# Project Report

**June-July 2019**

Multi-class Railway Complaints Categorization Using NLP

**THE WAY FORWARD TO BETTER RAILWAYS**

**PREPARED BY** Anmol Saxena & Anubhav Singh

**PRESENTED ON** July 15, 2019

# DECLARATION

I, Anmol Saxena , hereby declare that the work presented in the project report “Multi-Class Railway Complaints Categorization using NLP” submitted as a part of curriculum after completion of summer internship-2019 is an authentic record of my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person or material which to a substantial extent has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgement has been made in the text.

Date: 15th JULY 2019

ANMOL SAXENA
B.TECH (CSE)

This is to certify the above statement made by the candidate is to the best my knowledge.

**(Mr. PITAMBER VERMA)**
**CPE/WA**

# DECLARATION

I, Anubhav Singh ,hereby declare that the work presented in the project report “Multi-Class Railway Complaints Categorization using NLP” submitted as a part of curriculum after completion of summer internship-2019 is an authentic record of my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person or material which to a substantial extent has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgement has been made in the text.

Date: 15th JULY 2019

ANUBHAV SINGH
B.TECH (CSE)

This is to certify the above statement made by the candidate is to the best my knowledge.

**(Mr. PITAMBER VERMA)**
**CPE/WA**

# CERTIFICATE

This is to certify that Anmol Saxena, a student of B.tech-Computer Science from BVCOE has worked on the project entitled ‘Multi-Class Railway Complaints Categorization Using NLP‘ during the period of 4th June 2019 to 15th July 2019 under our supervision and guidance. During this period her performance and conduct were excellent. We wish her all the success in her endeavors.

**(Mr. PITAMBER VERMA)**
**CPE/WA**

# CERTIFICATE

This is to certify that Anubhav Singh, a student of B.tech-Computer Science from BVCOE has worked on the project entitled ‘Multi-Class Railway Complaints Categorization Using NLP‘ during the period of 4th June 2019 to 15th July 2019 under our supervision and guidance. During this period his performance and conduct were excellent. We wish him all the success in his endeavors.

**(Mr. PITAMBER VERMA)**
**CPE/WA**

# ACKNOWLEDGMENT

We would like to convey our sincere gratitude to the registrar of Centre for Railways Information Systems (CRIS), New Delhi for providing me the golden opportunity to take summer internship programme in WA Division from 4th June 2019 to 15th July 2019.
It is truly a matter of great pleasure for me to express our sincere thanks and gratitude to Mr. Sudhendu J. Sinha, GM(WA) for his supervision and encouragement throughout this project. We are also obliged to Mr. Pitamber Verma, CPE(WA) for his kind support, constant supervision and encouragement during the training period.
We would also like to thank Mr. Shailendra Kumar Singh for his valuable suggestions and constant supervision throughout the course of the Internship.
In the end, We would like to thank our guide, Miss Pranjali Rathi for her constant support.
I would like to thank all the people in CRIS for their constant and unsung support which made this project happen.

Anmol Saxena
Anubhav Singh
B-Tech, CSE, 6th
Semester

# Contents



**IN THIS REPORT**

A big project involves overseeing a lot of moving parts, oftentimes from different people. To have a successful rollout, project managers rely on a well-crafted project plan to ensure objectives are met on time and on budget.

# Contents



**IN THIS REPORT**

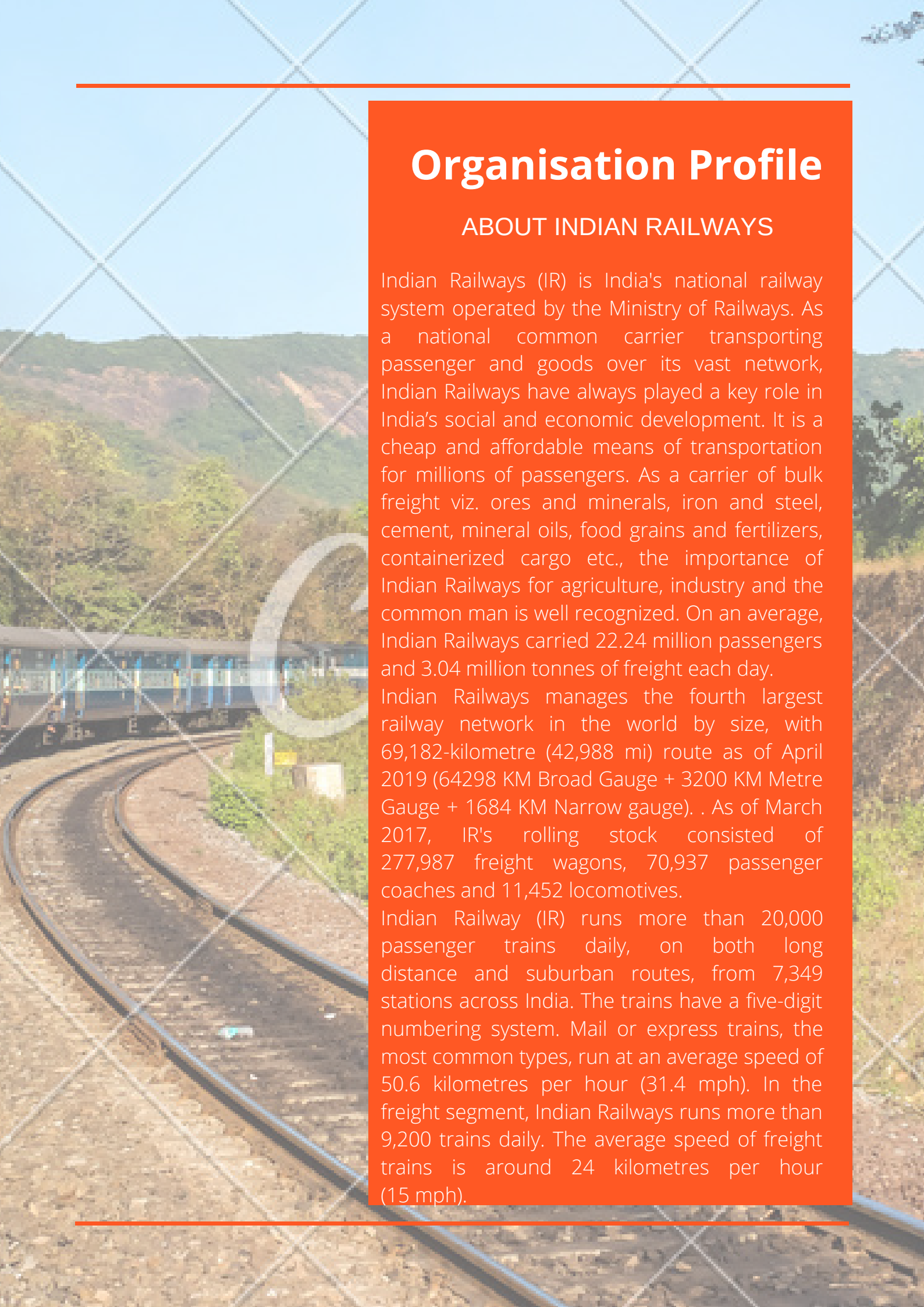
A big project involves overseeing a lot of moving parts, oftentimes from different people. To have a successful rollout, project managers rely on a well-crafted project plan to ensure objectives are met on time and on budget.

# Contents



**IN THIS REPORT**

A big project involves overseeing a lot of moving parts, oftentimes from different people. To have a successful rollout, project managers rely on a well-crafted project plan to ensure objectives are met on time and on budget.



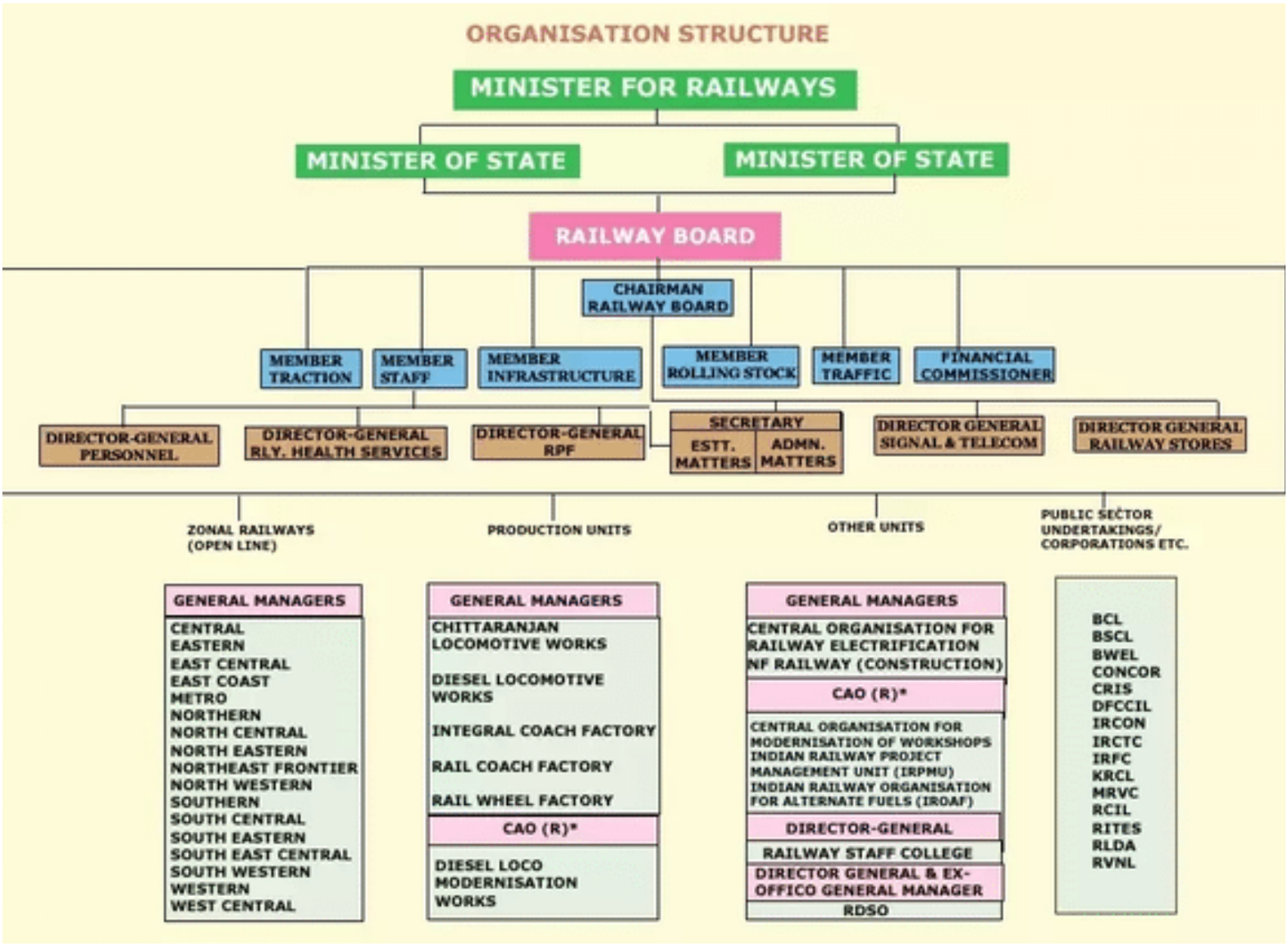
# Organisation Profile

## ABOUT INDIAN RAILWAYS

Indian Railways (IR) is India's national railway system operated by the Ministry of Railways. As a national common carrier transporting passenger and goods over its vast network, Indian Railways have always played a key role in India's social and economic development. It is a cheap and affordable means of transportation for millions of passengers. As a carrier of bulk freight viz. ores and minerals, iron and steel, cement, mineral oils, food grains and fertilizers, containerized cargo etc., the importance of Indian Railways for agriculture, industry and the common man is well recognized. On an average, Indian Railways carried 22.24 million passengers and 3.04 million tonnes of freight each day.

Indian Railways manages the fourth largest railway network in the world by size, with 69,182-kilometre (42,988 mi) route as of April 2019 (64298 KM Broad Gauge + 3200 KM Metre Gauge + 1684 KM Narrow gauge). . As of March 2017, IR's rolling stock consisted of 277,987 freight wagons, 70,937 passenger coaches and 11,452 locomotives.

Indian Railway (IR) runs more than 20,000 passenger trains daily, on both long distance and suburban routes, from 7,349 stations across India. The trains have a five-digit numbering system. Mail or express trains, the most common types, run at an average speed of 50.6 kilometres per hour (31.4 mph). In the freight segment, Indian Railways runs more than 9,200 trains daily. The average speed of freight trains is around 24 kilometres per hour (15 mph).


ORGANISATION STRUCTURE
MINISTER FOR RAILWAYS
MINISTER OF STATE
MINISTER OF STATE
RAILWAY BOARD
CHAIRMAN RAILWAY BOARD
MEMBER TRACTION
MEMBER STAFF
MEMBER INFRASTRUCTURE
MEMBER ROLLING STOCK
MEMBER TRAFFIC
FINANCIAL COMMISSIONER
DIRECTOR-GENERAL PERSONNEL
DIRECTOR-GENERAL RLY. HEALTH SERVICES
DIRECTOR-GENERAL RPF
SECRETARY
ESTT. MATTERS
ADMN. MATTERS
DIRECTOR GENERAL SIGNAL & TELECOM
DIRECTOR GENERAL RAILWAY STORES
ZONAL RAILWAYS (OPEN LINE)
PRODUCTION UNITS
OTHER UNITS
PUBLIC SECTOR UNDERTAKINGS/ CORPORATIONS ETC.
GENERAL MANAGERS
CENTRAL
EASTERN
EAST CENTRAL
EAST COAST
METRO
NORTHERN
NORTH CENTRAL
NORTH EASTERN
NORTHEAST FRONTIER
NORTH WESTERN
SOUTHERN
SOUTH CENTRAL
SOUTH EASTERN
SOUTH EAST CENTRAL
SOUTH WESTERN
WESTERN
WEST CENTRAL
GENERAL MANAGERS
CHITTARANJAN LOCOMOTIVE WORKS
DIESEL LOCOMOTIVE WORKS
INTEGRAL COACH FACTORY
RAIL COACH FACTORY
RAIL WHEEL FACTORY
CAO (R)*
DIESEL LOCO MODERNISATION WORKS
GENERAL MANAGERS
CENTRAL ORGANISATION FOR RAILWAY ELECTRIFICATION
NF RAILWAY (CONSTRUCTION)
CAO (R)*
CENTRAL ORGANISATION FOR MODERNISATION OF WORKSHOPS
INDIAN RAILWAY PROJECT MANAGEMENT UNIT (IRPMU)
INDIAN RAILWAY ORGANISATION FOR ALTERNATE FUELS (IROAF)
DIRECTOR-GENERAL
RAILWAY STAFF COLLEGE
DIRECTOR GENERAL & EX-OFFICO GENERAL MANAGER
RDSO
BCL
BSCL
BWEL
CONCOR
CRIS
DFCCIL
IRCON
IRCTC
IRFC
KRCL
MRVC
RCIL
RITES
RLDA
RVNL

# CRIS

ABOUT CENTRE FOR  RAILWAYS INFORMATION SYSTEMS(CRIS)


## ABOUT CRIS

(CRIS) is a dedicated, autonomous umbrella organisation which was set up by the Ministry of Railways in 1986. The organisation is responsible for all information technology-related activities on Indian Railways. In 1980s, when former prime minister Rajiv Gandhi and telecom engineer Sam Pitroda were preparing a blueprint for India's information and communication technology industry, the duo decided to keep the Indian Railways (IR) in its ambit too. Reason: IR was completely dependent on manual operations, often leading to delayed freight. So, in July 1986, a society—Centre for Railway Information Systems—was set up.

## PROJECTS TAKEN BY CRIS

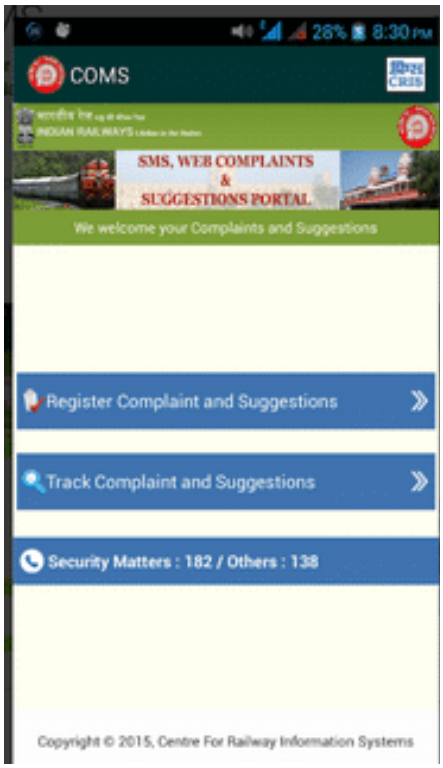
- Computerisation of the Freight Operations Information System
- Passenger Reservation System (PRS)
- Next Generation e-ticketing (NGeT)
- Computerisation of Indian Railways' Unreserved Ticketing System
- Workshop Information SystEm (WISE)
- Crew management

## IN A NUTSHELL

So, from being a society, now CRIS has become the backbone of the Indian Railways. It would be impossible for Indian Railways to run without the systems developed by CRIS, as CRIS has transformed Indian Railways from manual system to IT-enabled systems. Today, IT applications of CRIS fetches Indian Railways around ₹ 427 crore per day, while revenue from the Internet or e-commerce gateway is just ₹ 302 crore.

# ABSTRACT

Based on our studies, inclusive is what will keep this project afloat, making it an important factor to consider in planning.




## ABOUT

Due to increasing number of passengers in Indian Railways, there are huge number of complaints in front of Indian Railways coming every minute.To handle such a large amount of complaints require an innovative solution. To counter this issue, we with the use of Natural Language Processing are designing a model that classifies the complaint with the help of keywords in it into certain categories and sub-categories. Each category would have a unique priority and will be responded accordingly. Thus enabling Indian Railway at a better responsiveness towards complaints."

## DATA

This will be a Supervised Learning .We are training the machine on previous categorised data of complaints of CRIS(Centre For Railways Information Systems) having mainly three attributes:- Text,Complaint Category and Complaint sub-category..

## STEPS

- Steps
- Exploratory Data Analysis & Text Processing
- Labelling the data
- Featuring the data
- Dimensionality Reduction using Latent Semantic Analysis (LSA) which is a dimensionality reduction technique. LSA uses SVD or Singular Value Decomposition to reduce the number of dimensions and select the best ones.
- Model Selection and Evaluation
- Hyperparameter Tuning
- Prediction

# INTRODUCTION

Indian Railway has multiple ways to lodge complaints related to amenities, security, food services or any other complaints/suggestions. As Indian rail are used by wide verities of users from a poor man in general class to business man in AC class, everyone may face one or another problem during the travel. Consider this aspect in mind railway have time to time created a different channel of complaints system for make rail journey stress-free. Trains can be local, passenger, Rajadhani, shatabadi, mail, express, superfast, any type of train and same complaints system applied for all. Complaints can be lodged for a station amenities or problem during running train. There are many ways where user can register their complaints:

- Call/SMS – so a common man travelling in general class can lodge complaints
- Twitter – A Smartphone carrying travellers can lodge complain and escalate to highest level
- App/Website – So complaints can be lodged and tracked easily by travellers
- WhatsApp nos – so quickly share image/video of crime scene also ( only used by security team

## COMS

The Complaint Management System (COMS) portal consists of the following applications :- A mobile app based complaints and suggestions application (currently on the android platform). A web based complaints and suggestions application on URL http://www.coms.indianrailways.gov.in/. A SMS based complaints and suggestions application on the number 9717630982. A link to Centralised Public Grievance Redress and Monitoring System (CPGRAMS) will also be provided. It is one of the project of cris. Now, for betterment and fast services we are classifying the complaints into its subcategories by applying various algorithms and LSTM recurrent neural network models for text classification problems, for the sake of faster service assistance. The department who is responsible will immediately get notified and complaint gets resolved.

# Project Strategy

Complaints registered through different ways leads to huge amount of data. To manage and classify these huge railway data is a cubersome task, to divide them into categories and subcategories and which complaint belong under which department. Before getting the system to work on a complaint, a tough task is to find out which complaints needs how much attention. Software processes the information based on keywords to analyse the sentiments. Once analysed, the data are classified into categories and sub-categories and than based on the particular class the complaints get registered successfully.Therefore, through this classification the services get simulated.

Comlaints like medical assistance, cleanliness or police help are of higher priority. Others such as those about broken windows, catering issues and missing parcels are categorised as medium and low priority. The handling of complaints has been outsourced to a trained team that works 24x7 in shifts.

But not every compliant get reslved . Therefore categorization is necessary,we are revolutionising our grievance-redressal system. We follow every complaint closely and try to liquidate all the cases on daily basis.

## Different ways of Complaining

The railways receives complaints through several channels such as Facebook, Instagram, interactive voice response system (IVRS) and Centralized Public Grievance Redress And Monitoring System (CPGRAMs), besides Twitter.

Every day, the railways receives on average 200 complaints via CPGRAMS, 2,100 actionable tweets, 500 complaints on Facebook, 350 via COMS (railways-specific complaints management system that works through an Android app and mobile SMS) and 6,000 phone calls on number 138. It collects feedback from 1.2 lakh customers evey day through IVRS.

To make its complaint-redressal system more effective, the railways plan to integrate feedback received from different channels. The integrated customer complaint system will include Twitter, Facebook, Instagram, YouTube, and CPGRAMS.

# Our Key Goals

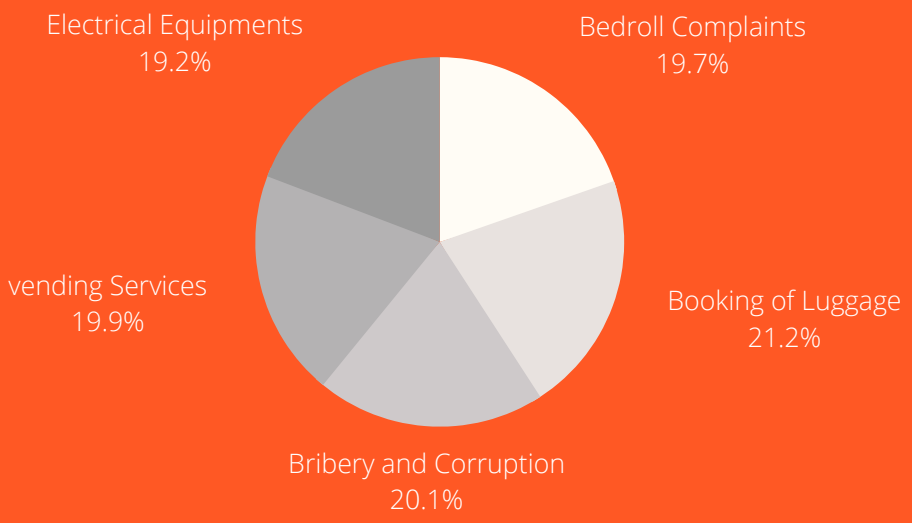
The System is dedicated to making its services accessible to sectors that need them the most.

## PURPOSE

The aim of this project  is to improve the quality of railways passengers. Passenger's satisfaction has always been the priority for Indian Railways and this system helps to make that happen.

## OBJECTIVES

· Providing a proper interface between passengers and railways admin.
· Acts as a feedback system which helps in improvement of railways  services.
· Providing structure to complaint redressal system of railways.




**95%**

Accuracy

The System should be consistent, accurate, correct , highly responsive, correct it should fulfill all the demands of the user.

# TEXT CLASSIFICATION

Text classification is the process of assigning tags or categories to text according to its content. It's one of the fundamental tasks in Natural Language Processing (NLP) with broad applications such as sentiment analysis, topic labeling, spam detection, and intent detection.

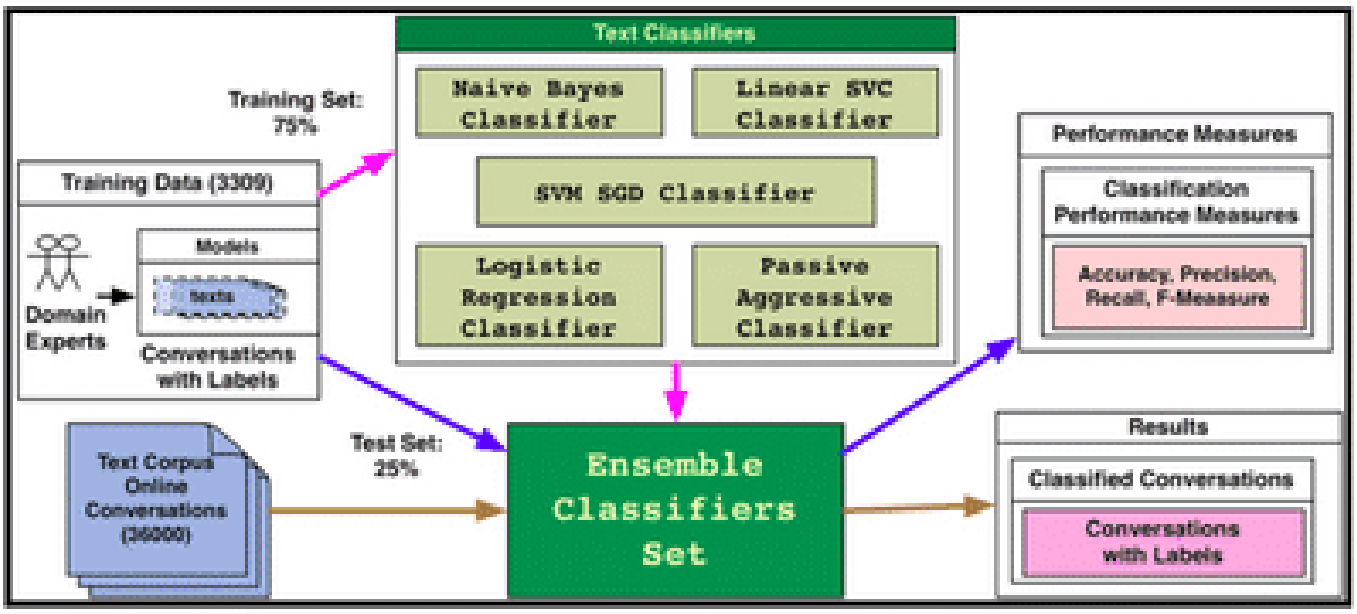
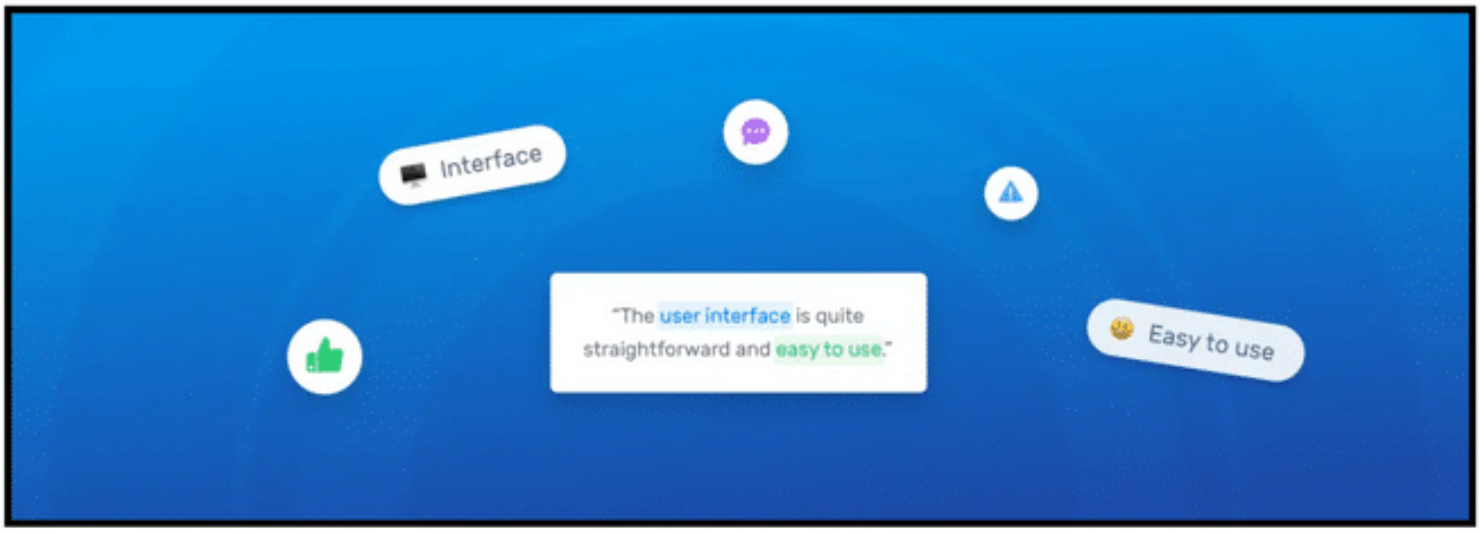


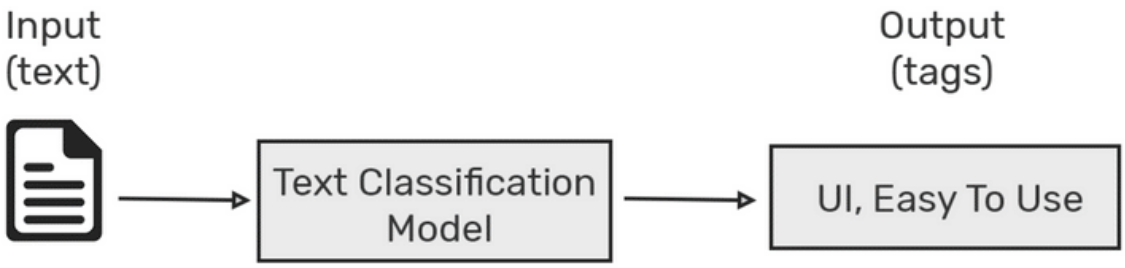

**FIG.1 TEXT CLASSIFICATION ARCHITECTURE**

Unstructured data in the form of text is everywhere: emails, chats, web pages, social media, support tickets, survey responses, and more. Text can be an extremely rich source of information, but extracting insights from it can be hard and time-consuming due to its unstructured nature. Businesses are turning to text classification for structuring text in a fast and cost-efficient way to enhance decision-making and automate processes.




*"The user interface is quite straightforward and easy to use."*

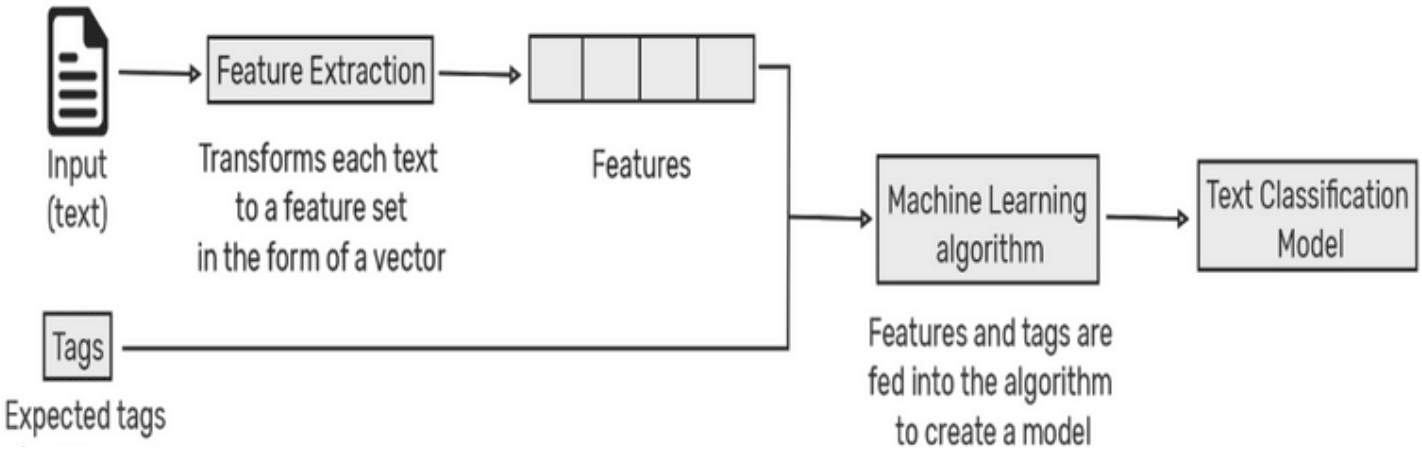
A classifier can take this text as an input, analyze its content, and then and automatically assign relevant tags, such as UI and Easy To Use that represent this text:




There are many approaches to automatic text classification, which can be grouped into three different types of systems:

- Rule-based systems
- Machine Learning based systems
- Hybrid systems

For example, if we have defined our dictionary to have the following words {This is, the, not, awesome, bad, basketball}, and we wanted to vectorize the text "This is awesome", we would have the following vector representation of that text: (1, 1, 0, 0, 1, 0, 0).Then, the machine learning algorithm is fed with training data that consists of pairs of feature sets (vectors for each text example) and tags (e.g. complaints, sub-complaints) to produce a classification model:




Once it's trained with enough training samples, the machine learning model can begin to make accurate predictions. The same feature extractor is used to transform unseen text to feature sets which can be fed into the classification model to get predictions on tags (e.g. complaints, suggestions on COMS app):

This will be a Supervised Learning .We are training the machine on previous categorised data of complaints of CRIS(Centre For Railways Information Systems) having mainly three attributes:-Text,Complaint Category and Complaint sub-category.

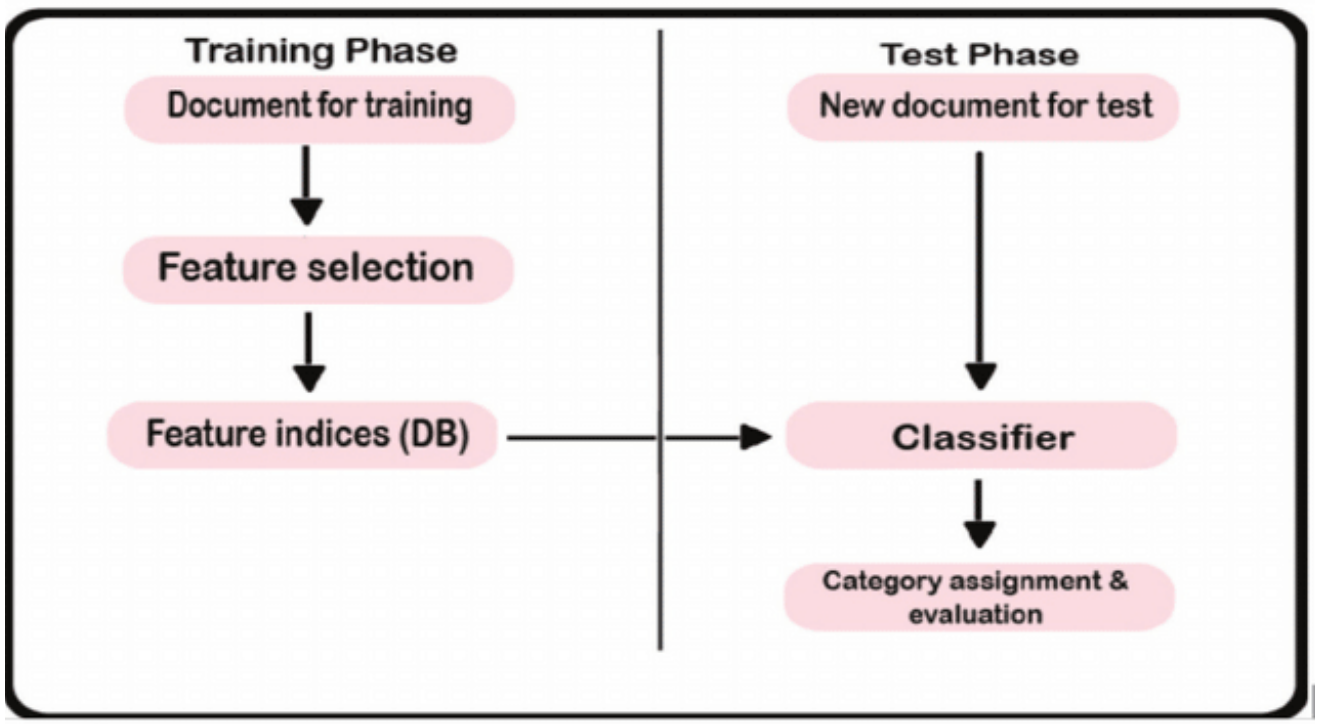



Fig. Data flow diagram of Text Categorization

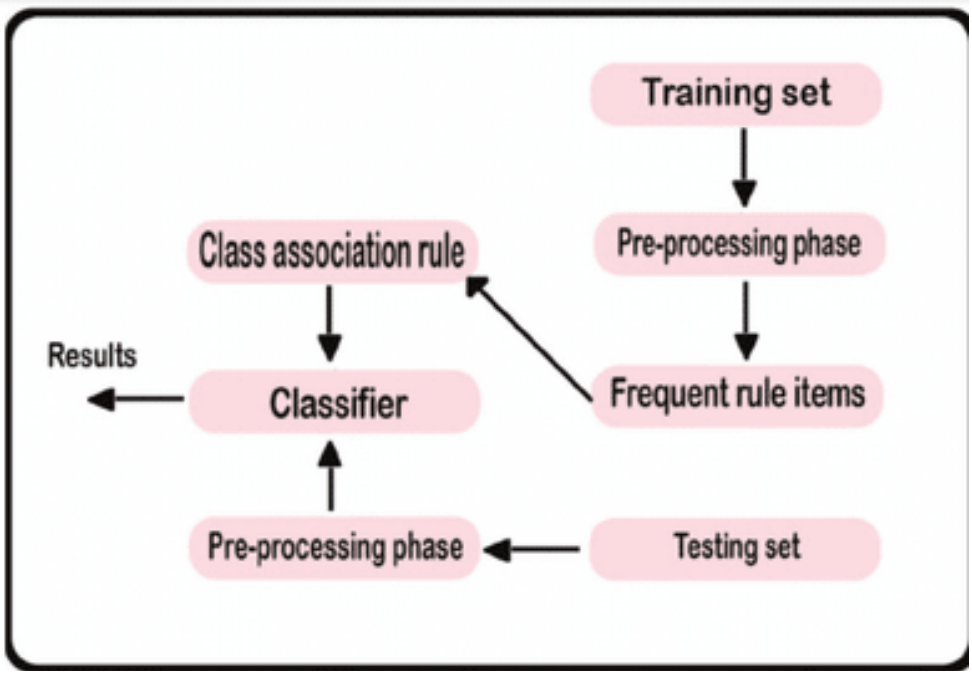



Fig: Data flow diagram of associative classification

# MULTI-CLASS TEXT CLASSIFICATION

**RAILWAYS COMPLAINTS DATA**

Text classification is a supervised learning technique so we need some labeled data to train our model. We have used these public complaints classification dataset. It's a manually labeled dataset of complaints register which fit into one of 14 classes: like: Misc. Cause, Malfunctioning of electrical Equipment, Maintenance/ Cleanliness of coaches,Punctuality etc. as given in the image below:

```
Malfunctioning of Electrical Equipment                44346
Maintenance / Cleanliness                             42085
Punctuality of Train                                  39848
Non availability of Water Sub                         22743
Catering and Vending Services                         17383
Unauthorised passengers in coaches                    13227
Reservation/Enquiry_Office Issues                     11998
Bedroll Complaints                                     9714
Improper behaviour of non railway/railway staff        7089
Booking of Luggage / Parcels / Goods                   6321
Feedback/Suggestions                                   2295
Thefts / Pilferages                                    1545
Bribery and corruption                                  882
Emergency Assistance                                    584
Name: Complaint, dtype: int64
```

# Available Dataset

We have the real time data of the complaints registered through the **COMS** app of **Indian Railways** in the form of textfile.

Let's visually inspect the json file:

# DATA EXTRACTION

Data was extracted from text data by removing unnecessary keywords such as ‘#SEPARATOR#’,’$’,double quotations, backslashes and then including them into separate files and then zipping them together into a single csv file.

```
import nltk
with open('smsdata.txt','r') as file:
    lines_in_files=file.read()
    print(type(lines_in_files))
    text2=lines_in_files.split('#SEPARATOR#')
```

```
sublist11=[]
sublist21=[]
sublist31=[]
for i in range(0,203470):
    sublist11.append(li[i].split('"$"')[0].split(":")[1])
    sublist21.append(li[i].split('"$"')[1].split(":")[1])
    print(li[i].split('"$"')[1].split(":")[1],i)
    sublist31.append(li[i].split('"$"')[2].split(":")[1])
```

# Exploratory Data Analysis & Text Processing

We have identified how many articles we have per class:

| | |
|---|---|
| Misc. Cause | 40507 |
| Malfunctioning of Electrical Equipment | 31067 |
| Maintenance / Cleanliness of coaches | 30956 |
| Punctuality of Train | 28000 |
| Non availability of Water | 16032 |
| Catering and Vending Services | 14351 |
| Unauthorised passengers in coaches | 8555 |
| Bedroll Complaints | 7201 |
| Booking of Luggage / Parcels / Goods | 5642 |

The problem is supervised text classification problem, and our goal is to investigate which supervised machine learning methods are best suited to solve it.

Given a new complaint comes in, we want to assign it to one of 14 categories. The classifier makes the assumption that each new complaint is assigned to one and only one category. This is multi-class text classification problemAll of the classes are perfectly balanced which is something we will almost never find in the wild so I will take a sub sample of the complaints and subcomplaints categories to make it imbalanced (i.e. more realistic).

I'll also hold out 5 articles from each category to use for predictions at the end to evaluate how well the classifiers did on unseen data which is the true test.

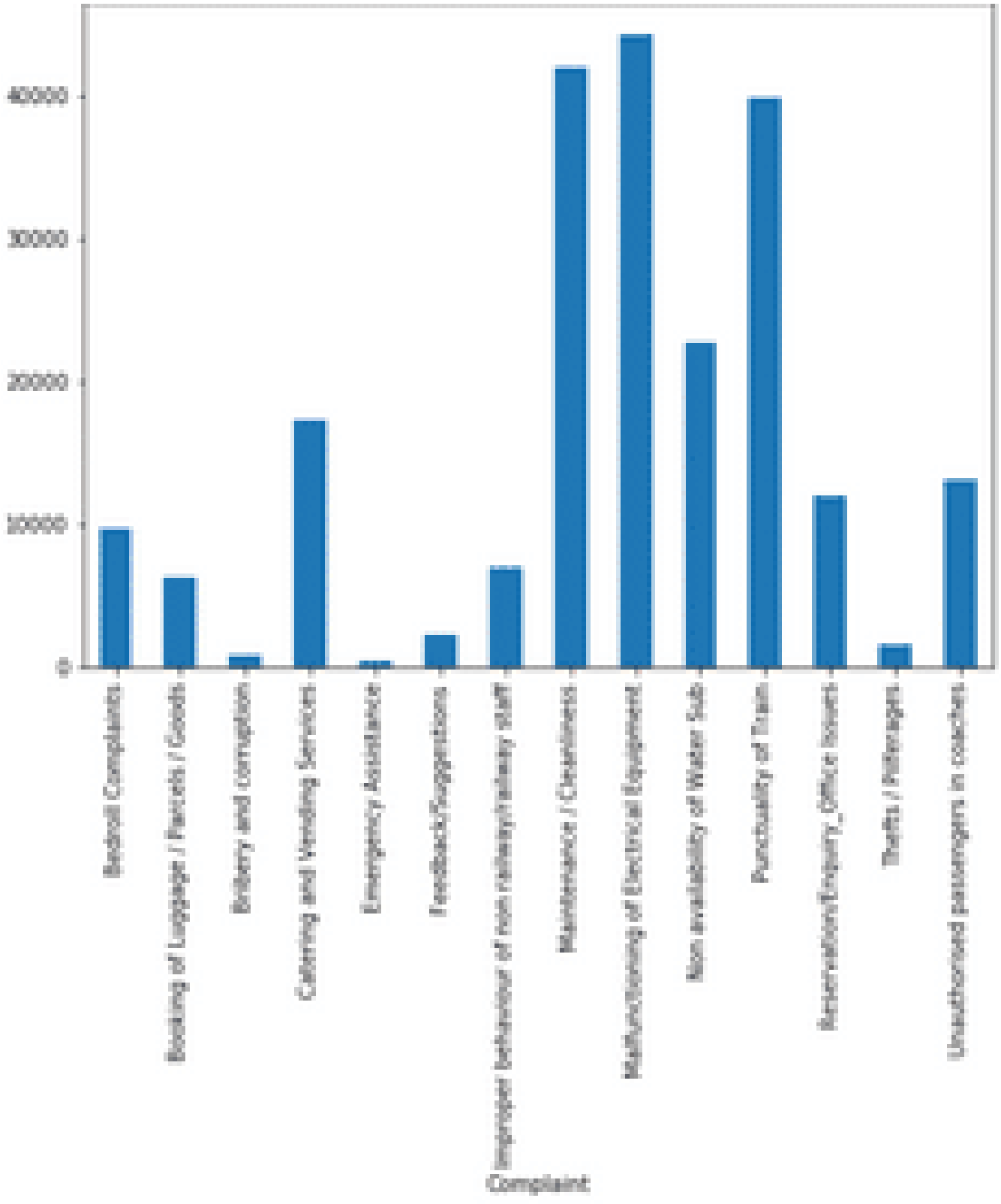
the text processing would be able to look at the most frequent words/bigrams. The processed text will also be what we use to create the features. The text will be tokenized, lower cased and lemmatized. It will have punctuation, numbers and stop words removed. The contractions will also be expanded out.

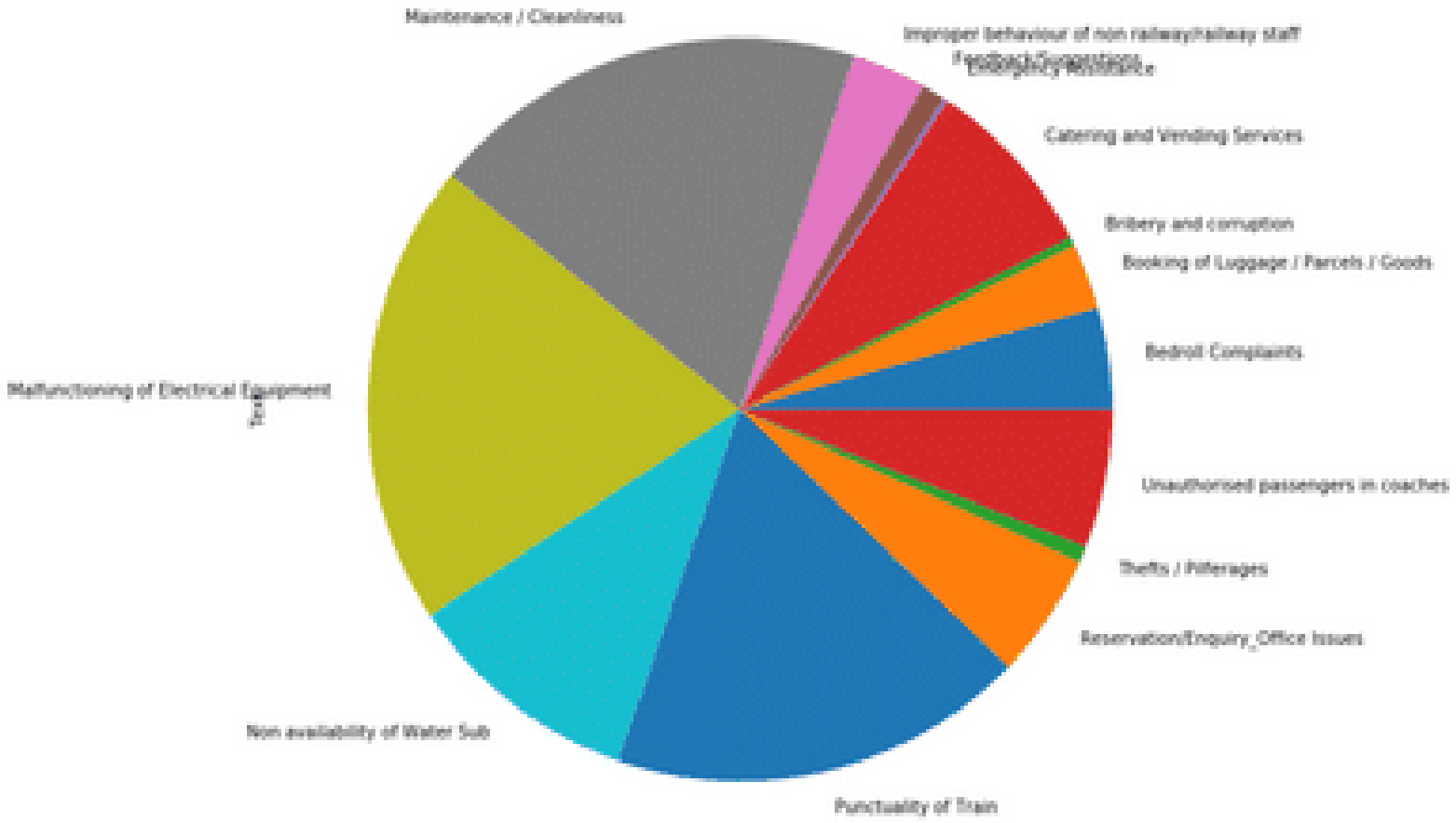
We have subdivided and categorized the text into using complaint_id : complaint,clean_text, sub_comlaint and the type of text.

| Text | Time | Complaint | Sub-Complaint | clean_text | clean_list |
|---|---|---|---|---|---|
| Sr. Citizen discount fare in tatkal sewa from ... | 2013-08-27 16:00:28 | Reservation/Enquiry_Office Issues | Any Other Issues (Enquiry) | sr citizen discount fare tatkal sewa jammu delhi | ['sr', 'citizen', 'discount', 'fare', 'tatkal'... |
| RATS ARE PRESENT IN COACH B1 OF TRAIN NUMBER 1... | 2013-08-14 21:12:49 | Maintenance / Cleanliness | Cockroaches | rat present coach train number ranakpur expres... | ['rat', 'present', 'coach', 'train', 'number',... |
| 12003 coach e1 toilet no 3 leaking vinay pnr 2... | 2013-08-27 18:31:57 | Maintenance / Cleanliness | A/C & Electrical fittings-Loose | coach toilet leak vinay pnr | ['coach', 'toilet', 'leak', 'vinay', 'pnr'] |
| Train no . 12321 pnr no 6122312493 complain _ ... | 2013-08-27 20:22:41 | Catering and Vending Services | Overcharging (Catering) | train pnr complain pentry car meal prize rs mi... | ['train', 'pnr', 'complain', 'pentry', 'car', ... |
| Train no . 12321 pnr no 6122312493 complain _ ... | 2013-08-27 20:22:41 | Catering and Vending Services | Overcharging (Catering) | train pnr complain pentry car meal prize rs mi... | ['train', 'pnr', 'complain', 'pentry', 'car', ... |

We have used matplotlib.pyplot that  is a collection of command style functions that make Matplotlib work like MATLAB. Each pyplot function makes some change to a figure: e.g., creates a figure, creates a plotting area in a figure, plots some lines in a plotting area, decorates the plot with labels, etc.

To know which complaint has got the maximum frequency , which has least frequency.




Maintenance / Cleanliness
Improper behaviour of non railway/railway staff
Catering and Vending Services
Bribery and corruption
Booking of Luggage / Parcels / Goods
Bedroll Complaints
Unauthorised passengers in coaches
Thefts / Pilferages
Reservation/Enquiry_Office Issues
Punctuality of Train
Non availability of Water Sub
Malfunctioning of Electrical Equipment

# DATA CLEANING

One of the biggest breakthroughs required for achieving any level of artificial intelligence is to have machines which can process text data. Thankfully, the amount of text data being generated in this universe has exploded exponentially in the last few years.

For having better accuracy we had to club together many different classes with similar department and thus decreased the number of classes. Finally we had the 14 classes of Complaint types.

It has become imperative for an organization to have a structure in place to mine actionable insights from the text being generated. From social media analytics to risk management and cybercrime protection, dealing with text data has never been more important.

We have some html in there that needs to be removed. We will take that out in the text processing step. As shown in the image the average word count by category:

| | Complaint | word_count |
|---|---|---|
| 0 | Allotment of seats - berths by train staff | 40.142310 |
| 1 | Bedroll Complaints | 26.492015 |
| 2 | Booking of Luggage / Parcels / Goods | 40.647111 |
| 3 | Bribery and corruption | 47.661212 |
| 4 | Catering and Vending Services | 32.932757 |
| 5 | Cleanliness at Station | 29.613687 |
| 6 | Complaints related to Sleeper Class | 32.273256 |
| 7 | Emergency Assistance | 30.643260 |
| 8 | Feedback/Suggestions | 37.361301 |
| 9 | Improper behaviour of commercial staff | 47.466766 |
| 10 | Improper behaviour of non commercial staff | 42.446475 |
| 11 | Improper behaviour of non railway staff | 34.272727 |
| 12 | Maintenance / Cleanliness of coaches | 23.356474 |
| 13 | Malfunctioning of Electrical Equipment | 21.071169 |
| 14 | Misc. Cause | 33.530526 |
| 15 | Multiple complaints | 37.122149 |
| 16 | Non availability of Water | 20.316679 |
| 17 | Passenger Booking | 40.006357 |
| 18 | Publicity | 31.347826 |
| 19 | Punctuality of Train | 30.249321 |
| 20 | Refund of tickets | 32.533146 |
| 21 | Reservation Issues | 42.026713 |
| 22 | Railway Police (GRUNSF) | 39.000000 |

## 1. Number of Words

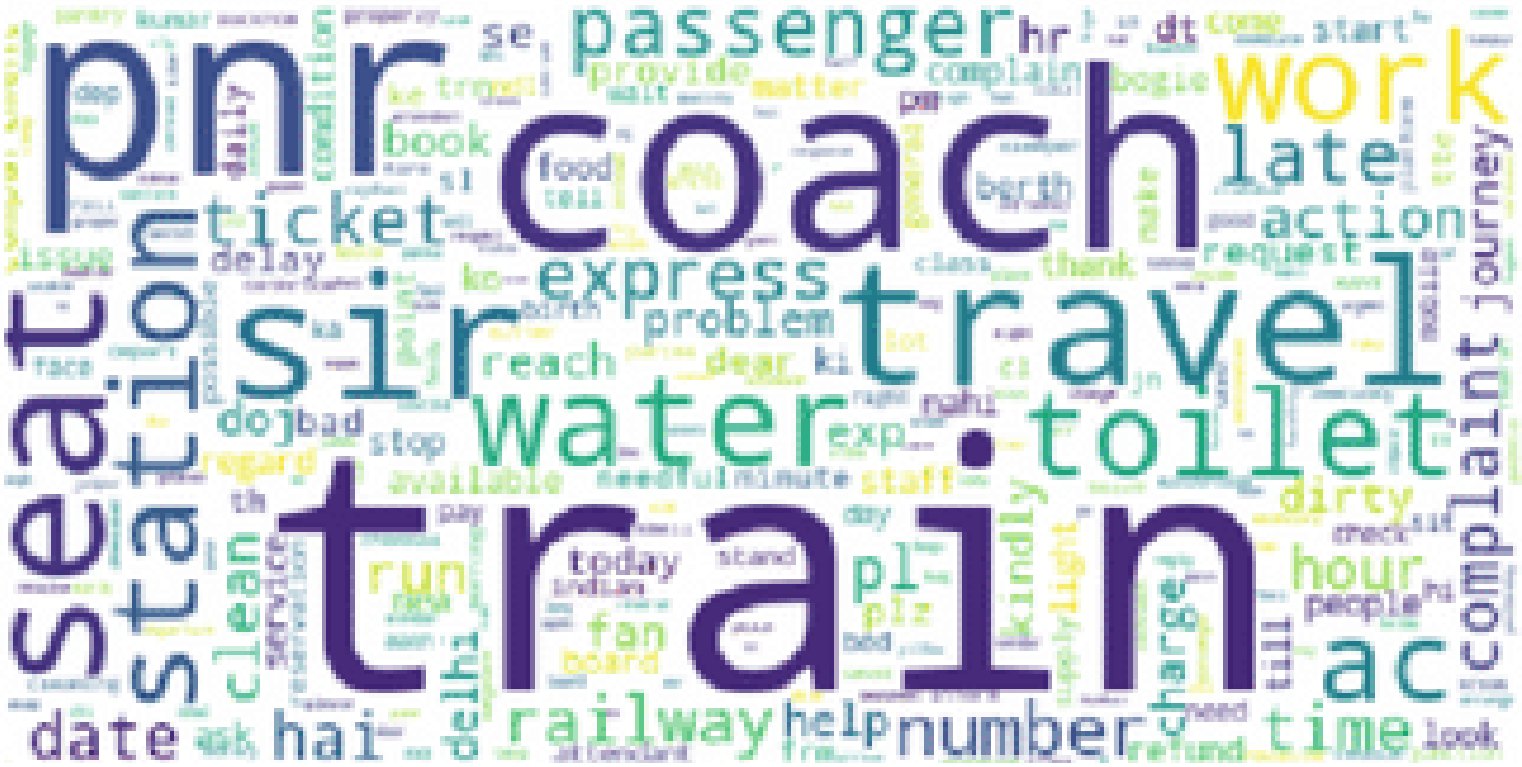
One of the most basic features we can extract is the number of words in each complaints. The basic intuition behind this is that generally,is to analyze how different classes have word count..

These word frequency can be also visualized in the form of WordCloud:

```
def word_freq(clean_text_list, top_n):
    """
    Word Frequency
    """
    flat = [item for sublist in clean_text_list for item in sublist]
    with_counts = Counter(flat)
    top = with_counts.most_common(top_n)
    word = [each[0] for each in top]
    num = [each[1] for each in top]
    return pd.DataFrame([word, num]).T
```

*##Top 12 most frequent words for all the articles*

| | | d |
|---|---|---|
| 0 | train | 215275 |
| 1 | coach | 95754 |
| 2 | pnr | 94722 |
| 3 | travel | 41403 |
| 4 | sir | 41310 |
| 5 | seat | 40268 |
| 6 | water | 39531 |
| 7 | work | 36183 |
| 8 | station | 36121 |
| 9 | toilet | 33629 |
| 10 | ac | 31200 |
| 11 | passenger | 29999 |
| 12 | express | 28730 |




## 2. Average Word Length

We will also extract another feature which will calculate the average word length of each tweet. This can also potentially help us in improving our model.

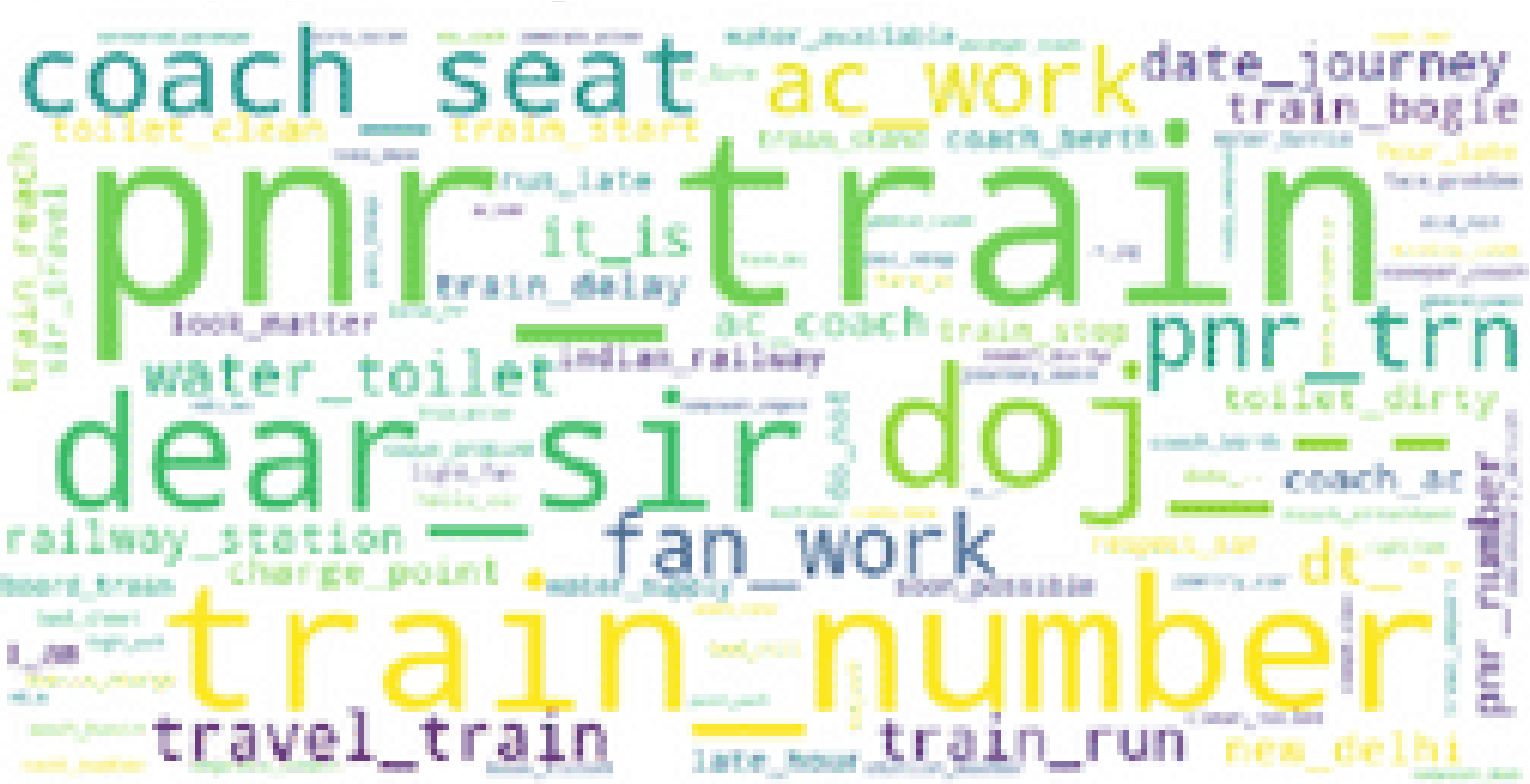
Here, we simply take the sum of the length of all the words and apply mean of the Complaints:

```
df['word_count'] = df['Text'].apply(word_count)
avg_wc = df.groupby('Complaint').mean().reset_index()
avg_wc[['Complaint','word_count']]
```

| | Complaint | word_count |
|---|---|---|
| 0 | Bedroll Complaints | 34.682914 |
| 1 | Booking of Luggage / Parcels / Goods | 44.880526 |
| 2 | Bribery and corruption | 55.589450 |
| 3 | Catering and Vending Services | 40.432834 |
| 4 | Emergency Assistance | 33.842105 |
| 5 | Feedback/Suggestions | 42.814880 |
| 6 | Improper behaviour of non railway/railway staff | 55.288318 |
| 7 | Maintenance / Cleanliness | 28.855554 |
| 8 | Malfunctioning of Electrical Equipment | 25.293531 |
| 9 | Non availability of Water Sub | 23.948973 |
| 10 | Punctuality of Train | 35.744415 |
| 11 | Reservation/Enquiry_Office Issues | 43.030197 |
| 12 | Thefts / Pilferages | 51.187622 |
| 13 | Unauthorised passengers in coaches | 40.228115 |

### 3.Bigrams

Some English words occur together more frequently. For example - Sky High, do or die, best performance, heavy rain etc. So, in a text document we may need to identify such pairs of words which will help in classification. First, we need to generate such word pairs from Such pairs are called bigrams.



In [31]:
```
bigrams1[:20]
```

Out[31]:

| | 0 | 1 |
|---|---|---|
| 0 | pnr_train | 22927 |
| 1 | train_number | 11830 |
| 2 | doj_-- | 9620 |
| 3 | dear_sir | 9105 |
| 4 | coach_seat | 8325 |
| 5 | ac_work | 7984 |
| 6 | pnr_trn | 7925 |
| 7 | fan_work | 6818 |
| 8 | travel_train | 6812 |
| 9 | water_toilet | 6270 |
| 10 | dt_-- | 5857 |
| 11 | train_run | 5847 |
| 12 | it_is | 5207 |
| 13 | date_journey | 4810 |
| 14 | pnr_number | 4314 |
| 15 | ac_coach | 3503 |
| 16 | train_bogie | 3485 |
| 17 | railway_station | 3461 |
| 18 | new_delhi | 3456 |
| 19 | charge_point | 2817 |

# FEATURES

We need to create features from the text. In order to do this, we need to turn the words into numbers because machines like numbers. The features will be created from the processed text, not the raw text.
There's a few ways to turn words into numbers, one is to create a matrix simply with the count of the words by article. We would use count vectorizer for that. Another way is to use tf-idf which stand for 'term frequency-inverse document frequency'.
The 'term frequency' is just the number of times a word appears in a document, divided by the total number of words in that document. The 'inverse document frequency' is the logarithm of the number of total documents divided by the number of documents the word appears in. Pretty straight forward.
This gives you a weight which is a good indicator of how important a word is to a document out of all the documents. That last part is important and gives context to the article.

```
from sklearn.feature_extraction.text import TfidfVectorizer
tfidf=TfidfVectorizer(ngram_range=(1,2),min_df=2,max_df=0.95)
X=tfidf.fit_transform(df['clean_text'].astype('str'))
```

```
y=df['complaint_id']
print(X.shape,y.shape)
```

```
(219820, 395639) (219820,)
```

- **Dimensionality Reduction**

Even though we removed stop words and are applying thresholds to the tf-idf vectorizer, it still leaves us with a lot of unique words (~15K), many of which we probably don't need and are redundant. So, let's also do Latent Semantic Analysis (LSA) which is a dimensionality reduction technique. LSA uses SVD or Singular Value Decomposition (and in particular Truncated SVD) to reduce the number of dimensions and select the best ones.

"LSA is known to combat the effects of synonymy and polysemy (both of which roughly mean there are multiple meanings per word), which cause term-document matrices to be overly sparse and exhibit poor similarity under measures such as cosine similarity"

```
#dimensionality reduction of 100 features
lsa=TruncatedSVD(n_components=100,n_iter=10,random_state=3)
X=lsa.fit_transform(X)
X.shape
```

```
(219820, 100)
```

- ## Model Selection and Evaluation

Everyone has their own process but the first thing we like to do is try a bunch of different kinds of classifiers and compare them with the default parameters. The huge caveat here is that an algorithm may not perform well right out of the box but will with the right hyperparameters.
We selected 6 different classifiers to test out along with sklearn's dummy classifier which is just random chance as a baseline. With 4 categories you would expect the accuracy to be around .25 and it is.
**Random Forest:** It is a supervised classification algorithm. Multiple number of decision trees taken together forms a random forest algorithm i.e the collection of many classification tree. It can be used for classification as well as regression.
**Decision Tree:** It belongs to supervised learning algorithm. Decision tree can be used to classification and regression both having a tree like structure. In a decision tree building algorithm first the best attribute of dataset is placed at the root, then training dataset is split into subsets. Splitting of data depends on the features of datasets.

**Naive-Bayes:** It is a technique for constructing classifiers which is based on Bayes theorem used even for highly sophisticated classification methods. It learns the probability of an object with certain features belonging to a particular group or class. In short, it is a probabilistic classifier.

**KNN:** This method is used for both classification and regression. It is among the simplest method of machine learning algorithms. It stores the cases and for new data it checks the majority of the k neighbours with which it resembles the most. KNN makes predictions using the training dataset directly.

**Gradient boosting and Ada Boost Algorithms :** Gradient boosting algorithm is a regression and classification algorithm. AdaBoost only selects those features which improves predictive power of the model.

In terms of the metrics to use to evaluate the different classifiers we're looking at:

**Accuracy -** simply the fraction of samples predicted correctly

**Precision -** the ratio of true positives to false positives or the ability of the classifier not to label a positive sample as negative

**Recall** - the ratio of true positives to false negatives or the ability of the classifier to find all the positive samples

**F1 Score** - The harmonic average of precision and recall

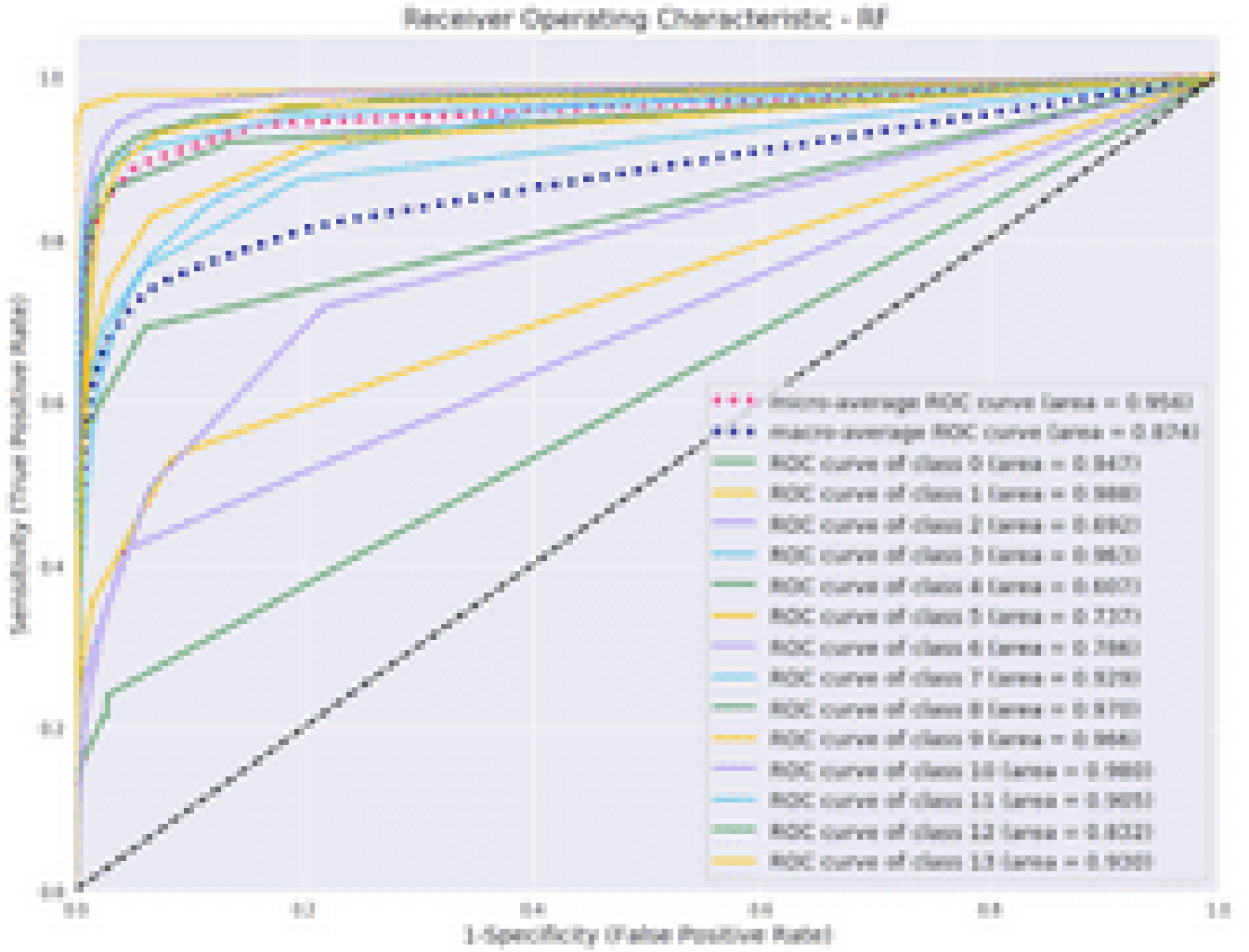
Precision-Recall Curve (graph): It shows the trade off between precision and recall. A high area under the curve (AUC) represents both high recall and high precision. High scores for both show that the classifier is returning accurate results (high precision), as well as returning a majority of all positive results (high recall)

| | model_name | accuracy_score | precision_score | recall_score | f1_score |
|---|---|---|---|---|---|
| 2 | Random Forest | 0.78 | 0.77 | 0.77 | 0.77 |
| 4 | AdaBoost | 0.75 | 0.74 | 0.74 | 0.74 |
| 1 | Stochastic Gradient Descent | 0.75 | 0.80 | 0.71 | 0.73 |
| 6 | K Nearest Neighbor | 0.71 | 0.71 | 0.69 | 0.70 |
| 5 | Gaussian Naive Bayes | 0.69 | 0.69 | 0.71 | 0.68 |
| 3 | Decision Tree | 0.69 | 0.67 | 0.67 | 0.67 |
| 0 | Dummy | 0.29 | 0.26 | 0.26 | 0.26 |

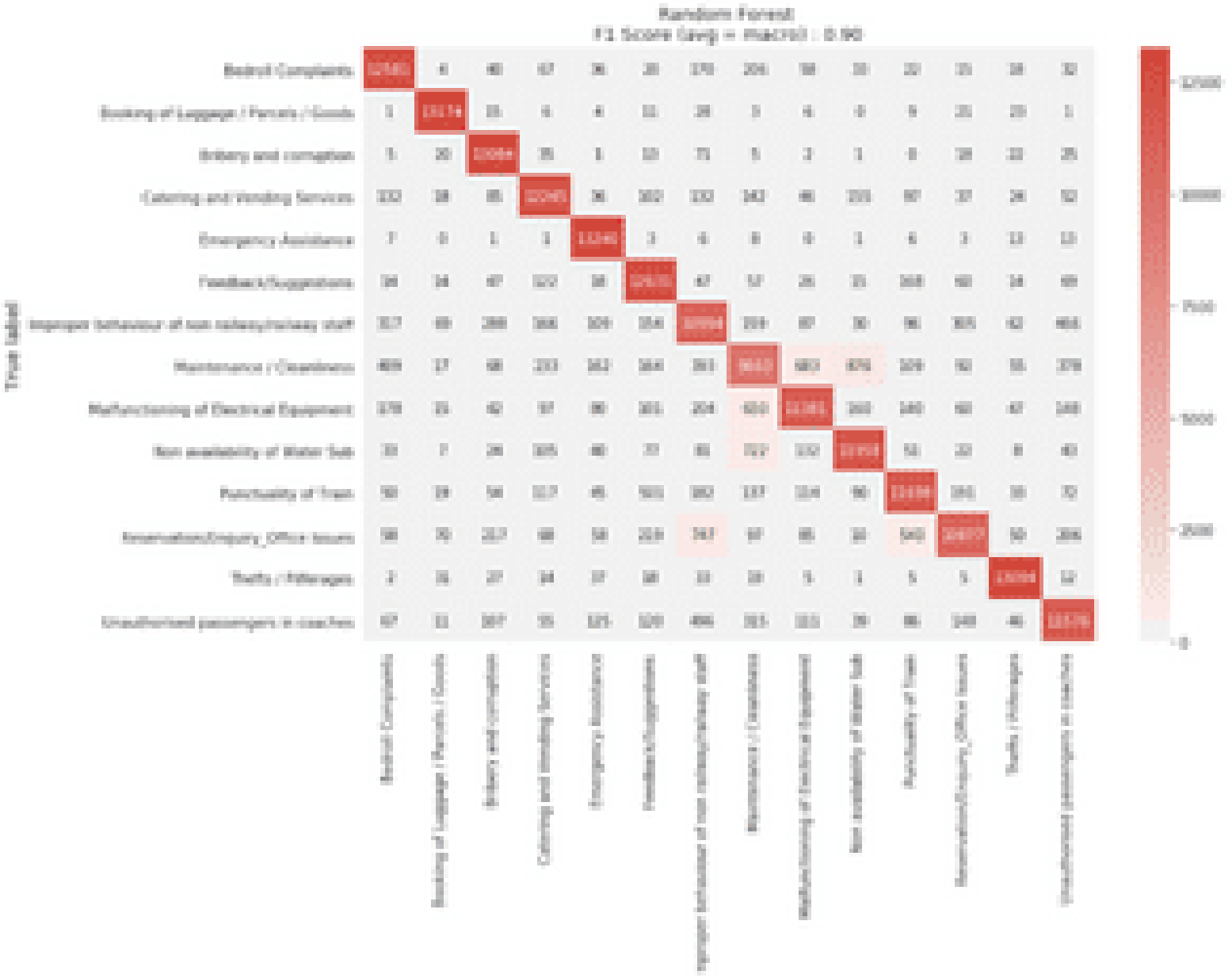
Random Forest has the highest F1 score (.77), followed by AdaBoost (.74) and the SGD (.73). For the sake of brevity we will going to continue with just two classifiers, Random Forest and SGD which implements a logistic regression. We've found SGD with logistic regression works well for text classification because it can deal with sparse data like we have with text.

We need to check or visualize the performance of the multi - class classification problem, we use AUC (Area Under The Curve) ROC (Receiver Operating Characteristics) curve.

Below is the ROC curve for SGD with the micro and macro averages along with each class:



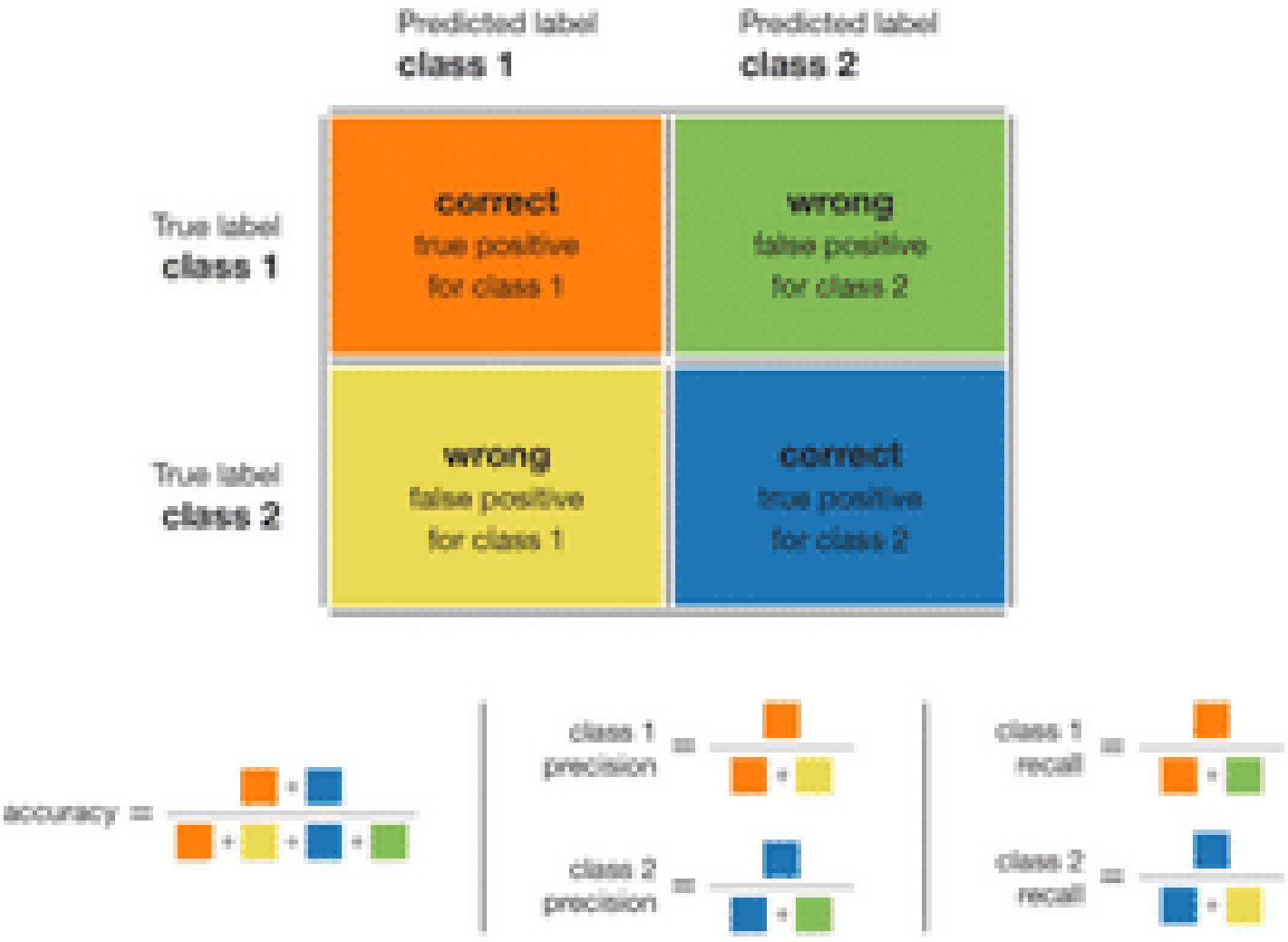

Following is the confusion matrix to see where the classifier is mixing up (confusing) categories:




We then made the classification report for the same on Random Classifier:

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| Bedroll Complaints | 0.91 | 0.95 | 0.93 | 13302 |
| Booking of Luggage / Parcels / Goods | 0.98 | 0.99 | 0.98 | 13302 |
| Bribery and corruption | 0.93 | 0.98 | 0.96 | 13302 |
| Catering and Vending Services | 0.92 | 0.92 | 0.92 | 13303 |
| Emergency Assistance | 0.95 | 1.00 | 0.97 | 13302 |
| Feedback/Suggestions | 0.89 | 0.95 | 0.92 | 13302 |
| Improper behaviour of non railway/railway staff | 0.81 | 0.83 | 0.82 | 13302 |
| Maintenance / Cleanliness | 0.79 | 0.73 | 0.76 | 13302 |
| Malfunctioning of Electrical Equipment | 0.89 | 0.86 | 0.87 | 13303 |
| Non availability of Water Sub | 0.89 | 0.90 | 0.90 | 13303 |
| Punctuality of Train | 0.90 | 0.88 | 0.89 | 13303 |
| Reservation/Enquiry Office Issues | 0.92 | 0.82 | 0.86 | 13302 |
| Thefts / Pilferages | 0.97 | 0.98 | 0.98 | 13303 |
| Unauthorised passengers in coaches | 0.88 | 0.87 | 0.88 | 13302 |
| accuracy | | | 0.90 | 186233 |
| macro avg | 0.90 | 0.90 | 0.90 | 186233 |
| weighted avg | 0.90 | 0.90 | 0.90 | 186233 |

We can see which complaint types have the highest precision, recall and f1-score where Booking of Luggage/Parcels/Goods(0.98) have the highest precision.
For better understanding of the terms we can see:


Predicted label
class 1
Predicted label
class 2
True label
class 1
correct
true positive
for class 1
wrong
false positive
for class 2
True label
class 2
wrong
false positive
for class 1
correct
true positive
for class 2
accuracy =
class 1
precision =
class 2
precision =
class 1
recall =
class 2
recall =

# LSTM MODELLING

- Vectorize consumer complaints text, by turning each text into either a sequence of integers or into a vector.
- Limit the data set to the top 5,0000 words.
- Set the max number of words in each complaint at 250.
- Now our aim is to get results from a Recurrent Neural Network (RNN) using the Long Short Term Memory (LSTM) architecture and how it can be implemented using Keras. We will use the same data source as we did Multi-Class Text Classification with Scikit-Learn.

```
# The maximum number of words to be used. (most frequent)
MAX_NB_WORDS = 50000
# Max number of words in each complaint.
MAX_SEQUENCE_LENGTH = 250
# This is fixed.
EMBEDDING_DIM = 100

tokenizer = Tokenizer(num_words=MAX_NB_WORDS, filters='!"#$%&()*+,-./:;<=>?@[\]^_`{|}~', lower=T
rue)
tokenizer.fit_on_texts(df['clean_text'].astype('str').values)
word_index = tokenizer.word_index
print('Found %s unique tokens.' % len(word_index))
```

- Truncate and pad the input sequences so that they are all in the same length for modeling.

```
X = tokenizer.texts_to_sequences(df['clean_text'].astype('str').values)
X = pad_sequences(X, maxlen=MAX_SEQUENCE_LENGTH)
print('Shape of data tensor:', X.shape)
```

```
Shape of data tensor: (219820, 250)
```

- Converting categorical labels to numbers.

```
Y = pd.get_dummies(df['Complaint']).values
print('Shape of label tensor:', Y.shape)
```

```
Shape of label tensor: (219820, 14)
```

- Train test split.

```
X_train, X_test, Y_train, Y_test = train_test_split(X,Y, test_size = 0.10, random_state =
42)
print(X_train.shape,Y_train.shape)
print(X_test.shape,Y_test.shape)
```

```
(197838, 250) (197838, 14)
(21982, 250) (21982, 14)
```

- The first layer is the embedded layer that uses 100 length vectors to represent each word.
- SpatialDropout1D performs variational dropout in NLP models.
- The next layer is the LSTM layer with 100 memory units.
- The output layer must create 13 output values, one for each class.
- Activation function is softmax for multi-class classification.
- Because it is a multi-class classification problem, categorical_crossentropy is used as the loss function.

```
model = Sequential()
model.add(Embedding(MAX_NB_WORDS, EMBEDDING_DIM, input_length=X.shape[1]))
model.add(SpatialDropout1D(0.2))
model.add(LSTM(100, dropout=0.2, recurrent_dropout=0.2))
model.add(Dense(14, activation='softmax'))
model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'
print(model.summary())
```

| Layer (type) | Output Shape | Param # |
|---|---|---|
| embedding_1 (Embedding) | (None, 250, 100) | 5000000 |
| spatial_dropout1d_1 (Spatial | (None, 250, 100) | 0 |
| lstm_1 (LSTM) | (None, 100) | 80400 |
| dense_1 (Dense) | (None, 14) | 1414 |

```
Total params: 5,081,814
Trainable params: 5,081,814
Non-trainable params: 0
```

```
epochs = 5
batch_size = 64

history = model.fit(X_train, Y_train, epochs=epochs, batch_size=batch_size,validation_spli
t=0.1,callbacks=[EarlyStopping(monitor='val_loss', patience=3, min_delta=0.0001)])
```

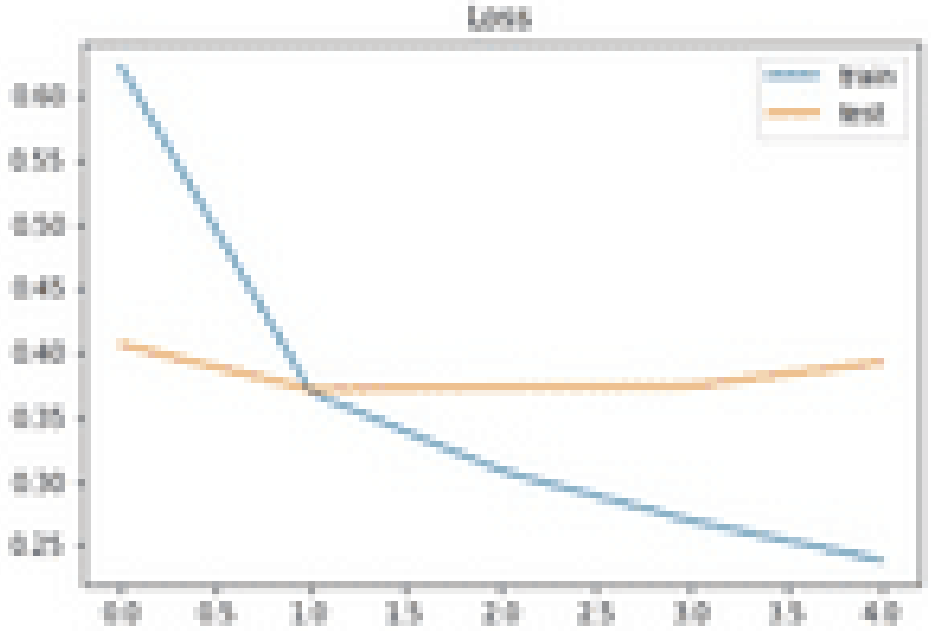
```
Train on 170854 samples, validate on 18984 samples
Epoch 1/5
170854/170854 [==============================] - 3253s 19ms/step - loss: 0.6228 - acc:
0.8138 - val_loss: 0.4005 - val_acc: 0.8820
Epoch 2/5
170854/170854 [==============================] - 3193s 19ms/step - loss: 0.3691 - acc:
0.8907 - val_loss: 0.3712 - val_acc: 0.8916
Epoch 3/5
170854/170854 [==============================] - 3428s 20ms/step - loss: 0.3081 - acc:
0.9071 - val_loss: 0.3734 - val_acc: 0.8951
Epoch 4/5
170854/170854 [==============================] - 3238s 19ms/step - loss: 0.2692 - acc:
0.9188 - val_loss: 0.3340 - val_acc: 0.8929
Epoch 5/5
170854/170854 [==============================] - 3180s 19ms/step - loss: 0.2386 - acc:
0.9272 - val_loss: 0.3928 - val_acc: 0.8908
```

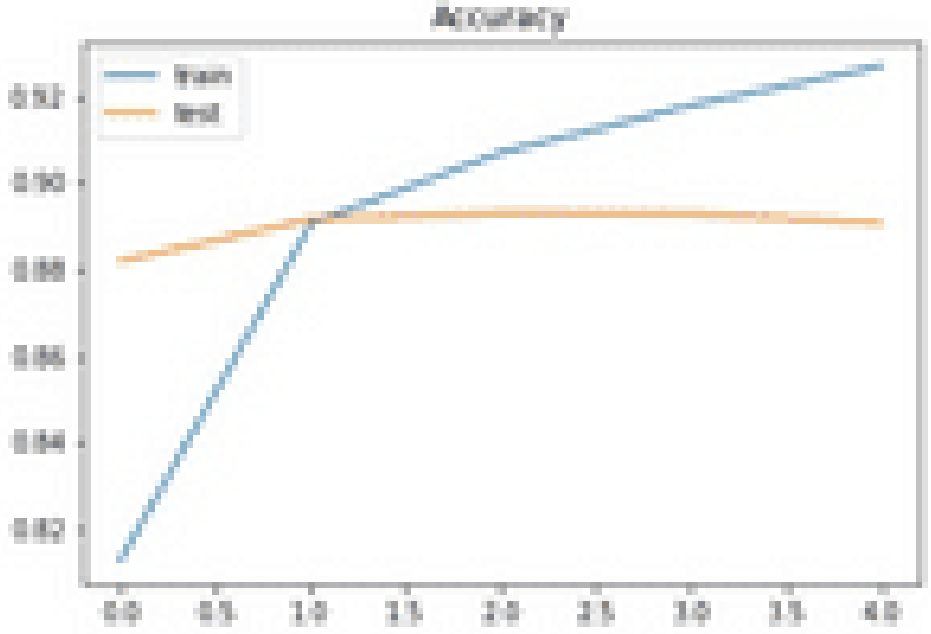
Accuracy(Testing):

```
accr = model.evaluate(X_test,Y_test)
print('Test set\n  Loss: {:0.3f}\n  Accuracy: {:0.3f}'.format(accr[0],accr[1]))
```

```
21982/21982 [==============================] - 34s 2ms/step
Test set
  Loss: 0.395
  Accuracy: 0.888
```

Accuracy And Loss Function Curve:



# RESULTS

Test with a New Complaint:

```
Hi! Welcome To Indian Railways:

Please provide us the following details:

Name:Anubhav Singh

Mobile Number:7353356814

Email-id:anubhav.singh2356@gmail.com

Complaint:I am travelling in Duronto Express to pune. The bedroll provided is too dirty and
no action was taken.Please help me asap.
Hello Anubhav Singh

Your Complaint is successfully registered under
Complaint Type: Bedroll Complaints  and will responded within 2 hours.

Inconvenience is deeply regretted
Thank You
```

```
Hi! Welcome to Indian Railways:

Please provide us the following details:

Name:[illegible]

Mobile Number:[illegible]

Email-id:[illegible]

Complaint:There no cleanliness and coaches are dirty no water available
Hello [illegible]

Your Complaint is successfully registered under
Complaint Type: Maintenance / Cleanliness  and will responded within 2 hours.

Inconvenience is deeply regretted
Thank You
```

# Future Scope

*The standard for creation, execution, and quality control*

Consistency is our top priority as we want to be able to serve everyone equally and to the best of our abilities.

**TWITTER CLASSIFICATION**

Before getting the system to work on a complaint, a tough task is to find out which tweet needs how much attention. “Since there is character limitation for tweets, our software processes the information based on keywords to analyse the complaint category. Once analysed, the actionable tweets are classified on the basis of priority—high, medium and llow .

Tweets which seek medical assistance, cleanliness or police help are kept in critical or high-priority category. Others such as those about broken windows, catering issues and missing parcels are categorised as medium and low priority. The handling of complaints has been outsourced to a trained team that works 24x7 in shifts.

## Project Report

**PREPARED BY** Anmol Saxena & Anubhav Singh

**PRESENTED ON** JJULY 15th , 2019