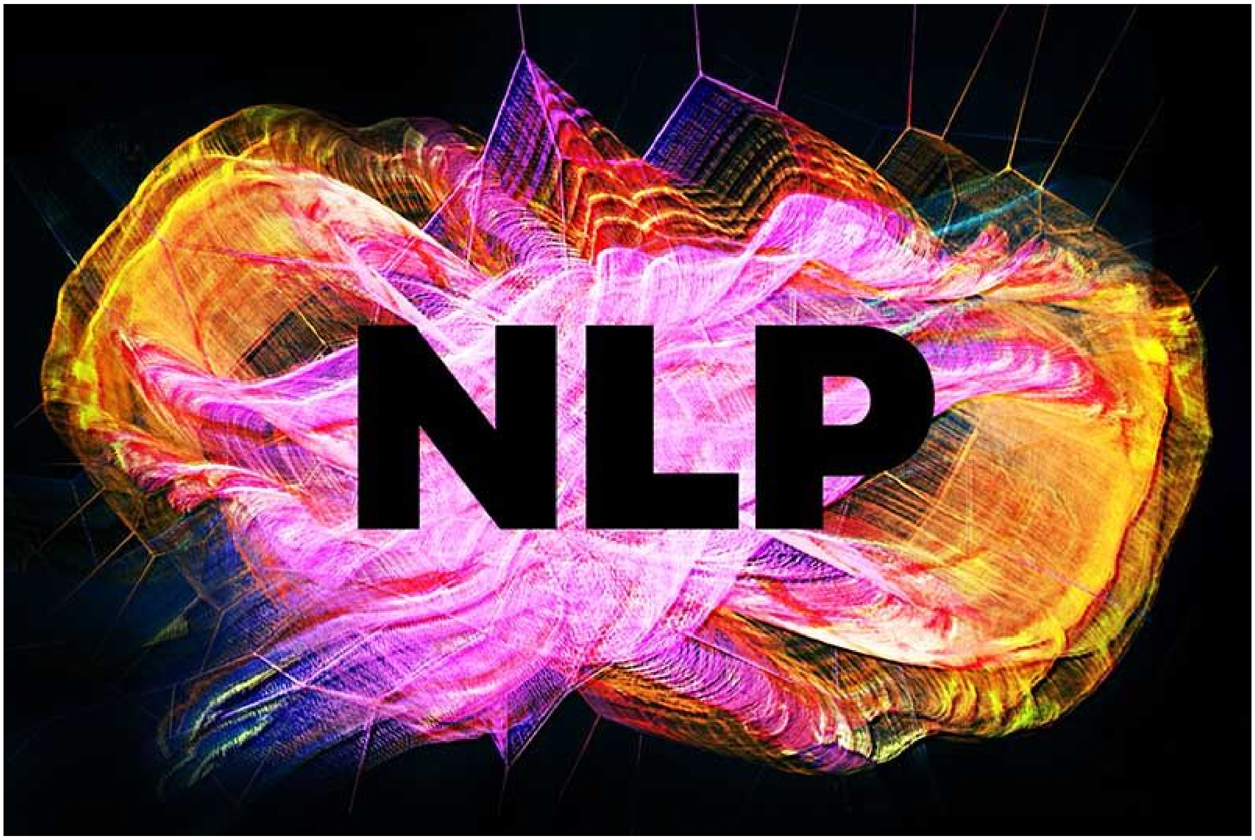


AIML Online Capstone - AUTOMATIC TICKET ASSIGNMENT - Interim Report

December 5th, 2020

Vineeth Govind, Goutham Jayaraman, Nishad Jadhav, Sridhar Krishnaswamy
AIML Online Batch Group 4 - Year 2020

Summary of problem statement, data and findings

Incident Management and Response is a key component of any IT Service Management Strategy. These are the typical steps involved in the Incident Management Process:

a. Receipt of the issue
b. Create a ticket
c. Review of the ticket by L1/L2 teams
d. Attempt to resolve the ticket using Standard Operating Procedures by L1/L2
e. If needed, transfer the ticket to the appropriate L3 team for further review and resolving.

Current 'Pain' Points

Currently the organization sees these issues in the Incident Ticket Management Process:

a. The process is largely 'manual'. L1/L2 teams need to spend time to review Standard Operating Procedures (SOPs) before assigning to functional teams. Minimum 25-30% incidents needs to be reviewed for SOPs before ticket assignment.

b. Minimum 1 FTE effort needed only for incident assignment to L3 teams

c. Human error - many times the incident gets assigned to the wrong L3 team. So additional effort needed to reassign to the correct team after re-review of the ticket, this not only increases the manual effort needed BUT *also leads to customer dis-satisfaction because the customer who opened the ticket is left frustrated because the ticket is in limbo being tossed between various teams before getting to the actual team who can help resolve the issued.*

Objective of this Project

The dataset provided has about 8500 records showing Ticket Long & Short Description, the caller, and the Group to which the Ticket has been assigned to.

Create various Machine Learning Models that can help classify incidents and assign them to the right Functional Group. Our objective is to create NLP models that can predict with at least 85% accuracy.

Milestone - 1

a. EDA - Explore and understand the dataset provided
    i. Visualizing different patterns
    ii. Visualizing different text features
    iii. Identify data discrepancies
b. Text preprocessing
    i. Dealing with missing values
    ii. Text Translation
c. Explore ways to augment the dataset provided without losing the relationship to the target label
d. Creating word vocabulary from the corpus of report text data
e. Creating tokens as required

Our Approach for Milestone-1

EDA:

- There are 8500 records in the dataset
- Each Dataset contains 4 columns
- The column 'Caller' seems to contain only junk. So we dropped the column
- We identified junk characters using ftfy library
- There were very few columns with Nulls

An example of data that looked like junk at an initial glance. But on applying the ‘fixes text for you - ftfy’ library, we discovered that the data is actually Ticket Description in Chinese Simplifed

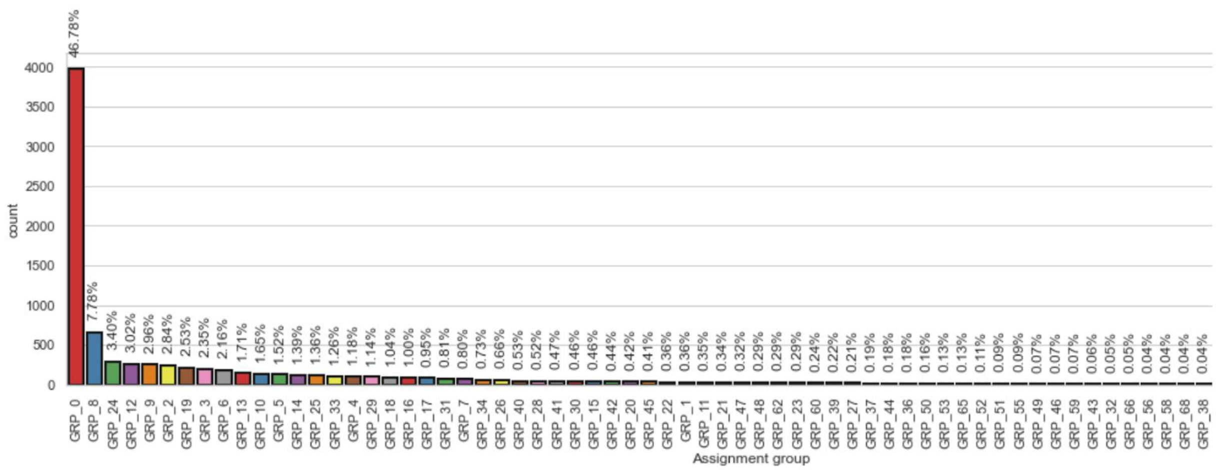
Junk text: **ç”µè„‘å¡ä¸”è¿žä¸ä¸Šå†…ç½‘**

Fixed text: **电脑卡且连不上内网**

We discovered this kind of brokeness is called Mojibake

- when someone has encoded Unicode with one standard and decoded it with a different one. This often shows up as characters that turn into nonsense sequences (called “mojibake”)

Fig 1. Distribution of Tickets amongst various groups:



As the figure 1 shows, Group 0 has the most number of tickets, followed by Group 8. The brief (problem statement) provided indicated that about 54% are resolved by L1/L2 groups. Since we also find from the above graph GRP_0 + GRP_8 also equals 54% roughly, we guess GRP_0 is L1 and GRP_8 is probably L2. The remaining groups are probably Functional/L3 teams. (The word cloud for GRP_8 also indicate this group could be related to monitoring tool OR job scheduler)

Some other interesting Figures are below

Fig 2.

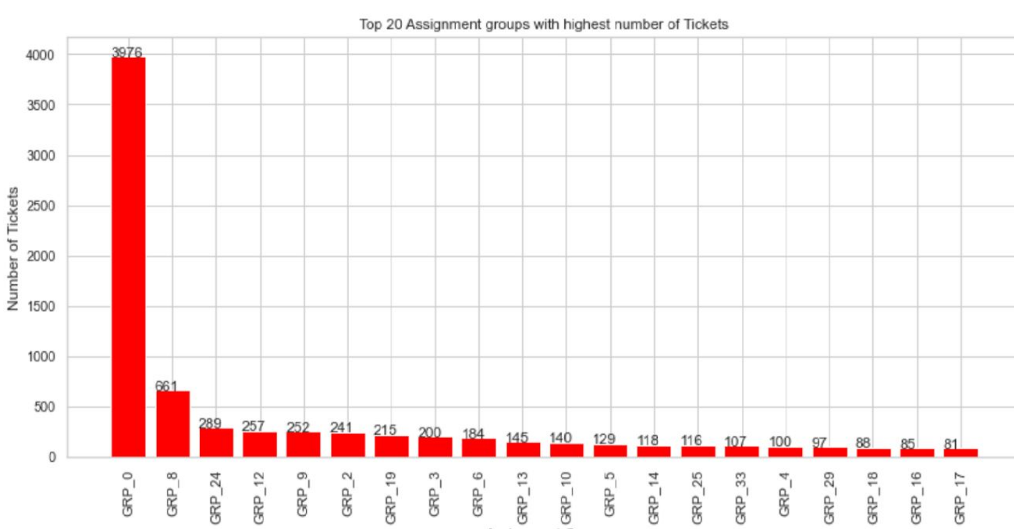


Fig 3.

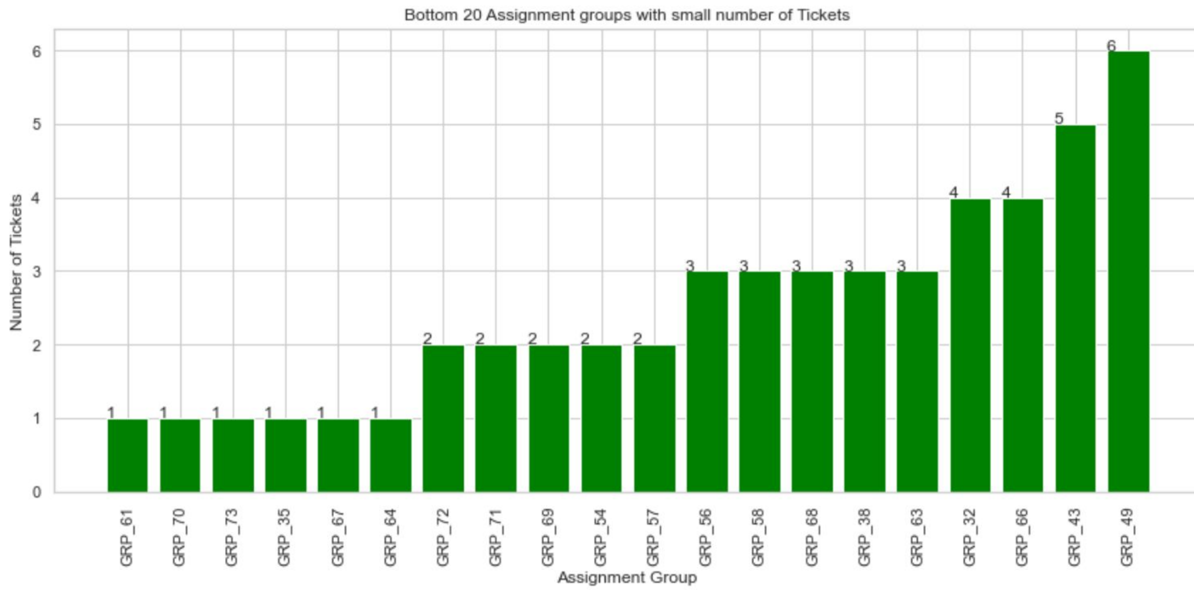

Bottom 20 Assignment groups with small number of Tickets
Number of Tickets
Assignment Group
GRP_61 GRP_70 GRP_73 GRP_35 GRP_67 GRP_64 GRP_72 GRP_71 GRP_69 GRP_54 GRP_57 GRP_56 GRP_58 GRP_68 GRP_38 GRP_63 GRP_32 GRP_66 GRP_43 GRP_49
1 1 1 1 1 1 2 2 2 2 2 3 3 3 3 3 4 4 5 6

Fig 4.

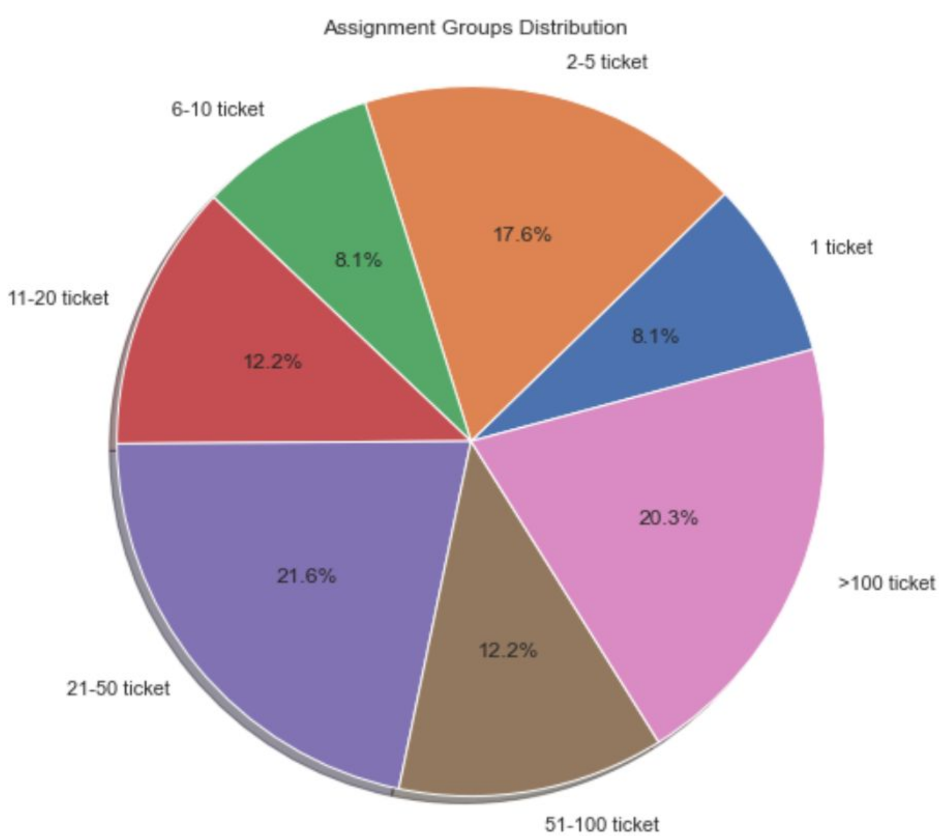

Assignment Groups Distribution
2-5 ticket
17.6%
6-10 ticket
8.1%
1 ticket
8.1%
11-20 ticket
12.2%
>100 ticket
20.3%
21-50 ticket
21.6%
51-100 ticket
12.2%

Fig 6a - Word Clouds for GRP_0

- These ticket descriptions indicate issues typically resolved by Help Desk using Standard Operating Procedures

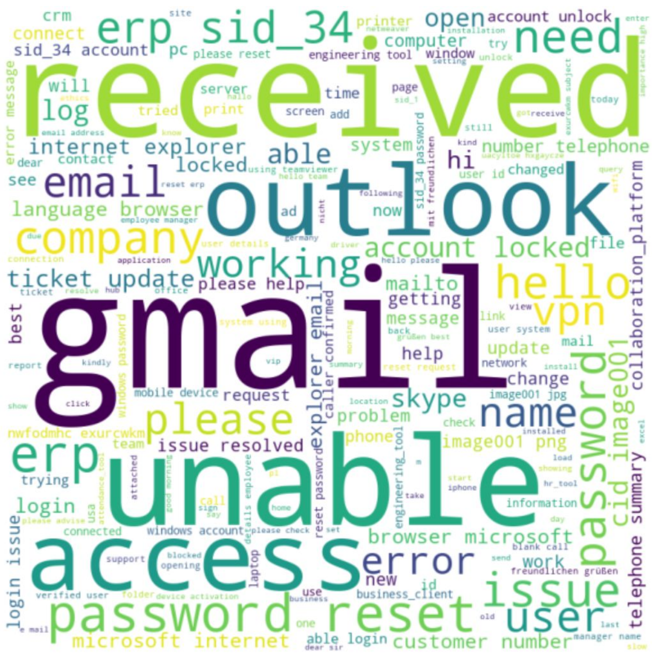


Fig 6b - Word Clouds for GRP_8

The words indicate issues reported by Monitoring tool OR tools like Job Scheduler

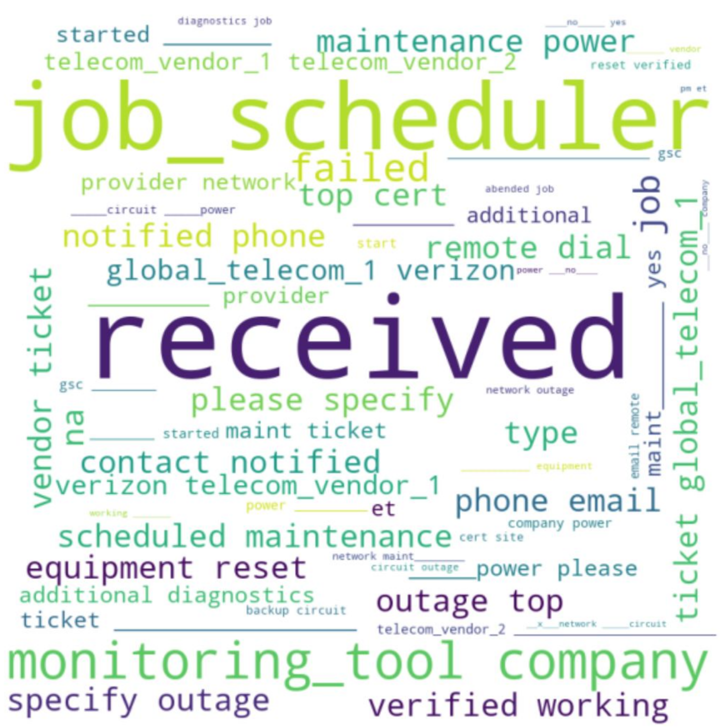
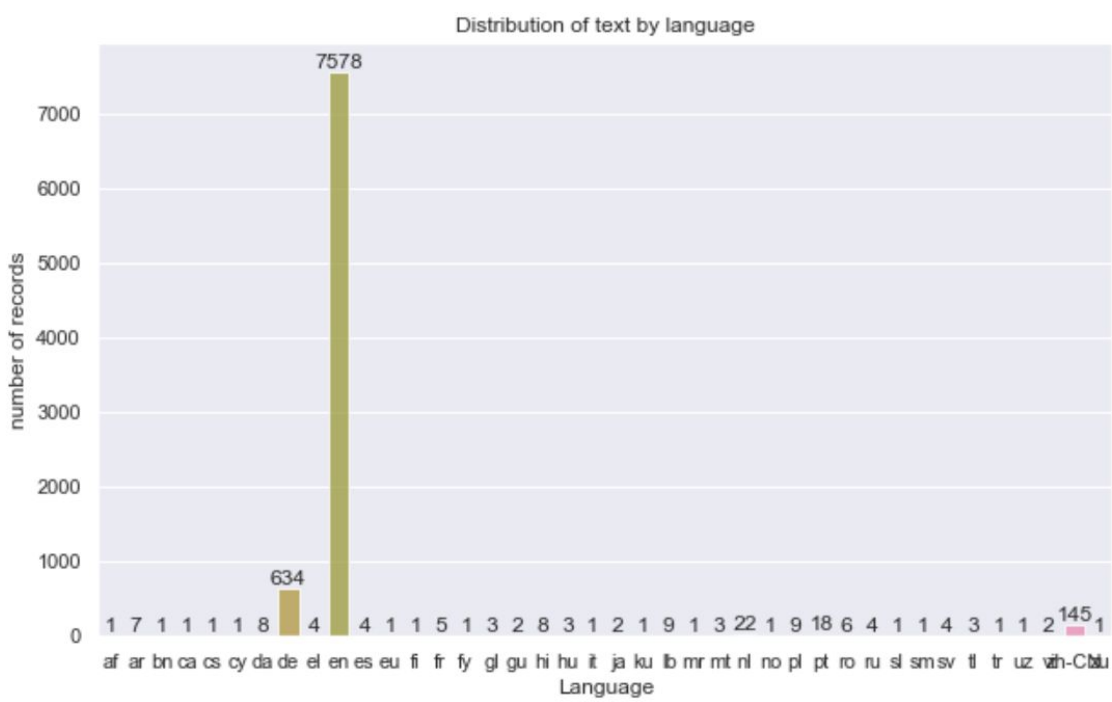


Fig 6c Word Cloud for GRP_24

GRP_24 Word Cloud led us into looking into Tickets with languages other than English. We used the Google Translation Engine to look into this further

As Fig 7 shows, most of the Tickets are in English followed by German and Simplified Chinese.



We also found in some tickets, the Short Description and Description were in different languages. An example below where the Description is in english and Shirt Description is in Portuguese

Fig 8.

	Short description	Description	Assignment group	Raw Combined description	raw_word_count	Combined description	language	language name	Translated Short description	
8498	machine não está funcionando	i am unable to access the machine utilities to...	GRP_62	machine não está funcionando i am unable to ac...	18	machine is not working i am unable to access t...	pt	portuguese	machine is not working	i am ur

Pre Processing

So we decided to translate the Short Description and Description fields separately first and then combine them for further preprocessing.

Fig 8 shows the Translated Short Description and Description field of a sample.

```
Short description                                    电话机没有声音
Description                                          电话机没有声音
Assignment group                                            GRP_30
Raw Combined description                             电话机没有声音
raw_word_count                                                   1
Combined description                       No sound from the phone
language                                                     zh-CN
language name                                  chinese (simplified)
Translated Short description              No sound from the phone
Translated Description                    No sound from the phone
```

We also found a couple of records where the Translator got confused so we fixed the data manually.

Fig 9 - In the sample below, the Translator mistook the sentences as Greek. We had to manually fix the data.

	Short description	Description	Assignment group	Raw Combined description	raw_word_count	Combined description	language	language name	Translated Short description	Translated Description
8043	setup new ws \xaqzisrk ahbgjrqz	setup new ws \xaqzisrk ahbgjrqz	GRP_24	setup new ws \xaqzisrk ahbgjrqz	5		el	greek	σετύπ νέω ως \ χάκζησρκ αχβγξρκζ	σετύπ νέω ως \ χάκζησρκ αχβγξρκζ
8072	setup new ws \pnwbkitv phbnwmkl	setup new ws \pnwbkitv phbnwmkl	GRP_24	setup new ws \pnwbkitv phbnwmkl	5		el	greek	σετύπ νέω ως \ πνωβκίτυ φβνωμκλ	σετύπ νέω ως \ πνωβκίτυ φβνωμκλ

We also fixed about 17 other records manually .

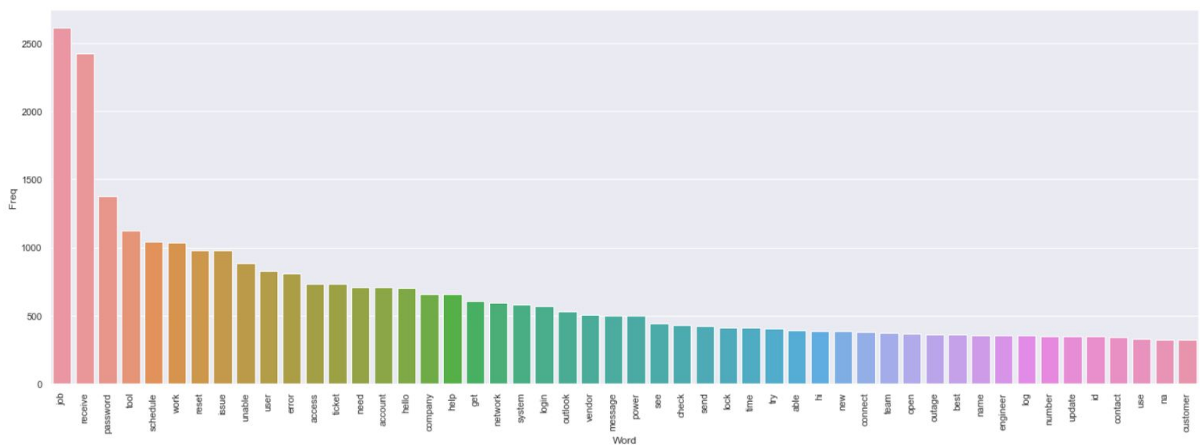
Dataset for Deep Learning

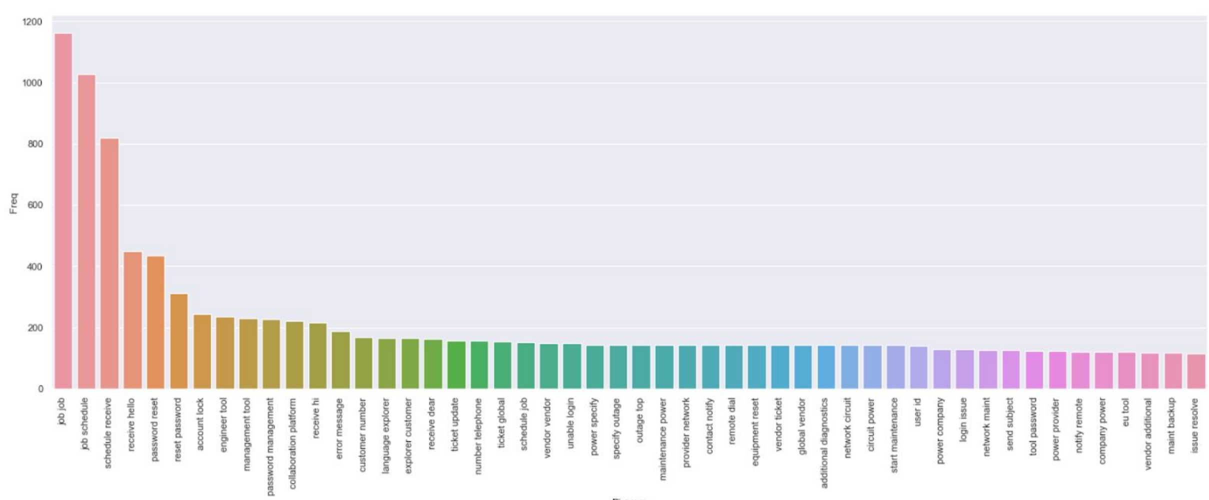
At this point, we created the dataset for Deep Learning. The reason we did this is because we wanted to retain the stop words and 'un lemmatized' words so the Deep Learning Algorithms will be able to process them in a context sensitive manner.

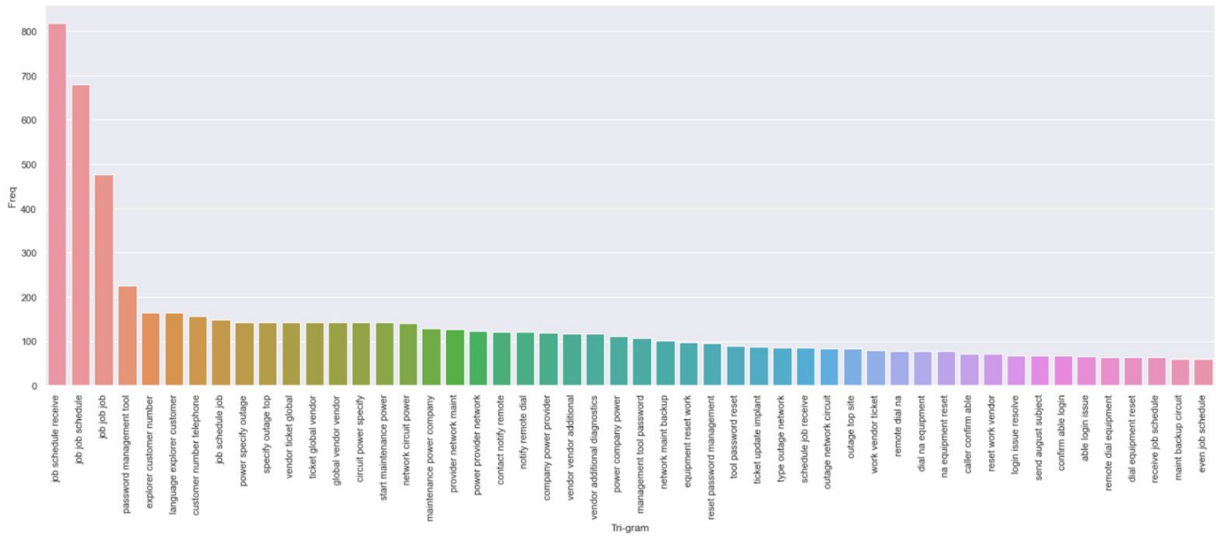
We then used the NLTK library for stopwords in English. We also used punkt and wordnet for lemmatization .

Total Corpus Word Count before lemmatization: 137508
Total Corpus Word Count after lemmatization: 82245

Fig 10. **Top 20 uni-grams, bi-grams & tri-grams**







After lemmatization and word - gram analysis we created the dataset required for the Machine Learning Models.

Deciding Models and Model Building

The dataset indicates a Multi-Classification problem with the 74 for labels. Our initial thinking is these models will be applicable in this class of problem.

Machine Learning Models:

1. Logistic Regression
2. KNN
3. Support Vector Machines
4. Ensembles - Bagging Classifier, Decision Tree, XGBoost

Deep Learning Models

1. DNN

2. RNN
3. GRU
4. LSTM

Transfer Learning Models

1. Transformers
2. BERT
3. ELMo

How to improve your model performance?

After machine learning, we ran an initial Model with Logistic Regression. It scored about 62% in accuracy and an f1 score of 57%

Fig 11.

	Algorithm	Accuracy	f1	Precision Score	Recall Score
1	LR 30% Raw Test Data after Cleansing	62.12%	56.91%	61.34%	67.72%

It became pretty clear immediately, we had do something to improve the scores.

We researched on Data Augmentation techniques and some pointers provided by our mentor proved very useful. We researched these techniques:

a. Translation based Data Augmention - we pick each ticket , translate it into random language and translate it back to English. While this was powerful, the amount of time it took was huge and it did NOT provide the volume of data needed for our scores to make difference

b. Spacy based technique - we used spacy to find synonyms of words. The technique is to substitute random words in the ticket with their synonyms. This again took too much time for spacy to generate synonyms, so we had to abandon it. We

We think above provide very good potential, but given the time we had we had to find another technique.

c. We used KNN to find the synonyms - These are technically not synonyms but the words are very close based on their 'distance' from the candidate word for which we are finding the synonym.

Examples:

'reset': ['disable', 'restart', 'manually', 'disconnect'],

'issue': ['problem', 'question', 'concern', 'concerned'],

'unable': ['cannot', 'able', 'failing', 'however'],

'you': ['sure', 'want', 'can', 'know'],

'have': ['they', 'could', 'already', 'would'],

'user': ['login', 'automatically', 'functionality', 'interface'],

'my': ['me', 'myself', 'own', 'got'],

'error': ['incorrect', 'invalid', 'problem', 'exception'],

'access': ['accessible', 'allow', 'provide', 'secure'],

'hello': ['hi', 'hey', 'yes', 'dear'],

'be': ['should', 'being', 'not', 'will'],
'account': ['payment', 'personal', 'same', 'however'],

d. We also used Parts of Speech tagging in a similar way to identify synonyms.

Examples:

'trying': ['try'],

'working': ['work'],

'home': ['house'],

'unlocking': ['unlock'],

'locked': ['lock'],

'getting': ['get'],

'need': ['want'],

'dynamics': ['dynamic'],

'regarding': ['regard'],

'phone': ['telephone'],

'client': ['customer'],

'duplication': ['duplicate'],

'gentle': ['soft'],
'display': ['showing'],

Because using KNN technique, we could find multiple synonyms for a word, we settled on using the synonyms from KNN to create Tickets where we subtitied a random word in the ticket with its synonym keeping the label the same.

For example, in the below sample, the word accessible was changed to reachable, so the same meaning is retained yet it help us create a new ticket for the same group.

accessible hand language icon customer number telephone	GRP_3
reachable hand language explorer customer number telephone	GRP_3

We also excluded the majority group GRP_0 and concentrated on increasing the number of records for the minority groups.

Fig 12 - Distribution before Augmentation

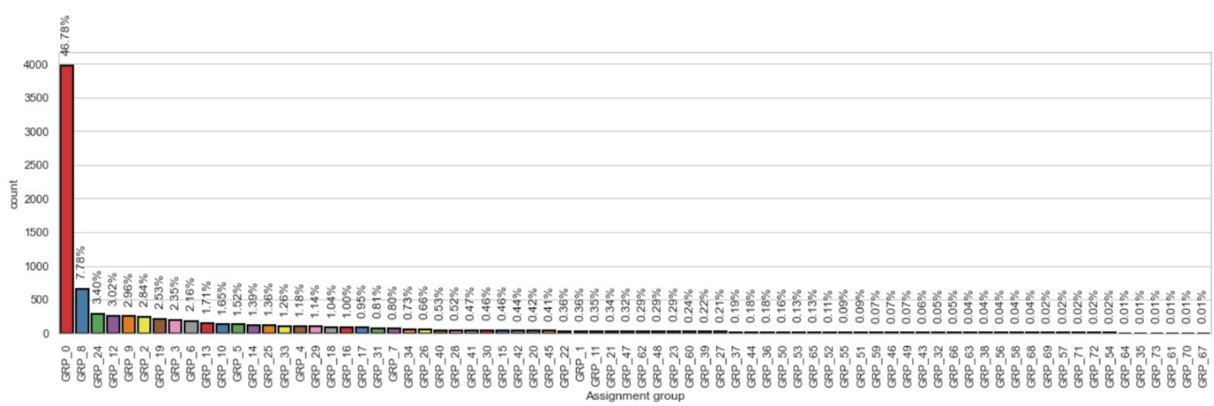
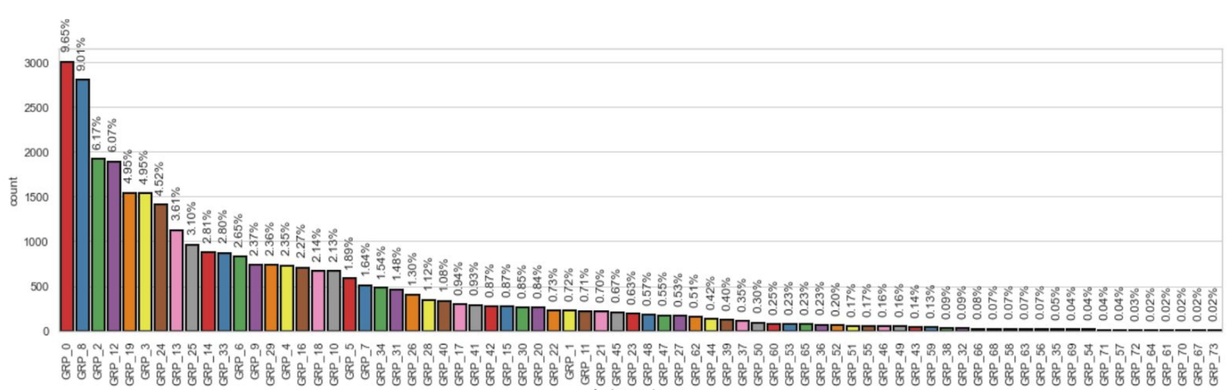


Fig 13 - Distribution after Augmentation



As the graphs indicate, we were able to significantly increase the representation of the minority groups. The initial results are promising.

	Algorithm	Accuracy	f1	Precision Score	Recall Score
1	SVC 30% Test Data after Data Augmentation Word Embedding Round 2	91.41%	91.42%	94.13%	89.94%
2	SVC 30% Test Data after Data Augmentation Word Embedding Round 1	86.26%	86.38%	88.67%	85.84%
3	LR 30% Test Data after Data Augmentation Word Embedding Round 2	76.58%	76.32%	78.94%	77.05%
4	LR 30% Test Data after Data Augmentation Word Embedding Round 1	64.02%	63.26%	70.10%	65.48%
5	LR 30% Raw Test Data after Cleansing	62.12%	56.91%	61.34%	67.72%

We were able to increase the Accuracy of Logistics Regression from 62% to 76.58% after a couple of rounds of Data Augmentation. SVC topped the charts with 91.41 %.

We also ran the ensembles but their results were not too impressive. We will need to look into the reasons. Attached box plot shows the different Machine Learning Algorithms. The SVC & LR score the highest