CAPSTONE PROJECT
AUTOMATIC TICKET ASSIGNMENT

FINAL REPORT – NLP 1 GROUP 2

Group Members:

1. Bhagyashree Borkar
2. Mohanasundari Ramalingam
3. M Nithin sai krishna
4. Qazi Tasawur
5. Sai Anusha

TABLE OF CONTENTS

CAPSTONE PROJECT – AUTOMATIC TICKET ASSIGNMENT

PROBLEM STATEMENT

The Problem is about the IT Company facing issue in assigning the Support tickets/incidents to the appropriate support person. The company trying to assign the tickets manually and then using the service manager to allocate the tickets to the support groups like L1, L2, and L3 to separate the category of the tickets. After assigning the tickets also there is chance of misallocating the tickets to wrong support person and that causes the company to involve in the process of re-assigning the tickets to right support person. This not only takes extra time and cost but also utilizing extra resource to get this done. This will create impact in the existing work carried on by the company.

To avoid the above issues, we should first concentrate on assigning/classifying the ticket groups that can be effectively done by the Artificial intelligence. Now the main objective of the problem statement is to classify the incidents.

DATA AND FINDINGS

The data provided for the project has below columns; let us understand the columns/features better.

```
df.head()
```

	Short description	Description	Caller	Assignment group
0	login issue	-verified user details.(employee# & manager na...	spxjnwir pjlcoqds	GRP_0
1	outlook	_x000D_\n_x000D_\nreceived from: hmjdrvpb.komu...	hmjdrvpb komuaywn	GRP_0
2	cant log in to vpn	_x000D_\n_x000D_\nreceived from: eylqgodm.ybqk...	eylqgodm ybqkwiam	GRP_0
3	unable to access hr_tool page	unable to access hr_tool page	xbkucsvz gcpydteq	GRP_0
4	skype error	skype error	owlgqjme qhcozdfx	GRP_0

Short description: It explains the type of issue faced by the employee in the company

Description: It explains the elaborate view of the issue face the employee in the company

Caller: It shows the right contact person for the issue mentioned by the employee

Assignment group: It displays the type of group the issue belongs

Shape of the data: It has 8500 rows and 4 columns.

```
df.shape
```

```
(8500, 4)
```

Data information:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8500 entries, 0 to 8499
Data columns (total 4 columns):
 #   Column             Non-Null Count  Dtype 
---  ------             --------------  ----- 
 0   Short description  8492 non-null   object
 1   Description        8499 non-null   object
 2   Caller             8500 non-null   object
 3   Assignment group   8500 non-null   object
dtypes: object(4)
memory usage: 265.8+ KB
```

All the columns have the data type as object.

Findings on the data

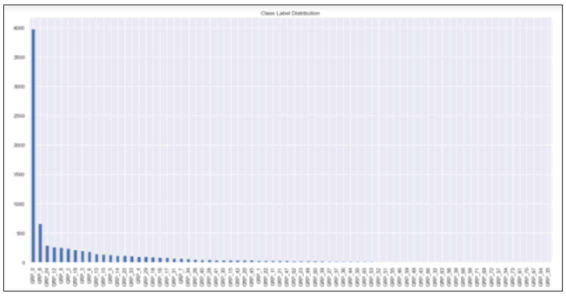
- The short description and description has the null values in it.
- After replacing the missing data and after removing duplicate rows, the rows are reduced to 8417.
- There are 74 unique assignment groups
- Among the groups, the GRP_0 has the more percentage of assignments say approximately 46 %
- Combining the short description & description fields as one column can reduce the usage of two columns
- The data has total corpus count as 307616
- 2818 records have same text/value for columns short description & description

APPROACH TO THE EDA:

The best way to understand the data is by Visualization. To analyze the data more we use EDA here and that includes many approaches. We are going to use the best approaches to our data that will give us insights about the data/features to develop the good model.

Visualization

Assignment group:



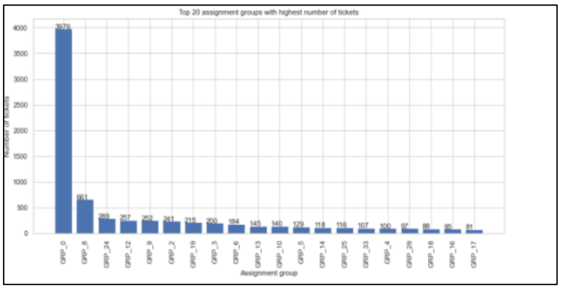
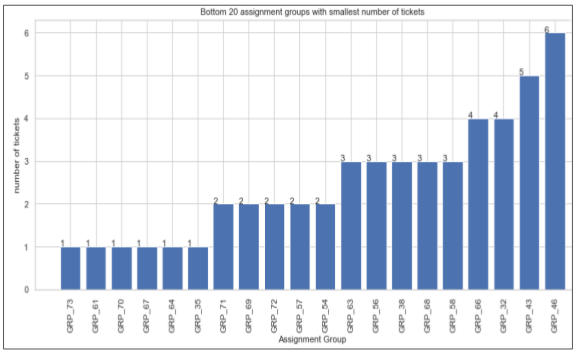


Figure 1.1

Figure 1.2

➢ From the Figures it is observed that more number of tickets assigned to the group_0



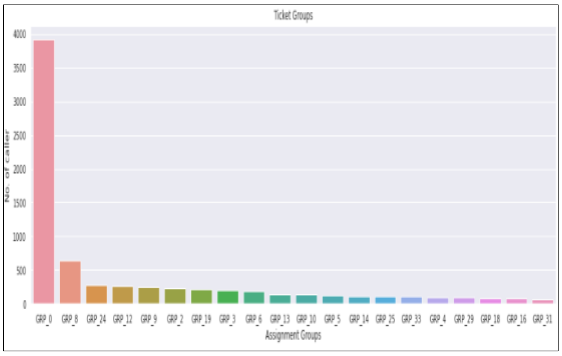


Figure 1.3

Figure 1.4

➢ From the Figures it observed that group_5 has the least number of assignments.

Top 10 assignment groups:

	index	Group	Count_of_Short_Desc
0	0	GRP_0	3934
1	72	GRP_8	645
2	17	GRP_24	285
3	4	GRP_12	257
4	73	GRP_9	252
5	12	GRP_2	241
6	11	GRP_19	215
7	23	GRP_3	200
8	56	GRP_6	183
9	5	GRP_13	145
10	2	GRP_10	140

CALLER:

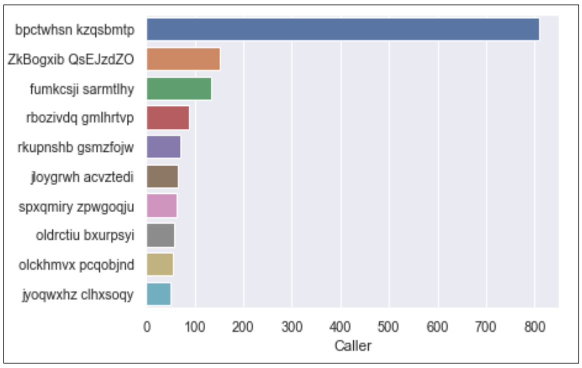


Figure 2.1

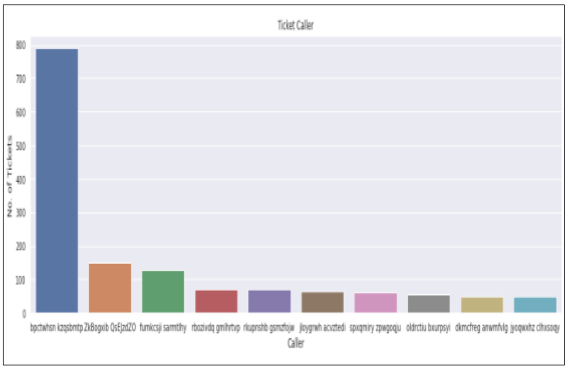


Figure 2.2

➢ The caller feature observation shows the more number of tickets has been assigned to the first caller support person.

SHORT DESCRIPTION & DESCRIPTION:

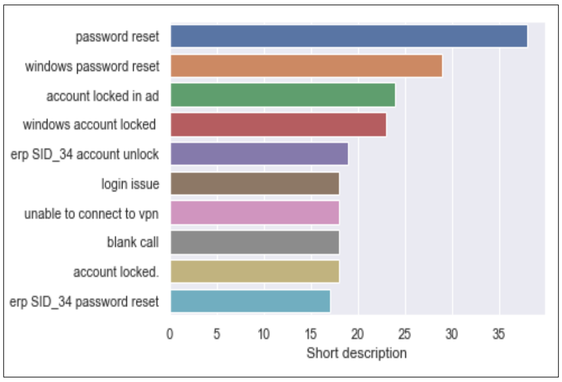


Figure 3.1

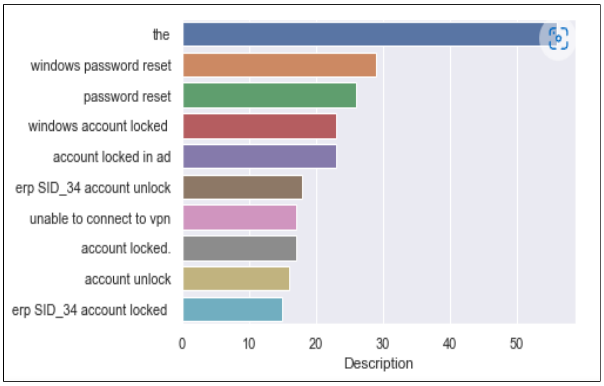


Figure 3.2

➢ It is observed that more number tickets created for the short description and description is different types of 'password reset' and least tickets are created for the 'account locked' cases.

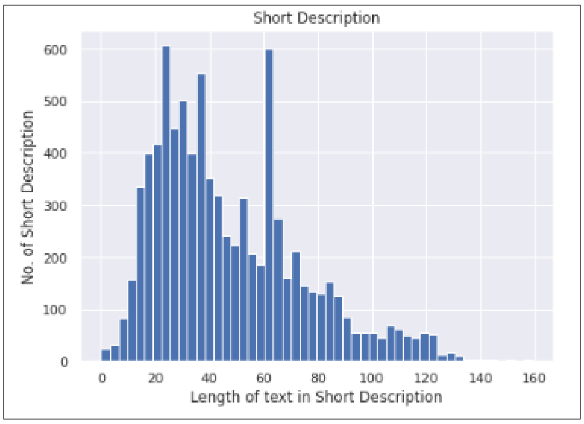


Figure 4.1

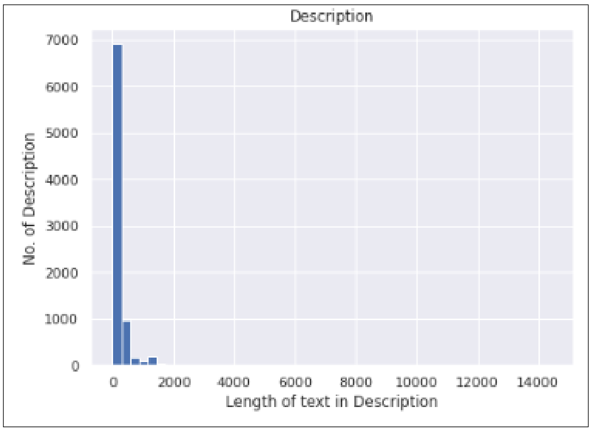
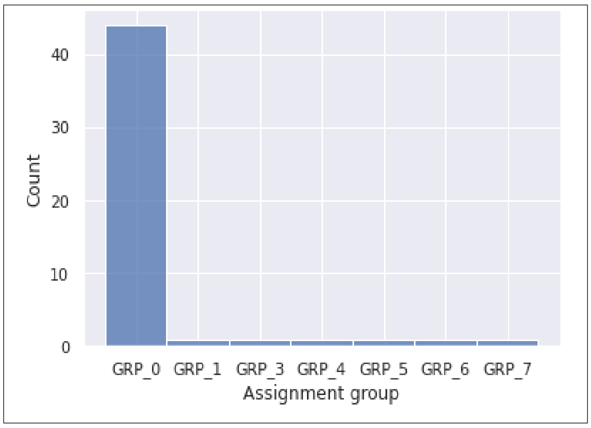


Figure 4.2

➢ It is observed that majority of 'Short Description' word count is between 10 and 200 and majority of 'Description' word count is between 1 and 1700.



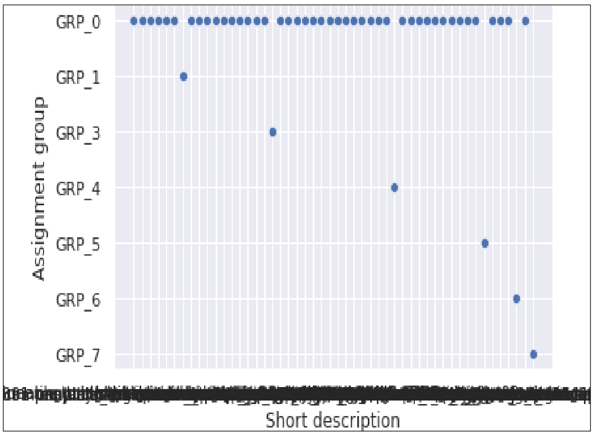
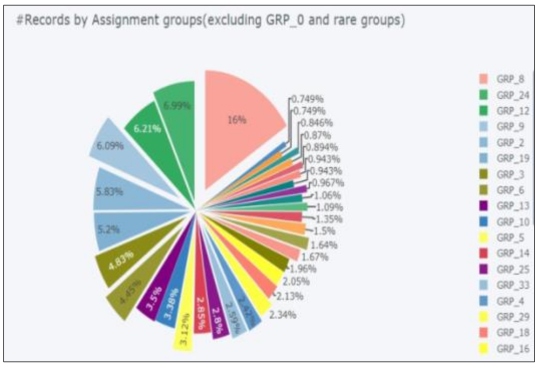


Figure 5.1

Figure 5.2

➢ The same insight of group_0 max assignments is observed through other EDA approaches.



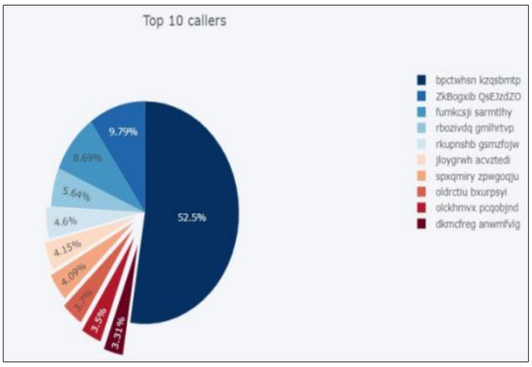


Figure 5.3

Figure 5.4

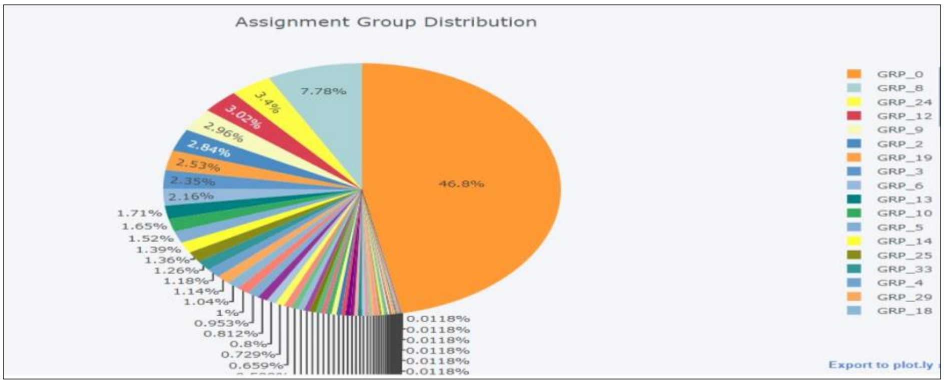


Figure 5.5

➢ Figures 5.3 to 5.5 in the EDA show the Assignment and caller feature percentages.

Overall Observations:

- ✓ The column 182 in the data has the maximum number of groups
- ✓ GRP_0 and GRP_8 has the maximum number of tickets, GRP_0 is first maximum and GRP_8 is second maximum
- ✓ From Figure 1.1 to 1.4, we can visualize the assignment group column and it shows the ranges and other Figures shows the different column ranges.
- ✓ 40 groups with minimal number of tickets say 30

APPROACH TO TEXT PREPROCESSING:

Before proceeding with the model building we have to make sure that the text we are using should be clean and perfect to get the best outcome. So the first process is to clean the data after getting vital information through the visualization. This step is preprocessing of text/data, which can be used to clean the important features.

Merging Short description and description text:

New column 'Text' created in existing dataframe. No missing values are present in dataframe.

Short description	Description	Caller	Assignment group	Text

Detect mojibakes and sanitize the dataset from mojibakes:

To display the original text instead of incorrect, unreadable characters, mojibakes is used. It is replacing incorrect, unreadable characters and returning readable/translatable text.

Language detection:

Detect the languages of text in the data. Segregate the text into English and non-English language. Then, translate the non-English text to English text.

Translation:

Once the language is detected, we can use the translator to translate non-English languages text to English text. Here, we have use the Google translator 'googletrans' to translate the text.

New column 'Translated_Text' created in existing dataframe

```
ticket_data['Translated_Text']
```

```
0       login issue verified user details employee man...
1       outlook x000d x000d received from hmjdrvpb kom...
2       cant log in to vpn x000d x000d received from e...
3       unable to access hr tool page unable to access...
4       unable to log in to engineering tool and skype...
                              ...
7527    emails not coming in from zz mail x000d x000d ...
7528    telephony software issue telephony software issue
7529    vip2 windows password reset for tifpdchb pedxr...
7530    machine n o est funcionando i am unable to acc...
7531    different programs cannot be opened on several...
Name: Translated_Text, Length: 7532, dtype: object
```

Figure 6.1

Cleaning the URL & special characters:

Selecting only alphabets/characters and numbers (A-Z/a-z, 0-9) from translated text.

Converting text to Lowercase:

If the text contains any uppercase letters, this process will change it to the lower case letters as highlighted in the figure. Review column is the processed text of description column.

Removing all extra white spaces:

Removing all extra white spaces from text

	Short description	Description	Caller	Assignment group	Text	Text_Lang	Translated_Text	Text_Lang_New
350	password reset for yscgjexz hxlbvjgf	password reset for yscgjexz hxlbvjgf	yscgjexz hxlbvjgf	GRP_0	password reset for yscgjexz hxlbvjgf password ...	no	password reset for yscgjexz hxlbvjgf password ...	en
1618	无法创建skype会议,outlook 日历上面没有 skype online meetin...	无法创建skype会议,outlook 日历上面没有 skype online meetin...	dqovxreg qswvlctg	GRP_31	无法创建skype会议,outlook 日历上面没有 skype online meetin...	no	unable to create a skype conference outlook sk...	en
1666	probleme mit laser02 \pfjwinbg ljtzbdqg	probleme mit laser02 \pfjwinbg ljtzbdqg	pfjwinbg ljtzbdqg	GRP_24	probleme mit laser02 \pfjwinbg ljtzbdqg proble...	no	problem with laser02 pfjwinbg ljtzbdqg problem...	en
2443	probleme mit skype und outlook \dardabthyr	probleme mit skype und outlook \dardabthyr	vzqomdgt jwoqbuml	GRP_0	probleme mit skype und outlook \dardabthyr pro...	no	probleme mit skype und outlook dardabthyr prob...	en
3198	mikrofon vom mobiltelefon gigaset sl 3 defekt!	bitte telefon tauschen.	qbsgwujo gvbzkjfq	GRP_33	mikrofon vom mobiltelefon gigaset sl 3 defekt!...	no	microphone vom mobile phone gigaset sl 3 defec...	en
3342	password reset for mii	password reset for mii_x000D_\n	mktwyzhx zhcenpvt	GRP_0	password reset for mii password reset for mii_...	no	password reset for mii password reset for mii ...	en
4437	barcode scanner defekt \paternoster \bur am orde	barcode scanner defekt \paternoster \bur am orde	xwirzvda okhyipgr	GRP_24	barcode scanner defekt \paternoster \bur am or...	no	barcode scanner defective paternoster bur am o...	en
4495	konto resetten	konto resetten	ughzilfm cfibdamq	GRP_0	konto resetten konto resetten	no	account reset account reset	en
4924	unable to login to skype	unable to login to skype	esfiuwcp pkaqnret	GRP_0	unable to login to skype unable to login to skype	no	unable to login to skype unable to login to skype	en
5222	tablet - dell 7350 - 电脑播放音频文件没有声音。	please provide details of the issue._x000D_\n 电...	riuhxcab jcsavihq	GRP_31	tablet - dell 7350 - 电脑播放音频文件没有声音。 please prov...	no	tablet dell 7350 please provide details of the...	en
6470	skype会议时不去	skype会议从邮箱里的链接进不去。	rekpvblc ufysatml	GRP_30	skype会议时不去 skype会议从邮箱里的链接进不去。	no	skype conference skype conference	en
6613	sipppr for help	dear it,\n\n以下是sipppr的入口链接。\n\n\n\n 我登录进去后 pdf文...	pfiyvdea uwbdsfmr	GRP_31	sipppr for help dear it,\n\n以下是sipppr的入口链接。\n\...	no	sippr for help dear it the following is sipppr...	en

Figure 6.2

Text
password reset for yscgjexz hxlbvjgf password ...
无法创建skype会议,outlook 日历上面没有 skype online meetin...
probleme mit laser02 \pfjwinbg ljtzbdqg proble...
probleme mit skype und outlook \dardabthyr pro...
mikrofon vom mobiltelefon gigaset sl 3 defekt!...
password reset for mii password reset for mii_...
barcode scanner defekt \paternoster \bur am or...
konto resetten konto resetten
unable to login to skype unable to login to skype
tablet - dell 7350 - 电脑播放音频文件没有声音。 please prov...
skype会议时不去 skype会议从邮箱里的链接进不去。

Translated_Text
password reset for yscgjexz hxlbvjgf password ...
unable to create a skype conference outlook sk...
problem with laser02 pfjwinbg ljtzbdqg problem...
probleme mit skype und outlook dardabthyr prob...
microphone vom mobile phone gigaset sl 3 defec...
password reset for mii password reset for mii ...
barcode scanner defective paternoster bur am o...
account reset account reset
unable to login to skype unable to login to skype
tablet dell 7350 please provide details of the...
skype conference skype conference

Figures 6.3 & 6.4

Word cloud:

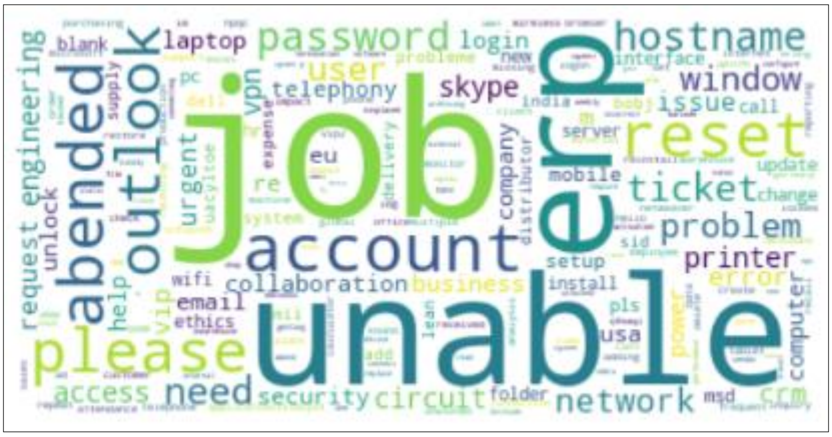


Figure 6.5

From the Figure it observed that the more occurring words in our text are unable, job, erp, password, outlook, need, ticket, abended, problem, plesae, account hostname, window, skype.

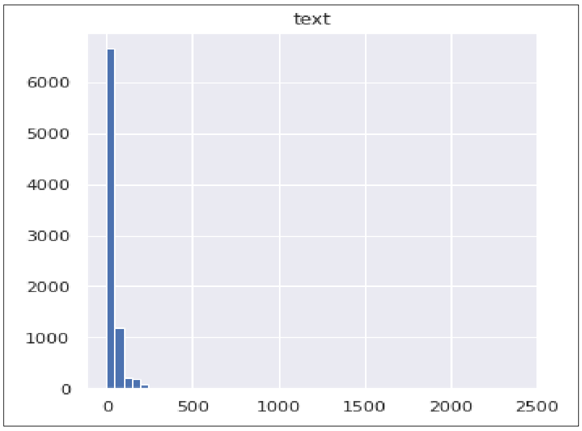


Figure 6.6

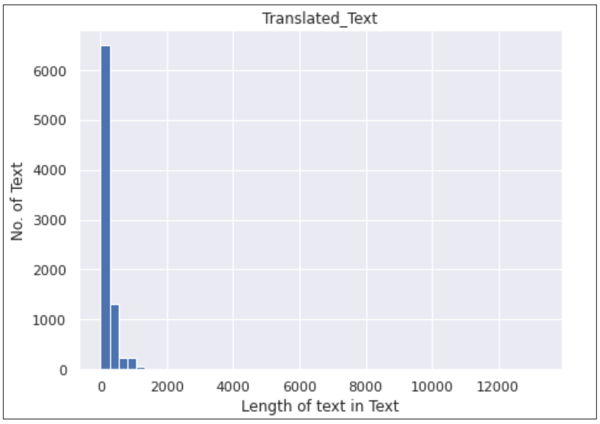


Figure 6.7

Figure 6.6 & 6.7 represents the maximum text words count is 2384, minimum words count is 2, maximum character in text is 13266 and minimum character in text is 3

MODEL BUILDING

Basic LSTM Model

After the visualization and text preprocessing, our data/text is cleaned to develop the NLP base model. To build the model we need the embedding vector length, length/number of words, frequent words from the text.

Training the model by giving proper parameters, assigning the loss function as cross entropy then optimizer and metrics are applied to study the model performance. Test the model with the parameters and compare the improvement achieved at the end of the model.

Model summary:

```
Model: "sequential_3"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding_3 (Embedding)     (None, 250, 32)           76288

 lstm_3 (LSTM)               (None, 128)               82432

 dropout_1 (Dropout)         (None, 128)               0

 flatten_3 (Flatten)         (None, 128)               0

 dense_9 (Dense)             (None, 256)               33024

 dense_10 (Dense)            (None, 128)               32896

 dense_11 (Dense)            (None, 74)                9546

=================================================================
Total params: 234,186
Trainable params: 234,186
Non-trainable params: 0
_________________________________________________________________
None
```

Figure 7.1

The observation gives the picture about the LSTM, embedding vectors. The number of parameters used by the layers is shown and LSTM shows the maximum number of parameters # 82432.

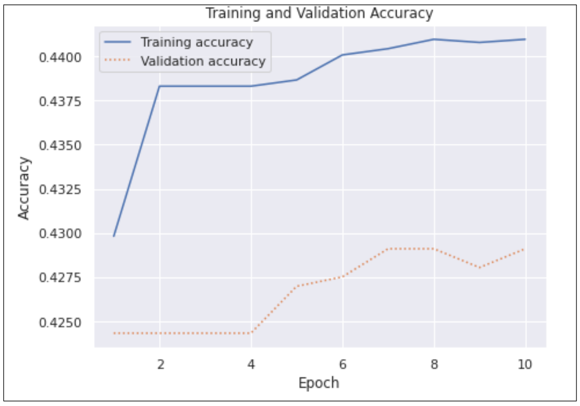


Figure 7.2

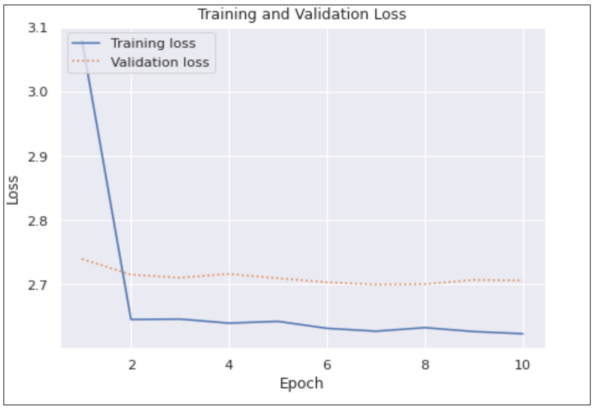


Figure 7.3

Figures 7.2 & 7.3 describe the accuracy and loss of the base model.

All the parameters taken from the layer are trainable and we have higher rate of chance for solving the issue and assign tickets to proper support person/caller.

The Accuracy of validation dataset for our model is calculated as 42% and below Figure 8.1 shows the classification report for test set.

Classification report for the test set:

Classification Report:

	precision	recall	f1-score	support
0	0.43	1.00	0.60	799
1	0.00	0.00	0.00	9
2	0.00	0.00	0.00	32
3	0.00	0.00	0.00	11
4	0.00	0.00	0.00	65
5	0.00	0.00	0.00	39
6	0.00	0.00	0.00	28
7	0.00	0.00	0.00	8
8	0.00	0.00	0.00	24
9	0.00	0.00	0.00	18
10	0.00	0.00	0.00	28
11	0.00	0.00	0.00	49
12	0.60	0.17	0.27	52
13	0.00	0.00	0.00	8
14	0.00	0.00	0.00	7
15	0.00	0.00	0.00	10
55	0.00	0.00	0.00	3
56	0.00	0.00	0.00	46
57	0.00	0.00	0.00	7
59	0.00	0.00	0.00	7
60	0.00	0.00	0.00	2
62	0.00	0.00	0.00	1
63	0.00	0.00	0.00	2
67	0.00	0.00	0.00	20
68	0.00	0.00	0.00	1
72	0.00	0.00	0.00	154
73	0.00	0.00	0.00	65
accuracy			0.43	1883
macro avg	0.02	0.02	0.01	1883
weighted avg	0.20	0.43	0.26	1883

Figure 7.4

Different Model Architectures:

Model with more LSTM Layers:

As the Previous model has the accuracy of 44% we are increasing the LSTM layer from 128 to 256 and run the built model to see the performance of the model, Below the Model summary of second model

Model Summary:

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_18 (Embedding)     (None, 200, 32)           76288

lstm_20 (LSTM)               (None, 200, 256)          295936

lstm_21 (LSTM)               (None, 200, 128)          197120

lstm_22 (LSTM)               (None, 74)                60088

flatten_3 (Flatten)          (None, 74)                0

dense_34 (Dense)             (None, 256)               19200

dense_35 (Dense)             (None, 128)               32896

dense_36 (Dense)             (None, 74)                9546

=================================================================
Total params: 691,074
Trainable params: 691,074
Non-trainable params: 0
_________________________________________________________________
None
```

Figure 8.1

Figure 8.1 shows the model summary. When we look into the accuracy of the model we get the same accuracy as 42.4%

So we observe adding/increasing the LSTM layers does not improve the efficiency for our model.

Bidirectional LSTM with ReLU Activation (GloVe Embedding (200d)):

LSTM model with ReLU Activation changes in spatial dropout, batch size and epochs and here applying the Glove embedding and below the Figure 9.3 shows the Model summary.

Model Summary:

```
Model: "model"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_1 (InputLayer)        [(None, 200)]             0

 embedding_2 (Embedding)     (None, 200, 200)          3359600

 bidirectional (Bidirectiona  (None, 256)              336896
 l)

 dropout_1 (Dropout)         (None, 256)               0

 dense_6 (Dense)             (None, 100)               25700

 dense_7 (Dense)             (None, 74)                7474

=================================================================
Total params: 3,729,670
Trainable params: 3,729,670
Non-trainable params: 0
_________________________________________________________________
```

Figure 8.2

This model gives the accuracy as 62%.

Bidirectional LSTM with TanH Activation (GloVe Embedding (200d)):

Model Summary:

```
Model: "model_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_2 (InputLayer)        [(None, 200)]             0

 embedding_3 (Embedding)     (None, 200, 200)          3359600

 bidirectional_1 (Bidirectio  (None, 256)              336896
 nal)

 dropout_2 (Dropout)         (None, 256)               0

 dense_8 (Dense)             (None, 100)               25700

 dense_9 (Dense)             (None, 74)                7474

=================================================================
Total params: 3,729,670
Trainable params: 3,729,670
Non-trainable params: 0
_________________________________________________________________
None
```

Figure 8.3

Now we have tried the Bidirectional LSTM model with the TanH activation, this model gives us the accuracy of 63% comparing our previous models and we could see more number of trainable parameters in the model summary (Figure 9.4). For classification report refer Figure 9.4.1 &9.4.2.

Classification report for Train set:

Classification Report:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	2476
1	1.00	0.89	0.94	18
2	0.99	1.00	1.00	106
3	1.00	1.00	1.00	19
4	1.00	0.97	0.98	183
5	1.00	0.98	0.99	102
6	0.98	0.99	0.98	88
7	1.00	1.00	1.00	29
8	1.00	1.00	1.00	61
9	0.96	1.00	0.98	46
10	1.00	1.00	1.00	60
11	1.00	1.00	1.00	148
12	0.99	0.98	0.99	185
13	0.96	0.96	0.96	27
14	1.00	1.00	1.00	21
15	1.00	1.00	1.00	21
61	1.00	1.00	1.00	1
62	1.00	1.00	1.00	9
63	1.00	1.00	1.00	2
64	1.00	1.00	1.00	1
65	1.00	1.00	1.00	3
66	1.00	1.00	1.00	2
67	1.00	1.00	1.00	48
69	1.00	1.00	1.00	2
70	0.00	0.00	0.00	2
71	1.00	1.00	1.00	1
72	0.96	0.92	0.94	488
73	0.98	0.85	0.91	185
accuracy			0.98	5649
macro avg	0.97	0.97	0.97	5649
weighted avg	0.98	0.98	0.98	5649

Figure 8.4

Classification report for Test set:

Classification Report:

	precision	recall	f1-score	support
0	0.80	0.84	0.82	799
1	0.43	0.33	0.38	9
2	0.64	0.56	0.60	32
3	0.67	0.18	0.29	11
4	0.66	0.48	0.55	65
5	0.49	0.49	0.49	39
6	0.46	0.39	0.42	28
7	0.00	0.00	0.00	8
8	0.47	0.33	0.39	24
9	1.00	1.00	1.00	18
10	0.50	0.43	0.46	28
11	0.25	0.22	0.24	49
12	0.45	0.58	0.51	52
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
55	0.00	0.00	0.00	3
56	0.53	0.59	0.56	46
57	1.00	0.14	0.25	7
59	0.00	0.00	0.00	7
60	0.00	0.00	0.00	2
62	0.00	0.00	0.00	1
63	1.00	0.50	0.67	2
66	0.00	0.00	0.00	0
67	0.59	0.85	0.69	20
68	0.00	0.00	0.00	1
72	0.67	0.66	0.67	154
73	0.61	0.57	0.59	65
accuracy			0.63	1883
macro avg	0.35	0.30	0.31	1883
weighted avg	0.62	0.63	0.62	1883

Figure 8.5

Bidirectional LSTM with ReLU Activation (GloVe Embedding (300d)):

```
Model: "model_5"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_6 (InputLayer)        [(None, 300)]             0

 embedding_7 (Embedding)     (None, 300, 300)          5039400

 bidirectional_5 (Bidirectio  (None, 256)              439296
 nal)

 dropout_6 (Dropout)         (None, 256)               0

 dense_16 (Dense)            (None, 100)               25700

 dense_17 (Dense)            (None, 74)                7474

=================================================================
Total params: 5,511,870
Trainable params: 5,511,870
Non-trainable params: 0
```

Figure 8.6

This model has the accuracy of 60% and the Figure 8.6 represents the model summary.

Bidirectional LSTM with TanH Activation (GloVe Embedding (300d)):

Below is the Model summary

```
Model: "model_6"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_7 (InputLayer)        [(None, 300)]             0

 embedding_8 (Embedding)     (None, 300, 300)          5039400

 bidirectional_6 (Bidirectio  (None, 256)              439296
 nal)

 dropout_7 (Dropout)         (None, 256)               0

 dense_18 (Dense)            (None, 100)               25700

 dense_19 (Dense)            (None, 74)                7474

=================================================================
Total params: 5,511,870
Trainable params: 5,511,870
Non-trainable params: 0
_________________________________________________________________
None
```

Figure 8.7

Classification report for test set:

Classification Report:

	precision	recall	f1-score	support
0	0.80	0.83	0.81	799
1	0.50	0.22	0.31	9
2	0.50	0.56	0.53	32
3	0.33	0.18	0.24	11
4	0.74	0.48	0.58	65
5	0.41	0.31	0.35	39
6	0.42	0.39	0.41	28
7	0.38	0.38	0.38	8
8	0.50	0.33	0.40	24
9	1.00	1.00	1.00	18
10	0.33	0.50	0.39	28
11	0.29	0.24	0.27	49
12	0.38	0.48	0.43	52
13	0.40	0.25	0.31	8
14	0.00	0.00	0.00	7

63	1.00	0.50	0.67	2
65	0.00	0.00	0.00	0
67	0.68	0.85	0.76	20
68	0.00	0.00	0.00	1
72	0.67	0.71	0.69	154
73	0.64	0.65	0.64	65
accuracy			0.63	1883
macro avg	0.35	0.32	0.31	1883
weighted avg	0.62	0.63	0.62	1883

Figure 8.8

This model has accuracy of 62% and Figure 8.8 represents the Classification report for test set.

LSTM model with ReLU activation (Word2Vec Embedding):

Using different models to fine tune the parameters and bring out the best accuracy. Here we are using the Word2Vec embedding for our model and below is the PCA visualization for the build model.

PCA visualization:

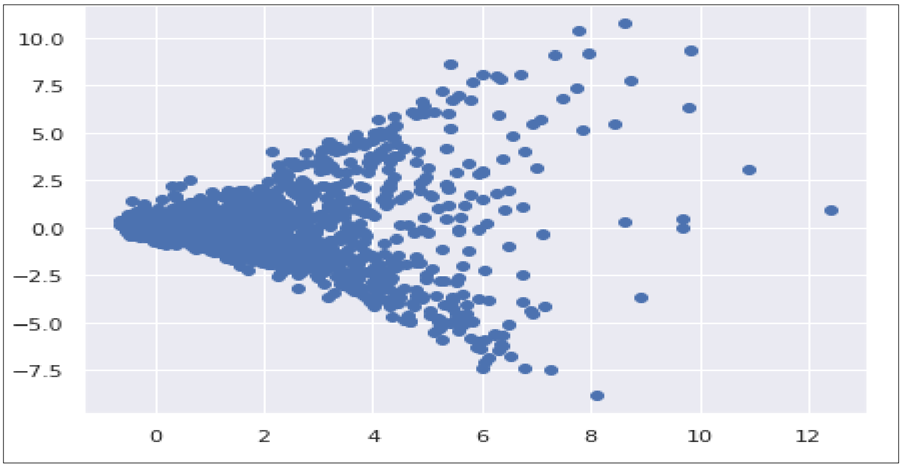


Figure 8.9

Model Summary:

```
Model: "sequential_2"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding_9 (Embedding)     (None, 200, 100)          2067900

 lstm_11 (LSTM)              (None, 128)               117248

 dropout_8 (Dropout)         (None, 128)               0

 flatten_2 (Flatten)         (None, 128)               0

 dense_20 (Dense)            (None, 256)               33024

 dense_21 (Dense)            (None, 128)               32896

 dense_22 (Dense)            (None, 74)                9546

=================================================================
Total params: 2,260,614
Trainable params: 2,260,614
Non-trainable params: 0
_________________________________________________________________
None
```

Figure 8.10

Accuracy of validation dataset is 51% and above is the model summary.

LSTM model with TanH activation (Word2Vec Embedding):

```
Model: "sequential_3"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_10 (Embedding)     (None, 200, 100)          2067900

lstm_12 (LSTM)               (None, 200, 256)          365568

lstm_13 (LSTM)               (None, 200, 128)          197120

lstm_14 (LSTM)               (None, 74)                60088

flatten_3 (Flatten)          (None, 74)                0

dense_23 (Dense)             (None, 256)               19200

dense_24 (Dense)             (None, 128)               32896

dense_25 (Dense)             (None, 74)                9546
=================================================================
Total params: 2,752,318
Trainable params: 2,752,318
Non-trainable params: 0
_________________________________________________________________
None
```

Figure 8.11

Accuracy for this model is 42.43% and above is the model summary.

Model on Top 5 Assignment groups data

Create new dataframe with top 5 assignment group data.

Confusion matrix for test set:

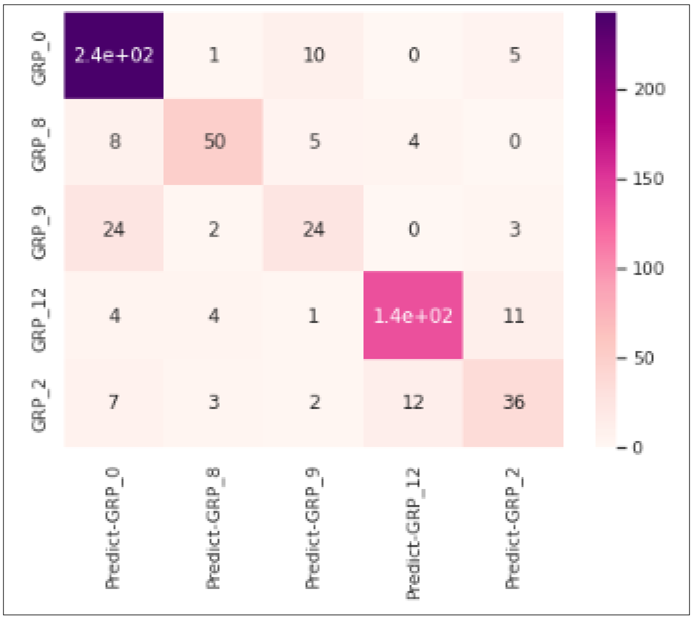


Figure 8.12

Building the model for top 5 assignment groups and observed the accuracy for the train & test set and we could see the highest accuracy of 82%. The Figure 9.11 shows the confusion matrix for the 5 assignment group for the test set.

Challenges:

- ✓ During the Text processing tried the different iteration for cleaning up the data.
- ✓ Using More Epochs we faced challenges in fitting the model thus by reducing the epochs with different iterations we could use different embeddings and produce the effective model
- ✓ In order to overcome the challenge in accuracy percentage we further fine tuned the model and arrived at best accuracy for our model.

Conclusion:

- ➢ Our basic model produces the accuracy of 42% (refer Model 1 in Jupyter notebook)
- ➢ Model with more LSTM Layers has the accuracy of 42.4% (refer Model 2 in Jupyter notebook)
- ➢ Bidirectional LSTM with ReLU Activation (GloVe Embedding(200d)) has the accuracy of 62% ((refer Model 3 in Jupyter notebook)
- ➢ Bidirectional LSTM with TanH Activation (GloVe Embedding(200d)) achieved the accuracy of 63% (refer Model 4 in Jupyter notebook)
- ➢ Bidirectional LSTM with ReLU Activation (GloVe Embedding(300d)) achieved the accuracy of 60% (refer Model 5 in Jupyter notebook)
- ➢ Bidirectional LSTM with TanH Activation (GloVe Embedding(300d)) achieved the accuracy of 62% (refer Model 6 in Jupyter notebook)
- ➢ LSTM model with ReLU activation (Word2Vec Embedding) has the accuracy of 51% (refer Model 7 in Jupyter notebook)
- ➢ LSTM model with TanH activation (Word2Vec Embedding) it produced 42% accuracy (refer Model 8 in Jupyter notebook)
- ➢ Top 5 Assignment group model achieved the accuracy of 82% (refer Model 9 in Jupyter notebook)

- ✓ **We are concluding the best model as Model 4 and Model 9 which gives the 63% and 82% of accuracy respectively.**

Future Enhancements:

- ✓ Transformation models can be designed further.
- ✓ BERT model can be included to analyze the accuracy of our model.
- ✓ Final Model can be saved/pickle and used for future reference
- ✓ Final Model can be Implemented as a web application so user can input and use the model for ticket assignment