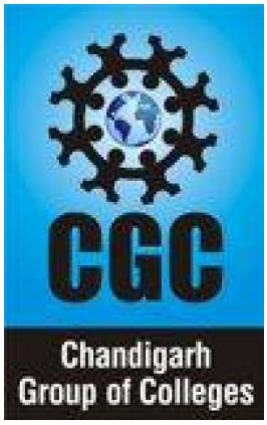
INDUSTRY ORIENTED PROJECT TRAINING REPORT

RailFeed

SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF

Degree of Bachelor of Technology in Computer Science & Engineering



Submitted By

CSE - E2

SUBMITTED TO: - Mr. Sumit Kumar

Department of Computer Science & Technology

CGC COLLEGE OF ENGINEERING, LANDRAN

CANDIDATE'S DECLARATION

I hereby declare that the Project Training Report entitled **"RailFeed"** is an authentic record of my own work as requirements of 6- months Industrial Oriented Project Training during the period from January to June for the award of degree of B. Tech. (Computer Science & Engineering) CGC College of Engineering under the guidance of (Ms. Anjali Sharma).

(Signature of Student)

Date: _____

Certified that the above statement made by the student is correct to the best of our knowledge and belief.

Signatures
Examined by:

Head of Department

(Signature and Seal)

ABSTRACT

The main idea of this project is to implement NLP (Natural Language Processing) and use it to build a fully functional Web Application. This is a web app which uses machine learning and NLP to carry out the task. Machine learning models are built and the app uses it to carry out the predictions. It has a frontend which is for the users and the backend which is only for predictions on the input from the user. The application takes input as the complaints against indian Railways and then it classifies into different available categories, so that then it can be further sent to the respective authorities for further action. This is a tool built to be used with application in various platforms. This project aims at automating the task to classify a complaint and then take action on it, rather than manual classification we are trying to achieve it through Machine Learning.

ACKNOWLEDGEMENT

I take this opportunity to express my sincere gratitude to the principal CGC College of Engineering, Landran for providing this opportunity to carry out the present work.

I am highly grateful to the Dr. Anuj Gupta HOD CSE, CGC College of Engineering, Landran (Mohali), for providing this opportunity to carry out the six month industrial training at Excellence Technology, Mohali. I would like to expresses my gratitude to other faculty members of Computer Science & Engineering department of CGC COLLEGE OF ENGINEERING Landran for providing academic inputs, guidance & Encouragement throughout the training period. I would like to express a deep sense of gratitude and thank Mr. Deepak Khashyap Branch Manager of Company, without whose permission, wise counsel and able guidance, it would have not been possible to pursue my training in this manner. The help rendered by Supervisor Ms. Anjali Sharma for Experimentation is greatly acknowledged. Finally, I express my indebtedness to all who have directly or indirectly contributed to the successful completion of my industrial training.

COMPANY PROFILE

Technology
EXCELLENCE
LEADER IN DEVELOPMENT

ABOUT COMPANY

EXCELLENCE TECHNOLOGY (ET) is India based leading strategic IT Company offering integrated IT solutions with the vision to provide Excellence in software solution. We at EXCELLENCE TECHNOLOGY bring innovative ideas and cutting edge technologies into business of customers. EXCELLENCE TECHNOLOGY is having rich experience in providing high technology end to end solutions in MOBILE APP AND WEB DEVELOPMENT.

EXCELLENCE TECHNOLOGY managing global clients across various business verticals and align IT strategies to achieve business goals. The various accreditations that we achieved for every service, we offer reflect our commitment towards the quality assurance. ET has won the NATIONAL AWARD for 2015-16 for the highly appreciable contribution in the field of computers fromHon'able Education Minister Of Punjab Dr.Daljit Singh Cheema

OUR TEAM:

The foundation upon which our team is created is based upon the premise that motivated people and long-standing relationships are the ultimate tools of success and creativity, energy, perseverance and loyalty are just as important as a platinum resume.

We have a team of highly qualified and experienced professionals with proven problem solving, consulting and analytical skills.

VISION:

To be a Leading information risk assessors by protecting the client organization's information system from threats and vulnerabilities.
To achieve the development objective to the best established practices and recognized standards worldwide and Regulatory as well as statutory obligation of the region in which organization is operating.

OUR SERVICES:

- ➢ RISK Management Services
- ➢ Quality Control
- ➢ Business Process Re-Engineering
- ➢ Network Risk Analysis
- ➢ Software Testing
- ➢ Mobile Application Testing
- ➢ Wireless Penetration Testing
- ➢ Network Penetration Testing
- ➢ Application Security Testing

OUR SERVICES IN SOFTWARE DEVELOPMENT:

We are proficient in all platforms of software Development practices — Agile, SCRUM, Lean, Waterfall, Prototype, Incremental, Iterative, and V-Model.

With the EXCELLENCE TECHNOLOGY experience the incredible services such as agile software development and the problems related to outsourcing. We comprise of the team of experienced and professionals members who with their skills efficiently get the job done and innovatively help you to transform your ideas into the successful business.

Why Choose Us?

- EXCELLENCE TECHNOLOGY is steadfast to undertake the projects cutting edge to technology competence and know-how abilities. The project execution is held with dedication and responsibility to perform our best with the essence of knowledge, creativity and skills to the utmost and efficiently.
- At EXCELLENCE TECHNOLOGY, we have competence to expand and adjust as per client specific requirements.
- **Skilled Workforce**: At EXCELLENCE TECHNOLOGY you deal with the highly professional and proficient employees.
- **Cost Efficiency**: We help you to reduce the unnecessary investment and ask for the reasonable amount of money.
- **Quality Of the Product**: Our software service sector has been maintaining the highest international standards of quality.
- **Infrastructure**: Well organized team and tools to handle the projects with responsible approach Hardware, Software, Networking, Voice, Conferencing, disaster recovery all infra all you need for international projects.
- **Ongoing Involvement**: EXCELLENCE TECHNOLOGY products are "built for change" as we are well responsive that the necessity to improve a Web solution generally arises even before the solution is out of the door. We delivers long-term product enhancement if desired.
- **Partnership**: EXCELLENCE TECHNOLOGY considers every client a partner. From the initial stages, you are closely involved into the procedure of technical classification, development, and testing.

TABLE OF CONTENTS

INTRODUCTION

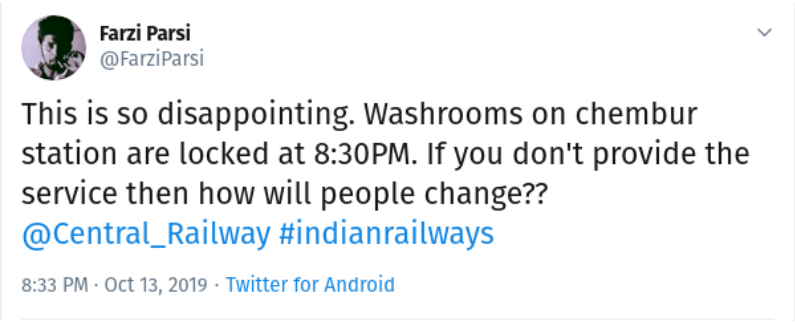
Indian Railways transport almost 2.5 Crores people daily. Passengers encounter a lot of problems daily, from cleanliness to theft. Our population is a big factor of the cause of the problems. People litter all over the seats, don't flush the toilet, steal the faucets, damage the property like seats, toiletries etc.

A lot of passengers face these problems but some just ignore while others take measures to register a complaint about it. There are a few methods, but the most frequently used two methods are either a call to the phone number given for respective problem or use Internet and tweet about the problem to the hashtag Indian Railways or the top authorities of the Ministry Of Railways.

It works quite well, as we have all seen it over the years. Required action is taken very quickly, but the task of categorizing the problem into respective categories is done manually. The question is, what if all the categorizing is automatically done by the software and then the complaint just sent to the respective railway authorities for the action.

It can work really well for the complaints made on twitter as all the complaints can be passed on to the software as a file/stream and then the application will do its job.
What the application does is that it takes the user input as a single complaint or a spreadsheet/csv file of the complaints and then predicts the categories it should fall into and then generates a new csv file. Application uses Machine Learning for the predictions, it uses NLP or Natural Language Processing for all the text processing, cleaning and tokenizing then the Naive Bayes Classifier for the classification. It comes under the Multiclass classification problem.

Below are examples of the complaints made on twitter. These are in forms of text, images, videos. But our Machine Learning model only works on text data.



Process:

- All the complaints made on twitter are scraped using scraping tools like tweepy, a tool for python. All the scraped data is then used to form a CSV file, further which will be used for the process.
- Data cleansing is the next process, as explained above our model only understands and operate on the text complaints. I cannot process images and videos or text in any language other than English. So, all the accepted complaints are kept and all the others are deleted/removed. Ads regarding the trending tags are also removed as they play no role in the process. All the tags/addresses present in the particular tweet is removed as they too play no role in the processing and prediction. Then only the complaints left are then forwarded for the further process.
- Next process is the manual categorization of the complaints, as they are needed by the machine learning model as this comes under supervised learning.
- Machine Learning model is built, it's a case of multi-class classification, we use the Linear Support Vector Classifier for the predictions. Model is then trained on the dataset.
- The Backend Server is built, API endpoints are made available for use.
- The Front-end is built using React (a JavaScript framework) and deployed to Netlify.

What Is Natural Language Processing?

Natural Language Processing, or NLP for short, is broadly defined as the automatic manipulation of natural language, like speech and text, by software.

The study of natural language processing has been around for more than 50 years and grew out of the field of linguistics with the rise of computers.

Natural language refers to the way we humans communicate with each other.

Namely, speech and text.

We are surrounded by text.

Think about how much text you see each day:

- Signs
- Menus
- Email
- SMS
- Web Pages
- *and so much more…*

The list is endless.

Now think about speech.

We may speak to each other, as a species, more than we write. It may even be easier to learn to speak than to write.

Voice and text are how we communicate with each other.

Given the importance of this type of data, we must have methods to understand and reason about natural language, just like we do for other types of data.

How does it work?

Let's understand using examples:-

Text: “It was a good movie. !!!: -)”

Stop Words and Tokenizing: - Bag of words which are of no use in the prediction of the text or analysis. So we tend to remove them from the text. Tokenizing means converting the text string in the form of a list.

Text: “It was a good movie. !!!: -)”

Text: - [‘It’, ‘was’, ‘a’, ‘good’, ‘movie’]

After removing stopwords:-

Text: - [‘good’, ‘movie’]

In this text, only the word movie and good were relevant so removing stop words will decrease the computation.

Tfidfvectorizer:- Convert a collection of raw documents to a matrix of TF-IDF features.

```
from sklearn.feature_extraction.text import TfidfVectorizer

corpus = ['This is the first document.',

   'This document is the second document.',

   'And this is the third one.',

   'Is this the first document?',]

vectorizer = TfidfVectorizer()

X = vectorizer.fit_transform(corpus)

print(vectorizer.get_feature_names())

print(X.shape)
```

Training the model

```
import pandas as pd
import matplotlib.pyplot as plt
df = pd.read_csv("../data/cleaned_data.csv")
df.head()
```

	complaint	category
0	Rly station gets bag scanner, still lacks adeq...	security
1	The sensitivity of towards the corridor betw...	journey
2	If you have Loyalty points then you can easil...	journey
3	Dear sir, I am travelling to ghazipur by 22434...	others
4	The sensitivity of towards the corridor betw...	journey

```
df.shape
(602, 2)
```

```
df['category_id'] = df['category'].factorize()[0]

cat_id_df = df[["category", "category_id"]].drop_duplicates().sort_value
cat_to_id = dict(cat_id_df.values)
cat_id_df
```

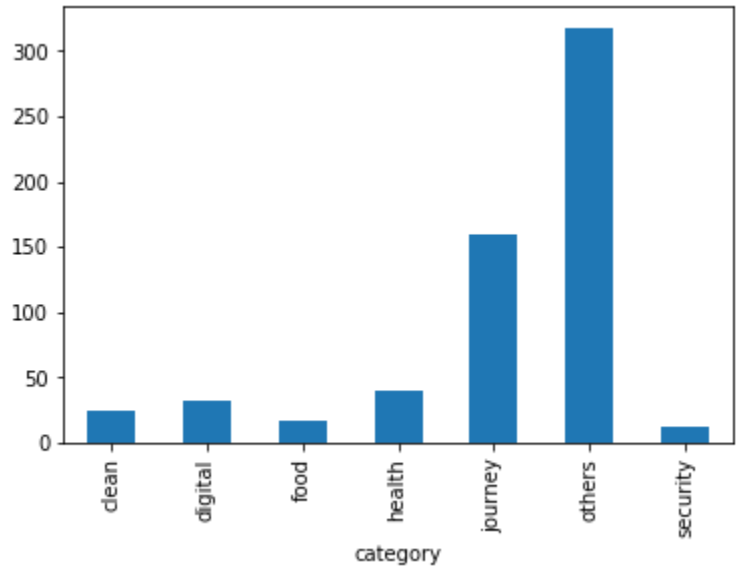
	category	category_id
0	security	0
1	journey	1
3	others	2
6	clean	3
9	digital	4
37	health	5
62	food	6

```
id_to_cat = dict(cat_id_df[['category_id','category']].values)
id_to_cat
{
'security',
'journey',
'others',
'clean',
'digital',
```

```
'health',
'Food'}

<Figure size 576x432 with 0 Axes>

df.groupby('category').complaint.count().plot.bar(ylim=0)
<matplotlib.axes._subplots.AxesSubplot at 0x7f2de9f8f278>
```

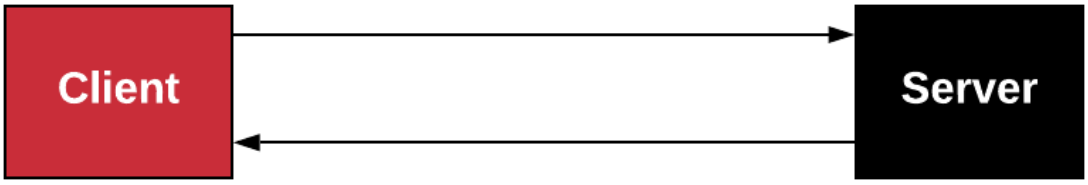



```
from sklearn.feature_extraction.text import TfidfVectorizer
tfidf = TfidfVectorizer(
sublinear_tf= True, #use a logarithmic form for frequency
min_df = 5, #min numbers of documents a word must be present in to be kept
norm= 'l2', #ensure all our feature vectors have a euclidian norm of 1
ngram_range= (1,2), #to indicate to consider both unigrams and bigrams.
stop_words ='english') #to remove all common pronouns to reduce the number
of noisy features
```

SYSTEM DESIGN

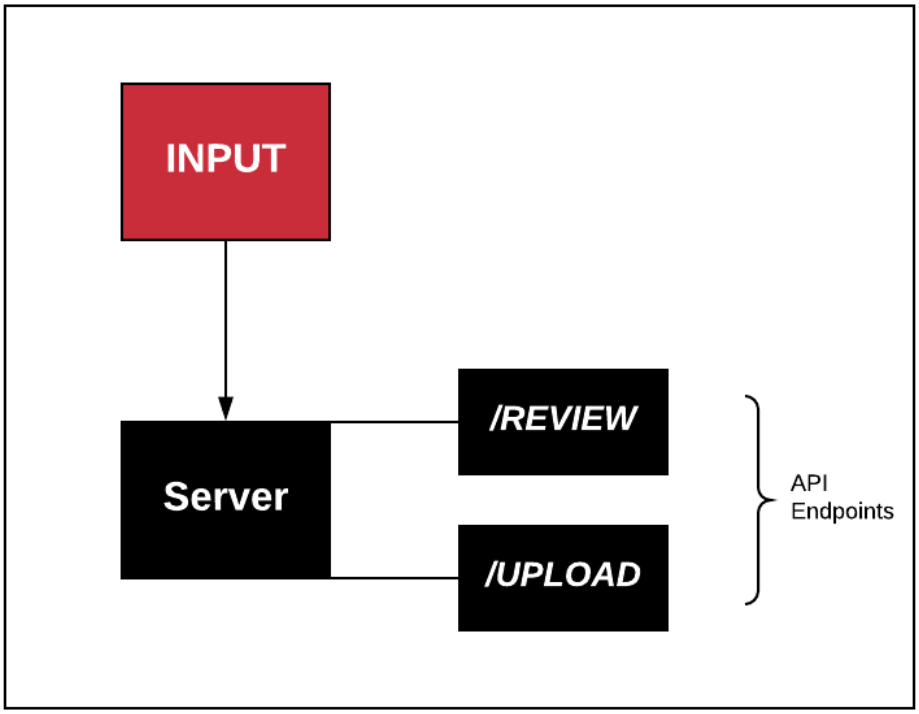
The Application follows a client-server architecture, where there is a client side application which communicates with the backend server.
There are several aspects of it, and explained in detail below:
The first design was meant to be as a single application with both server and the static files. It would consists all the features but are built using the single framework i.e. Flask(a python micro-framework), it would be the server and also serve the static files.
The problem with that is, it cannot be used as a tool according to user requirements as building this way would lose options for customizations. Hence, it was decided to make it in the form of API, which irrespective of the front-end platform we can make calls to the server and get the results.

The new formed design model looks like this:



The client communicates with the server through API calls in form of get/post requests. This is the 0-level design of the application.
Separate Client and Server, makes it very independent and more customizable at the client side.
No, it can be used in several platforms as, the client side application can consist of Web, Android/IOS, Windows/Linux/MacOS application.

Server:
The server has certain end-points which takes different inputs and do different tasks, and returns different results. It receives the http requests in form of GET/POST requests



Endpoints:

/review: This endpoint is made for testing of single review/complaint, it recieves the input as text and then returns the prediction or the classified category in JSON format

/upload: It requires a file to be uploaded, as the endpoint requires a csv file as input, then it gets stored in the server. Then prediction is made on all the complaints present in the file and then a new csv file is made which will contain the complaints as well as the predicted categories. Then it will return the response to the client. And file is made ready to download.

Environment:

Language Used:

Server: Python

Client : JavaScript

Frameworks Used: Flask(a Python micro framework), React

Development Tools: Git, VsCode, Heroku, Netlify, Jupyter Notebooks

VERIFICATION AND VALIDATION

Model validation is referred to as the process where a trained model is evaluated with a testing data set. The testing data set is a separate portion of the same data set from which the training set is derived. The main purpose of using the testing data set is to test the generalization ability of a trained model.

Model validation is carried out after model training. Model validation aims to find an optimal model with the best performance.

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(df['complaint'], df
from sklearn.feature_extraction.text import CountVectorizer
count_vect = CountVectorizer()
from sklearn.feature_extraction.text import TfidfTransformer
X_train_counts = count_vect.fit_transform(X_train)
tfidf_transformer = TfidfTransformer()
X_train_tfidf = tfidf_transformer.fit_transform(X_train_counts)
from sklearn.svm import LinearSVC
clf = LinearSVC().fit(X_train_tfidf, y_train)

y_pred = clf.predict(count_vect.transform(X_test))
from sklearn import metrics
print(metrics.classification_report(y_test,y_pred,labels=
df.caprint(metrics.classification_report(y_test,y_pred, labels=
df.category, target_names=df['category'].unique()))

features = tfidf.fit_transform(df.complaint).toarray()
labels = df.category_id
features.shape
(602, 321)
```

	precision	recall	f1-score	support
security	0.00	0.00	0.00	3
journey	0.51	1.00	0.67	35
others	0.51	1.00	0.67	35
clean	0.94	0.63	0.75	78
digital	0.51	1.00	0.67	35
health	0.51	1.00	0.67	35
food	0.50	0.50	0.50	8
micro avg	0.76	0.69	0.72	31520
macro avg	0.74	0.68	0.67	31520
weighted avg	0.85	0.69	0.73	31520

CRITICAL EVALUATION

In the end, the end product or the web application is made as a tool which can be used irrespective of the platforms used by client. It has good design which can be scaled to a big level without any problems. It has a modular design, most functional parts are built separately and hence easy to modify, or make changes to it.
The downsides of it is the accuracy of the predictions, due to the lack of good data, the models does not give good results. The complaints are taken from what tweets people made on Twitter, most of the complaints are not formed correctly, as some contains images, videos or other things. If the model would be fed good data that it can be trained on then it will give accurate results.

SUMMARY AND CONCLUSION

The intention was to create an application which will auto-categorize the complaints against Indian Railways into respective categories which will be further ease the process of taking action against it. The application developed takes input as csv file and then makes the predictions and returns the new file with categories mentioned as a new column in the file.
It does some of the tasks but still it requires development of some features which will make the application work in the way it is intended.
The first thing is the input data, now it has to be collected separately using twitter API wrapper libraries. But we want to make it convenient, like collection streams of data automatically and converting it into a csv file and simultaneously doing the further process and this should be happen in regular intervals, that’s what the intention is because then we can say it’s automating the problem.
The last thing is a way to make these auto categorized complaints reach the respective authorities so that action be taken.