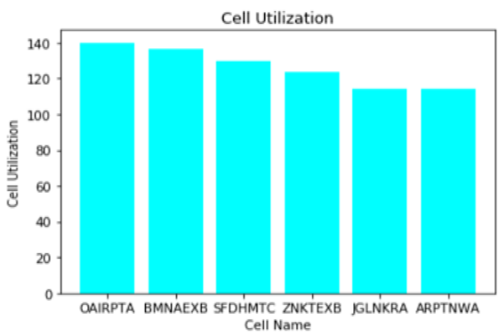

Figure 12: The sites where cell utilization is more will depict that all TCH channels are utilized

RESULTS GENERATED BY CLUSTERING

Steps for K-means clustering algorithm:

- First, import all the required libraries.
- Read the CSV file.
- We will be taking different variables from the data for generating various results"TCH Drop(%), TCH Traffic (Erlang), SDCCH Succ Calls(CMSESTAB), SDCCH Blocking, TCH Succ Calls(TMSESTB), TCH Dropped Connections(TNDROP)".
- We will be taking these variables and visualizing the data points between them.
- In this step, we have to choose the number of clusters (k) and we have to select random centroids for each cluster. We will pick 3 clusters and then elect random observations from the data as the centroids.
- Red dot represents the 3 centroid for each cluster.
- Our plots are generated with 3 clusters represented by 3 colors (blue, green, cyan) And 3 centroids (red dots).

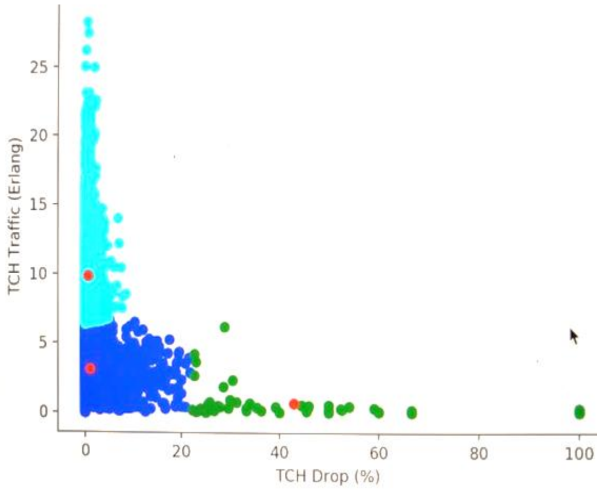

Figure 13: TCH Drop (%) and TCH Traffic (Erlang)

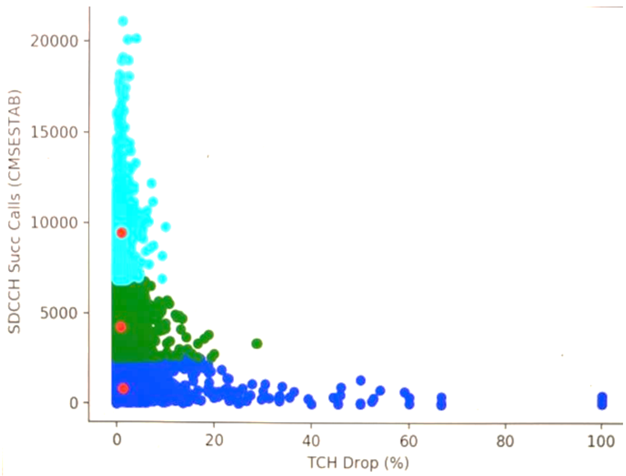

Figure 14: TCH Drop (%) and SDCHH Success Calls (CMSESTAB)

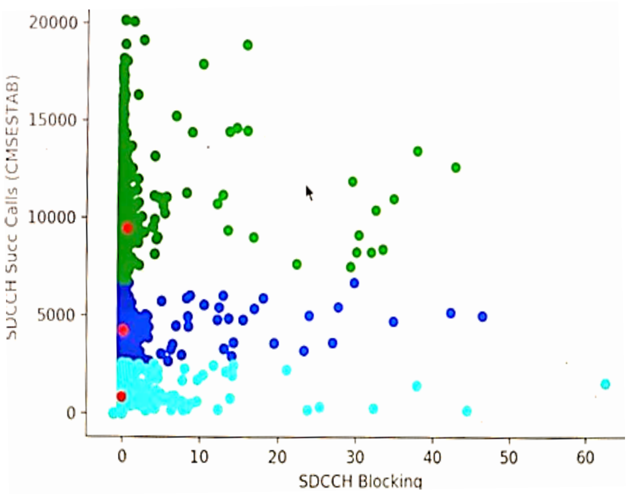

Figure 15: SDCCH Blocking and SDCCH Succ Calls (CMSESTAB)

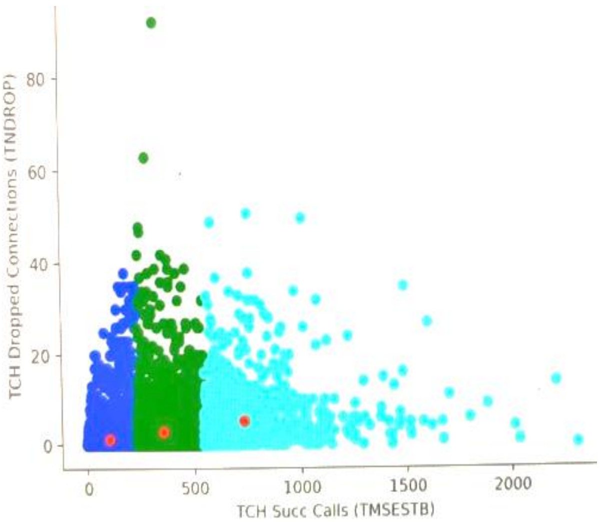

Figure 16: TCH Success Calls (TMSESTB) and TCH Dropped connections (TNDROP)

All these plots helped us in determining various parameters such as TCH Drop (%), TCH Traffic (Erlang), SDCCH Succ Calls, SDCCH Blocking, TCH success Calls, TCH Dropped Connections (TNDROP).

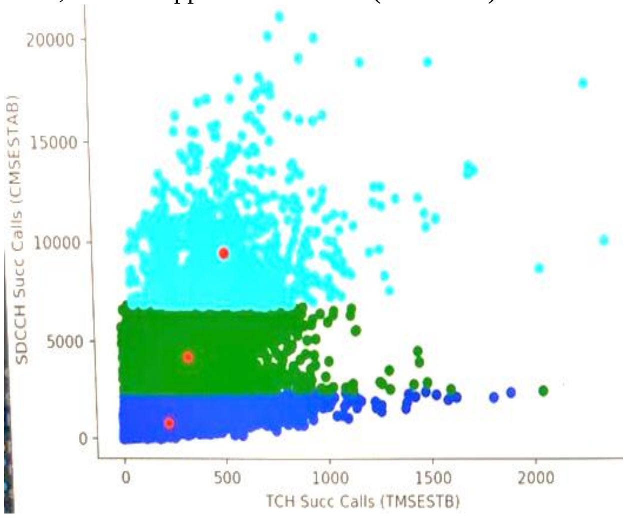

Figure 17: TCH Success Calls (TMSESTB) and SDCCH Success calls (CMSESTAB)

By determining these parameters, we can easily take a step to improve our network by eliminating all the faults whether it is software fault or hardware fault.

Table 2: Result table

TCH	Traffic	Results
Drop calls percentage	TCH Traffic	We found that in one of the clusters drop percentage is more when the traffic is less.
Drop call percentage	SDCHH Success calls	We found that the success rates are high in SDCHH, and call drop rates are less.
Success Calls	TCH Dropped Connections	TCH success calls increase, and RCH dropped connection will decrease.
Success calls	SDCHH Success Calls	Drop percentage decreases as success call increases.

We determined the top traffic sites and traffic rate per channel. We Analyzed sites with SDCCH block calls, Cell utilization, SDCCH drop percentage, Total calls in site, TCH drop, Total cell derivatives and Reverse generating sites

We Analyzed sites with SDCCH block calls, Cell utilization, SDCCH drop percentage, Total calls in site, TCH drop, Total cell derivatives and Reverse generating sites. While analyzing this data sets of telecom networking sites, we determined the sites that are in loss, the sites that provide highest revenue and the sites which are usually down (non-working sites).as shown in results 3

V. CONCLUSION

In this model, BSNL Telecommunications Operator data set has been used. With this data set I used pre-process approaches to understand the feasibility and examined numerous algorithms. The results reveal that, when we have historical data on prior customer adherence, our methodology is capable of inferring the best product or service to announce customers, suggesting that the algorithms may enhance suggestions inside a telecommunication provider.

VI. REFERENCES

[1] Mulizwa Soft, David Makadani Zulu, RuziveMazhandu : Department of Computer Science, University of Zambia, Lusaka, Zambia. Department of Mathematics,Universityof Leeds, Leeds,United Kingdom.

[2] R. Burke, “Hybrid recommender systems: survey and experiments,” User Modeling and User-Adapted Interaction, vol. 12, no. 4, pp. 331–370, 2002.View at: PublisherSite

[3] P. Lops, M. De Gemmis, and G. Semeraro, “Content-based recommender systems: state of the art and trends,” in Recommender Systems Handbook, pp. 73– 105, Springer, Boston, MA, USA, 2011.View at: Publisher Site | GoogleScholar

[4] M. D. Ekstrand, J. T. Riedl, and J. A. Konstan, “Collaborative filtering recommender systems,” Foundations and Trends in Human-Computer Interaction, vol. 4, no. 2, pp. 81–173, 2011.View at: Publisher Site | GoogleScholar

[5] J. Zhang, Q. Peng, S. Sun, and C. Liu, “Collaborative filtering recommendation algorithm based on user preference derived from item domain features,” Physica A: Statistical Mechanics and Its Applications, vol. 396, pp. 66–76, 2014.View at: Publisher Site | GoogleScholar.

[6] J. Zhang, Q. Peng, S. Sun, and C. Liu, “Collaborative filtering recommendation algorithm based on user preference derived from item domain features,” Physica A: Statistical Mechanics and Its Applications, vol. 396, pp. 66–76, 2014.View at: Publisher Site | GoogleScholar.

[7] DerrickMwiti Data Scientist | Writer | Mentor linkedin.com/in/mwitiderrick

[8] "Immunity: Knowing You're Secure". Archived from the original on 16February 2009