Industrial safety NLP based Chatbot

FINAL report group 3

|  |
| --- |
| Team mentor |
| Gaurav Srivastav |

|  |
| --- |
| Team members |
| Hima Bindu |
| Chandini |
| Akshay Ashok Bhamare |
| Kathambari |
| Vishnu |
| Soham Parik |

[Team Details 2](#_heading=h.gjdgxs)

[**1.**](#_heading=h.2et92p0) **Summary of the problem statement, Data and findings** 5

[1.1 Problem](#_heading=h.tyjcwt) Statement 5

[1.2 Data & Findings 5](#_heading=h.3dy6vkm)

[**2.**](#_heading=h.1t3h5sf) **Overview of the Final Process** 7

[**3.**](#_heading=h.4d34og8) **EDA and Pre-Processing** 8

[**4.**](#_heading=h.2s8eyo1) **Visualization** 10

[4.1 GRP\_0 10](#_heading=h.17dp8vu)

[4.2 GRP\_8 10](#_heading=h.3rdcrjn)

[4.3 GRP\_24 11](#_heading=h.26in1rg)

[4.4 Plot showing Frequency Distribution of words 11](#_heading=h.lnxbz9)

[4.5 Inter-topic Distance Map 11](#_heading=h.35nkun2)

[4.6 Heat Map 14](#_heading=h.1ksv4uv)

[**5**](#_heading=h.44sinio) **Model building and evaluation** 15

[5.1 Model Approach 15](#_heading=h.2jxsxqh)

[5.2 Model creation 15](#_heading=h.z337ya)

[5.2.1 Logistic Regression Model 15](#_heading=h.3j2qqm3)

[5.2.2 Random Forest Model 16](#_heading=h.1y810tw)

[5.2.3 KNN Model 16](#_heading=h.4i7ojhp)

[5.2.4](#_heading=h.2xcytpi) LSTM Model 16

[5.2.5](#_heading=h.1ci93xb) Bidirectional LSTM with Time Distributed 17

[5.3 Model Summary 17](#_heading=h.3whwml4)

[**6**](#_heading=h.2bn6wsx) **Bench Marking - Comparison of experiments** 18

[6.1 Traditional Models 18](#_heading=h.qsh70q)

[6.1.1 Output of Logistic Regression 18](#_heading=h.3as4poj)

[6.1.2 Output of Logistic Regression with Balanced Data 18](#_heading=h.1pxezwc)

[6.1.3 Output of Random Forest 19](#_heading=h.49x2ik5)

[6.1.4 Output of KNN 19](#_heading=h.2p2csry)

[6.1.5 Summary of ML Models 19](#_heading=h.147n2zr)

[6.2 NLP Models 20](#_heading=h.3o7alnk)

[Approach to select the final model based on the outcome 20](#_heading=h.23ckvvd)

[6.2.1 LSTM 20](#_heading=h.ihv636)

[6.2.2 Bidirectional LSTM with Time distributed 22](#_heading=h.32hioqz)

[6.2.3 NLP Model Summary 22](#_heading=h.1hmsyys)

[**7**](#_heading=h.41mghml) **Implication - business value derived** 23

[**8**](#_heading=h.2grqrue) **Limitations and Scope of improvement** 23

[8.1 Limitations of data 23](#_heading=h.vx1227)

[8.2 Scope of Improvement 23](#_heading=h.3fwokq0)

[**9**](#_heading=h.1v1yuxt) **Closing Reflection** 24

[**10**](#_heading=h.4f1mdlm) **Final Note** 24

[**11**](#_heading=h.2u6wntf) **Code and References** 24

# 

# Summary of the problem statement, Data and findings

## **Problem Statement**

## **Project: Industrial safety. NLP based Chatbot**

The database comes from one of the biggest industries in Brazil and in the world. It is an urgent need for industries/companies around the globe to understand why employees still suffer some injuries/accidents in plants. Sometimes they also die in such environment

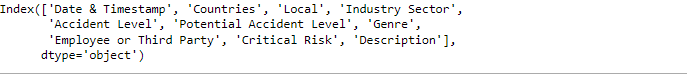
## **Abstract**

Applying traditional machine learning and neural network-based NLP to automatically classify records of accidents from 12 different plants in 03 different countries which every line in the data is an occurrence of an accident.

#### **Project Details:**

This The database is basically records of accidents from 12 different plants in 03 different countries which every line in the data is an occurrence of an accident.

#### **Import Dataset and data columns**

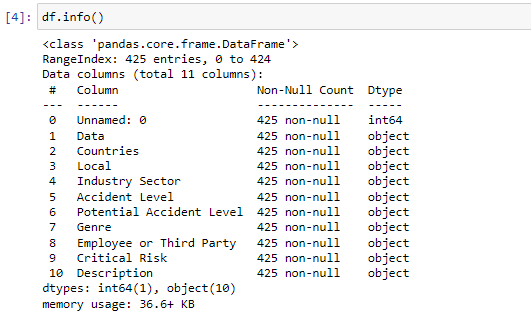


**Columns description:**

* Data: timestamp or time/date information
* Countries: which country the accident occurred (anonymized)
* Local: the city where the manufacturing plant is located (anonymized)
* Industry sector: which sector the plant belongs to
* Accident level: from I to VI, it registers how severe was the accident (I means not severe but VI means very severe)
* Potential Accident Level: Depending on the Accident Level, the database also registers how severe the accident could have been (due to other factors involved in the accident)
* Genre: if the person is male of female
* Employee or Third Party: if the injured person is an employee or a third party
* Critical Risk: some description of the risk involved in the accident
* Description: Detailed description of how the accident happened.
* The dataset is in csv format. Basic exploration of the data is as below

#### **Check for Shape, null values, Datatype &Missing Values**





* Non for the column have any null data.
* There are no missing values
* Total 10 columns and 425 rows of data.
* All the column datatype os object type.
* From the above output, we see that except first column all other columns data type is object

**Categorical columns**

'Countries', 'Local', 'Industry Sector', 'Accident Level', 'Potential Accident Level', 'Genre', 'Employee or Third Party', 'Critical Risk', 'Description'

Date column - 'Data'

**Given Data Summary**

There are about 425 rows and 11 columns in the dataset.

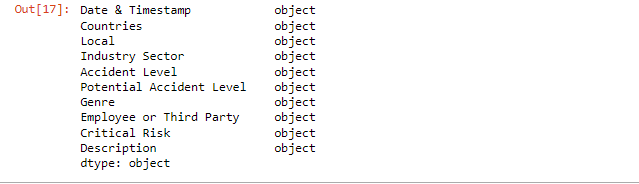
We noticed that except a 'date' column all other columns are categorical columns.

**Data Cleansing**

**Remove 'Unnamed: 0', Rename - 'Data' columns, Dropping Duplicates**

The field Unnamed: 0", is dropped and columns are renamed following

* ‘Data': Date & Timestamp



**Observation**

* We observed that there are records of accidents from 1st Jan 2016 to 9th July 2017 in every month. So there are no outliers in the 'Date' column.
* There are only three country types so there are no outliers in 'Countries' column.
* There are 12 Local cities where manufacturing plant is located and it's types are in sequence so there are no outliers in 'Local' column.
* There are only three Industry Sector types which are in sequence so there are no outliers in 'Industry Sector' column.
* There are only five Accident Level types which are in sequence so there are no outliers in 'Accident Level' column.
* There are only six Potential Accident Level types which are in sequence so there are no outliers in 'Potential Accident Level' column.
* There are only two Gender types in the provided data so there are no outliers in 'Gender' column.
* There are only three Employee types in the provided data so there are no outliers in 'Gender' column.
* There are quite a lot of Critical risk descriptions, and we don't see any outliers but with the help of SME we can decide whether this column has outliers or not.

**Data Cleansing Summary**

* Removed 'Unnamed: 0' column and renamed - 'Data' columns in the dataset.
* There are no outliers in the dataset.
* No missing values in dataset.
* We are left with 425 rows and 10 columns after data cleansing.

# 2.Overview of the Final Process

# The brief approach for the solution is given below

1. Solution requires model building based on the Classification model approach to design a ML/DL based chatbot utility which can help the professionals to highlight the safety risk as per the incident.
2. Data cleansing and Pre-Processing are important to have a good, cleaned input dataset for the model to predict the expected output. Hence the data cleansing and pre-processing steps are given in a detailed manner.
3. Visualization has been given to understand the dataset that feed into the model. This also helps to understand the structure of dataset
4. Two approaches of model creation is defined.

The first one is based on the conventional Machine learning algorithms Logical regression, Random Forest, KNN.

The second approach is based on the NLP algorithms LSTM and Bi-directional LSTM.

The evaluation approach is given as well.

1. The benchmarking of outcome has been captured. The performance of the model is tuned based on the different iterations with different parameters
2. The business derived value based on the outcome of the model is analysed.
3. Limitations of the model and scope of improvement has been covered.
4. The lessons learnt on each of the step of the project is noted down and summary is provided as Learnings

# 3.EDA and Pre-Processing

Below are the Pre-Processing steps applied while performing Elaborate Data Analysis on the input data.

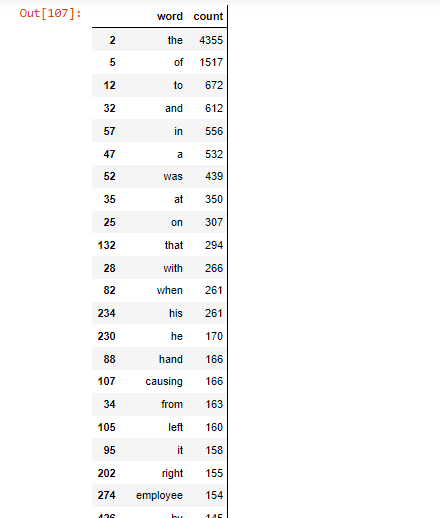
1. Verified the rows for Null values – no rows detected with null values
2. Checked if any column has missing value- There is no missing value found
3. Dropped the unused first column 'Unnamed'
4. Renamed the column with meaningful label
5. Visualisation and preprocessing of each variable
6. Plotted the most frequent words with word cloud
7. Observed few rows of Description has trailing spaces which affects analysis. Hence removed the trailing spaces.
8. Observed punctuations, tags, tabs, digits and special characters that expects to make noise. Removed the same.
9. Converting the text to lower case, avoid any capital cases and also Converting apostrophe to the standard lexicons.
10. #Most of the groups have less data but it expected to make more noise as the data is imbalanced in counts. Hence removing the rows that has lesser than threshold value of 31.
11. This constitutes 95% of data.
12. #The augmented description column is copied into 2 new columns. ML\_Description – for the traditional ML models and NL\_Description that will be used for the Neural network models.
13. Using fast text library, the non-English descriptions are identified and removed the rows.
14. 88% of data are English and others are detected as different language.
15. Punctuation marks part of English grammar are retained for the NL Model.
16. All special characters are removed for the traditional models
17. Using nltk - POS and Lemmatizer, we have further enhanced the ML\_Desription data to help the model to predict effectively.
18. #Still we observed non-ascii characters using word cloud that makes noise, hence removed them.
19. In Description, English STOP words, all special characters are removed for the traditional models
20. Using Glove Word Embeddings identified 400000-word vectors

# 4.Visualization

Top 3 Groups – in Word Cloud

## Plot showing Frequency Distribution of words

**Analyzing the frequent words**



# 5.Model building and evaluation

## 5.1 Model Approach

Solution requires model building based on the Classification model approach to predict the safety risk as per the incident description.

We can approach solution with both Conventional model and using NLP.

In Conventional Model, we are using Logistic regression, Random Forest and NN models to predict the safety risk.

The second approach is using LSTM and Bi-directional LSTM

## 5.1 Model creation

Following Model and accuracy scores are given as per the initial interim stage.

Further Model tuning and performance has been given in the next section

### **Logistic Regression Model**

##Using TfidfTransformer library in sklearn, bag of words is created to get the vocabulary (ngram 1,3)

Using Vectorizer transformation, features are mapped to training.

Logistic regression model is created and trained with 80-20 train test split.

**Logistic Regression Summary**

**Train Accuracy: 81%**

**Test Accuracy: 65%**

Another Logistic model is also built after balancing the input data using imblearn library

**##Logistic Regression with Balanced Training data Summary**

**Train Accuracy: 94%**

**Test Accuracy: 70%**

### **Random Forest Model**

Random Forest Model created and trained with 80 – 20 Train Test Split. The parameters for the Random forest model are

max\_depth=15

max\_features

max\_leaf\_nodes

n\_estimators

**Random Forest Summary**

**Train Accuracy: 75.29%**

**Test Accuracy: 73.47%**

### KNN Model

Model created and trained using KneighboursClassifier using neighbours = 5 and weights as distance

**KNN algorithm**

**Training Accuracy: 81%**

**Testing Accuracy: 66%**

### LSTM Model

Layer (type) Output Shape Param #

=================================================================

embedding\_1 (Embedding) (None, 95, 128) 256000

spatial\_dropout1d (SpatialD (None, 95, 128) 0

ropout1D)

bidirectional (Bidirectiona (None, 392) 509600

l)

dense\_5 (Dense) (None, 5) 1965

=================================================================

Total params: 767,565

Trainable params: 767,565

Non-trainable params: 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Training Accuracy 80.1%**

**Testing Accuracy 64.76%**

### LSTM Model with adding more layers

Embedding model is created and trained using glove 6B 300 Dimensions

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Layer (type) Output Shape Param #

=================================================================

embedding\_2 (Embedding) (None, 95, 128) 256000

spatial\_dropout1d\_1 (Spatia (None, 95, 128) 0

lDropout1D)

bidirectional\_1 (Bidirectio (None, 392) 509600

nal)

dense\_6 (Dense) (None, 64) 25152

dense\_7 (Dense) (None, 32) 2080

dense\_8 (Dense) (None, 5) 165

=================================================================

Total params: 792,997

Trainable params: 792,997

Non-trainable params: 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

None

**Train Accuracy is 80%**

**Test Accuracy is 64%**

### 5.1.6 RNN Model

**Train Accuracy is 49.4%**

**Test Accuracy is 65.5%**

## Model Summary

Summary of Model outputs

Below are the accuracy based on the Classes as 34

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sl.No** | **Model** | **Accuracy** | **Recall** | **Precision** | **F1-score** |
| 1 | Random Forest | 0.764706 | 0.764706 | 0.736652 | 0.727011 |
| 2 | Decision Tree | 0.647059 | 0.647059 | 0.681448 | 0.663082 |
| 3 | KNN Model | 0.811765 | 0.811765 | 0.666807 | 0.73218 |
| 4 | KNN Grid Search | 0.8 | 0.8 | 0.66506 | 0.726316 |
| 5 | DT Grid Search | 0.788235 | 0.788235 | 0.65528 | 0.715635 |
| 6 | LR Grid Search | 0.811765 | 0.811765 | 0.658962 | 0.727426 |
| 7 | RF Grid Search | 0.811765 | 0.811765 | 0.658962 | 0.727426 |
| 8 | KNN with SMOTE | 0.329412 | 0.329412 | 0.665794 | 0.422824 |
| 9 | NN Model | 0.505882 | 0.505882 | 0.614286 | 0.554839 |
| 10 | LSTM | 0.8 | 0.8 | 0.647619 | 0.715789 |
| 11 | LSTM Updated | 0.8 | 0.8 | 0.64 | 0.711111 |
| 12 | RNN | 0.658824 | 0.658824 | 0.670296 | 0.662457 |
| 13 | Bert Mode | 0.741176 | 0.741176 | 0.549343 | 0.631002 |

## **Bench Marking - Comparison of experiments**

## Traditional Models

Model has been tried to have Classes as 5 groups

### Output of Logistic Regression

Train Accuracy –; Test Accuracy -

**Confusion Matrix**

**Classification report**

### Output of Logistic Regression with Balanced Data

Train Accuracy – ; Test Accuracy :

**Classification report**

### Output of Random Forest

Train Accuracy : ; Test Accuracy :

**Classification report**

### Output of KNN

Training Accuracy : ; Testing Accuracy :

**Confusion Matrix**

**Classification report**

### Summary of ML Models

Although the outcome of Logistic regression accuracy is good, but the model is not expected to perform well for text processing of high volume of data.

Random Forest and KNN are not performed well.

## NLP Models

## Approach to select the final model based on the outcome

### LSTM

Initial LSTM model was created using

Embedding with SpatialDropout1D(0.05)

LSTM(3, dropout=0.05, recurrent\_dropout=0.05)

Dense(34, activation='softmax')

34 Assignment Group classes have been taken.

The model is little modified with

LSTM(128, dropout=0.2, recurrent\_dropout=0.2)(embedded\_sequences)

Dense(34, activation='softmax')

As a result, we got good training accuracy but no improvement in testing accuracy.

Various combination of parameters changed to check the accuracy

|  |  |  |
| --- | --- | --- |
| Parameters | Training Accuracy | Testing Accuracy |
| epochs = 10;batch\_size = 100  DIM EMBEDDINGS – 50, MAX SEQ LENGTH = 170 | 70 | 57 |
| epochs = 10;batch\_size = 100  DIM EMBEDDINGS – 50, MAX SEQ LENGTH = 200 | 74 | 61 |
| epochs = 15;batch\_size = 100  DIM EMBEDDINGS – 60, MAX SEQ LENGTH = 250 | 89.9 | 61.23 |
| epochs = 15;batch\_size = 100  DIM EMBEDDINGS – 80, MAX SEQ LENGTH = 250 | 90.78 | 63.89 |
| epochs = 15;batch\_size = 100  DIM EMBEDDINGS – 100, MAX SEQ LENGTH = 250 | 93.83 | 62.97 |

Training Accuracy is increased beyond 90, but no significant difference in Testing Accuracy

Tried in changing N\_Classes Parameter.

|  |  |  |
| --- | --- | --- |
| N\_Classes | Training Accuracy | Testing Accuracy |
| 74 | 92 | 68 |
| 10 | 95.47 | 76.45 |
| 8 | 96.49 | 78.02 |
| 5 | 94.10 | 90.76 |

Since Assignment Group are having imbalanced data, testing accuracy is not being improved with more classes.

Top 5 groups has been given as input for classes, it has given good results.

Training Accuracy - 94.10, Testing Accuracy – 90.76

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Model: "model\_5"

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Layer (type) Output Shape Param #

=================================================================

input\_6 (InputLayer) [(None, 250)] 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

embedding\_5 (Embedding) (None, 250, 100) 1500000

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

lstm\_5 (LSTM) (None, 128) 117248

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dense\_5 (Dense) (None, 5) 645

=================================================================

Total params: 1,617,893

Trainable params: 1,617,893

Non-trainable params: 0

### Bidirectional LSTM with Time distributed

Similar pattern is observed in Bidirectional LSTM with Time distributed as well.

|  |  |  |
| --- | --- | --- |
| N\_Classes | Training Accuracy | Testing Accuracy |
| 34 | 84.4 | 57.5 |
| 8 | 93.6 | 81 |
| 5 | 91.38 | 88.96 |

Model: "model\_7"

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Layer (type) Output Shape Param #

=================================================================

input\_8 (InputLayer) [(None, 100)] 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

embedding\_7 (Embedding) (None, 100, 200) 2129400

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

bidirectional\_1 (Bidirection (None, 100, 200) 240800

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

batch\_normalization\_1 (Batch (None, 100, 200) 800

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

time\_distributed\_1 (TimeDist (None, 100, 100) 20100

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

flatten\_1 (Flatten) (None, 10000) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dense\_10 (Dense) (None, 100) 1000100

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dense\_11 (Dense) (None, 34) 3434

=================================================================

Total params: 3,394,634

Trainable params: 3,394,234

Non-trainable params: 400

### NLP Model Summary

LSTM model has performed well when compared to Bidirectional LSTM.

It shows bidirectional wont perform well for text processing as there would be no additional significant difference when performing bidirectional

# Implication - business value derived

# xxxxx……

# Limitations and Scope of improvement

## Limitations of data

## Scope of Improvement

As mentioned in the limitation,

1. The pre-processing can be improved
2. Requesting business to give proper balanced data between the groups
3. Collecting more data for other Classes and reduce class difference within 15%
4. Equal and Standard unique words 200 or 500 on all the groups
5. Smote synthetic data.
6. Combine the similar classes with business decisions

The model is open for further improvement.

# Closing Reflection

1. XXXXX
2. XXXXXX

**Learnings**

* Had a good opportunity to learn preprocessing of text using various modules
* Learnt and used Language detection modules and translation
* Visualization using Word cloud to visualize the different combination of words in a group
* XXX
* Handling multi-classes distribution
* Handling imbalanced data and its implication

# Final Note

Thanks to Great Learning team for the help to learn AIML and to do this Capstone Project

Thanks to Aditi who has helped in many ways to complete this course

Many thanks to our Gaurav Srivastava, his experience in the field of AIML has guided us in learning throughout this course. The team appreciates his patience. His practical knowledge gives us a lot of insights to tackle the issues.

# Code and References