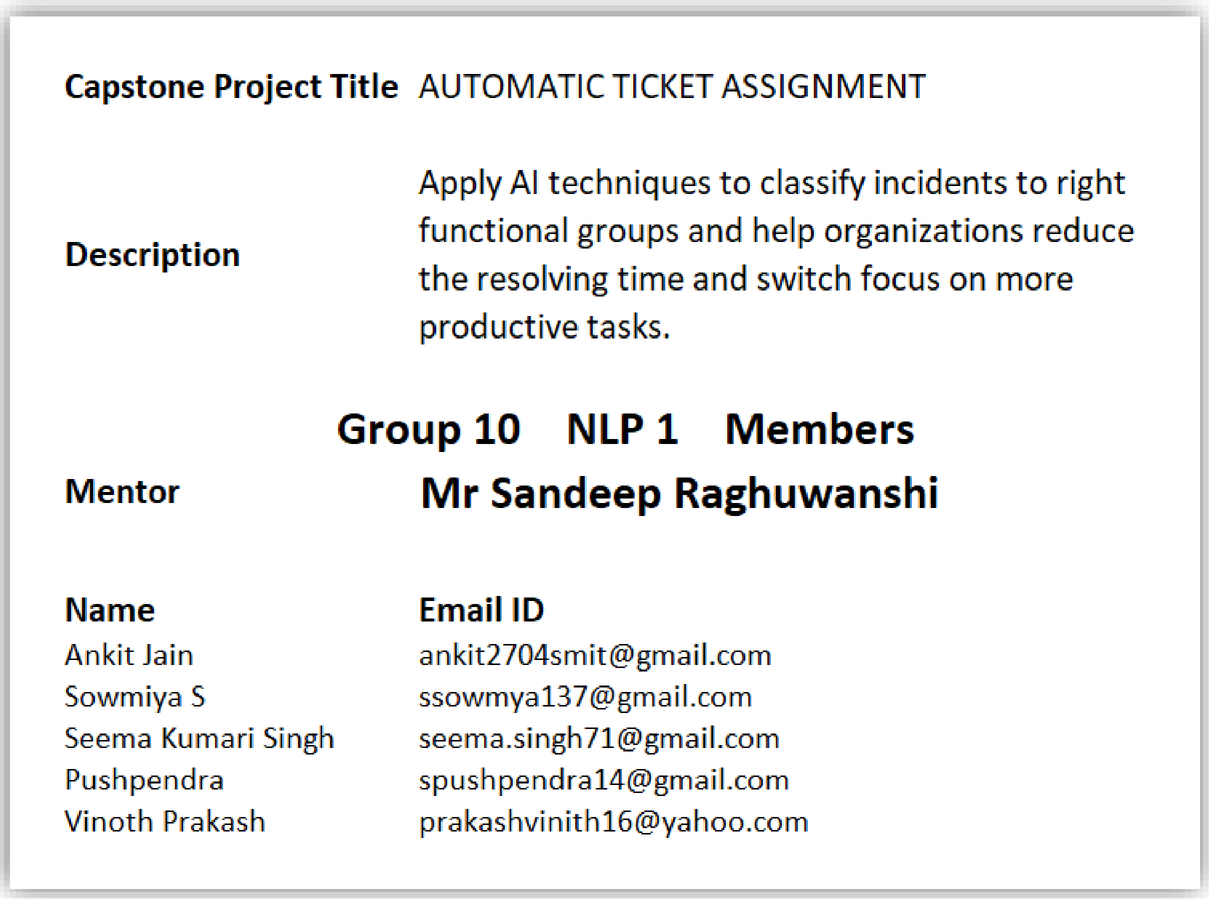
Automatic Ticket Assignment Capstone Project Report

AIML

Index

Team Details



Capstone Project Title AUTOMATIC TICKET ASSIGNMENT

Description Apply AI techniques to classify incidents to right functional groups and help organizations reduce the resolving time and switch focus on more productive tasks.

Group 10 NLP 1 Members

Mentor **Mr Sandeep Raghuwanshi**

Name	**Email ID**
Ankit Jain	ankit2704smit@gmail.com
Sowmiya S	ssowmya137@gmail.com
Seema Kumari Singh	seema.singh71@gmail.com
Pushpendra	spushpendra14@gmail.com
Vinoth Prakash	prakashvinith16@yahoo.com

Summary of the problem statement, Data and findings

Understanding the Business

IT leverages Incident Management process to ensure there is no disruption to business operations. Any unplanned disruption can cause interruption to business services. Incident management process helps in identification of issues or problems faced by users or operation teams. Manual assignment of incidents is time consuming and requires human intervention. There may be lag in incident resolution due to human errors or if incidents are not routed appropriately. On the other hand, manual assignment also increases the response and resolution times which result in user satisfaction deterioration / poor customer service.

Risks Involved

The manual assignment of these incidents might have below disadvantages:

❖ More resource usage and expenses.

❖ Human errors - Incidents get assigned to the wrong assignment groups

❖ Delay in assigning the tickets

❖ More resolution times

❖ If a particular ticket takes more time in analysis, other productive tasks get affected for the Service Desk

Background and Objective

To apply techniques and learnings to make ticket assignment more cost-effective, less resolution time so that service desk team can focus on other productive tasks. We are able to see that the current system is capable of assigning 70+% of the tickets correctly. Our target is to automatically classify tickets and directing them to appropriate groups at the earliest, helps in improving the throughput in the ticketing pipeline of an organization.

Data & Findings

Data format	CSV
Total Records	8500

Data Fields

Short description	A brief overview of the issue faced by the user
Description	Detailed description of the issue
Assignment group	GRP_0 ~ GRP_73 (total 74 classes of Assignment group)

Sample data

Short description	**Description**	**Assignment group**
login issue	-verified user details.(employee# & manager name) -checked the user name in ad and reset the password. -advised the user to login and check. -caller confirmed that he was able to login. -issue resolved.	GRP_0
Outlook	received from: hmjdrvpb.komuaywn@gmail.com hello team, my meetings/skype meetings etc are not appearing in my outlook calendar, can somebody please advise how to correct this? kind	GRP_0

Short description	**Description**	**Assignment group**
cant log in to vpn	received from: eylqgodm.ybqkwiam@gmail.com hi i cannot log on to vpn best	GRP_0

Observations

1. High imbalance seen in data with GRP_0 having 40%+ percent of representation
2. Many groups/classes are with very little representation.
3. Null values:

Short description	8
Description	1
Assignment group	0

4. Very few tickets have non-English descriptions
5. Four columns – Short Description, Description, Caller and Assignment group
6. 74 Assignment groups found - Target classes
7. Caller names in a random fashion (may not be useful for training data)
8. European non-English language also found in the data
9. Email/chat format in description
10. Symbols & other characters in the description
11. Hyperlinks, URLS & few image data found in the description
12. Blanks found either in the short description or description field
13. Few descriptions same as the short description
14. Few words were combined together
15. Spelling mistakes and typo errors are found

Summary of the approach to EDA and Pre-Processing

Data Pre-Processing and Cleaning

Below steps have been performed for initial pre-processing and clean-up of data:

1. Dropped the caller field as the data was not found to be useful for analysis
2. Replaced Null values in short description & description with space.
3. Merged Short Description & Description fields for analysis
4. Contraction words found in the merged Description are removed for ease of word modelling
5. Changed the case sensitivity of words to lower case
6. Removed Hashtags, Hyperlinks, URLs, HTML tags, Emoji and non-ASCII symbols from merged fields.
7. Translating all languages (German) to English
8. Tokenization of merged data
9. Removal of Stop words and Meaningless words
10. Lemmatization
11. WordCloud created
12. Attempted to do spell check
13. Created Plot to understand the distribution of words
14. Removal of line breaks and tabs (\r\n\t)
15. Removal of special characters
16. Removal of punctuation
17. Removal of extra spaces
18. Missing value imputation

Data cleaning

Data cleaning has been taken care by splitting the various functions which would remove and clean any unwanted or misleading information.

All words have been converted to lower case.
Header and sender information from the emails have been removed.

All the numbers, non-dictionary characters, newline characters, hashtag, HTML entities, emojis, hyperlinks, extra spaces and unreadable characters have been removed.
We have ensured to remove any caller names included in the description column.

Lemmatization & Stop words removal

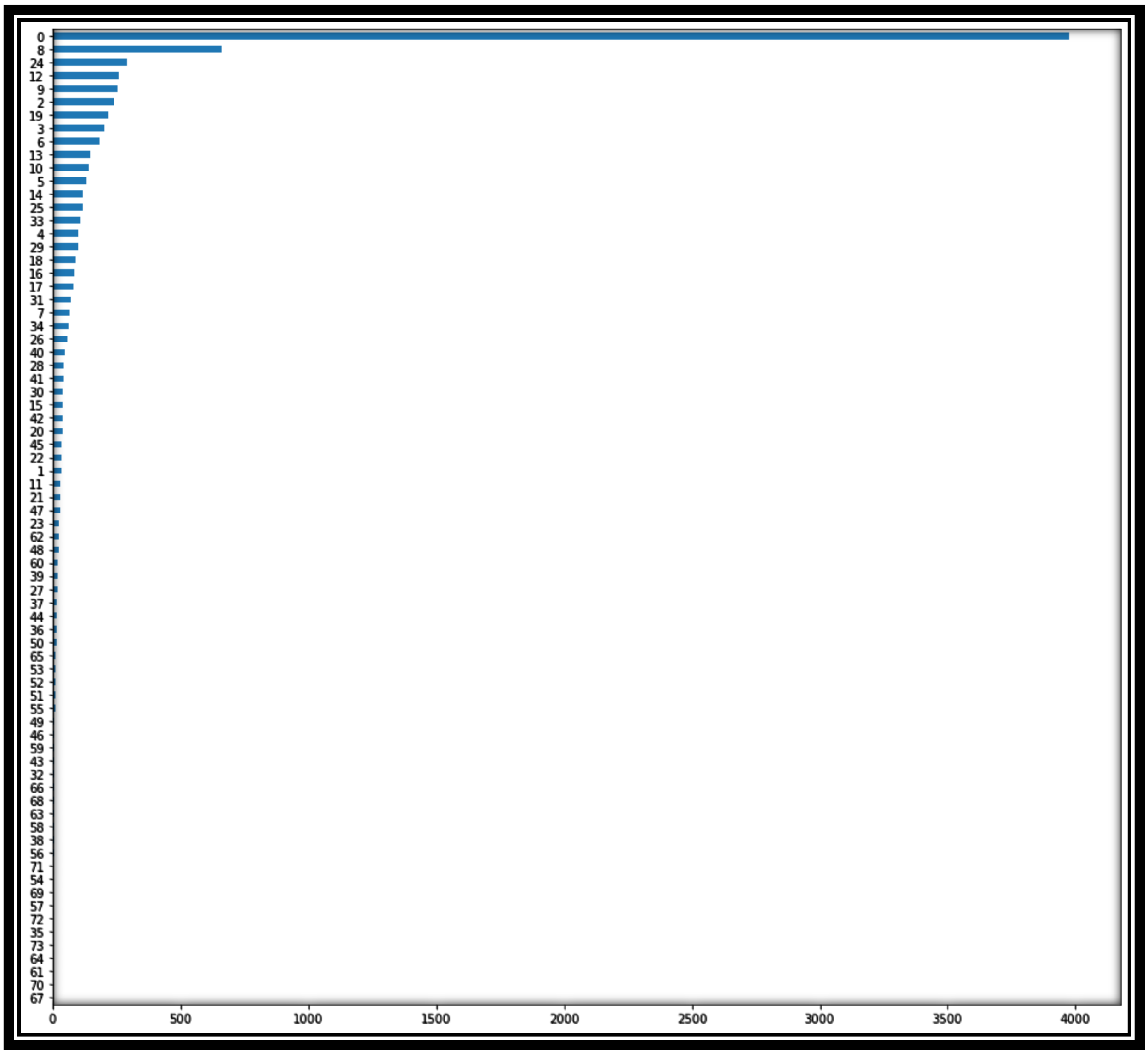
Stop words have been removed using nltk corpus modules.

Lemmatization is the process of grouping together the different inflected forms of a word so they can be analyzed as a single item. Lemmatization is similar to Stemming but it brings context to the words. So, it links words with similar meanings to one word.

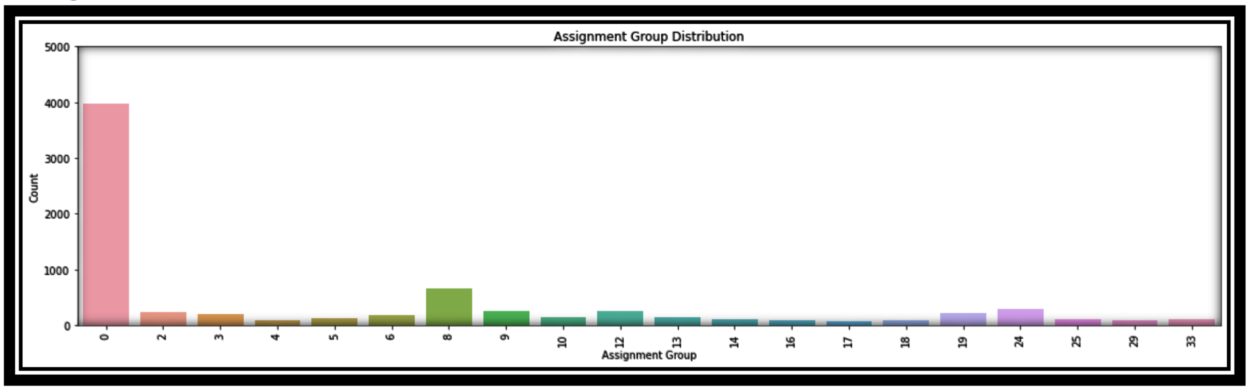
We have preferred Lemmatization over Stemming because lemmatization does morphological analysis of the words

Data Visualization

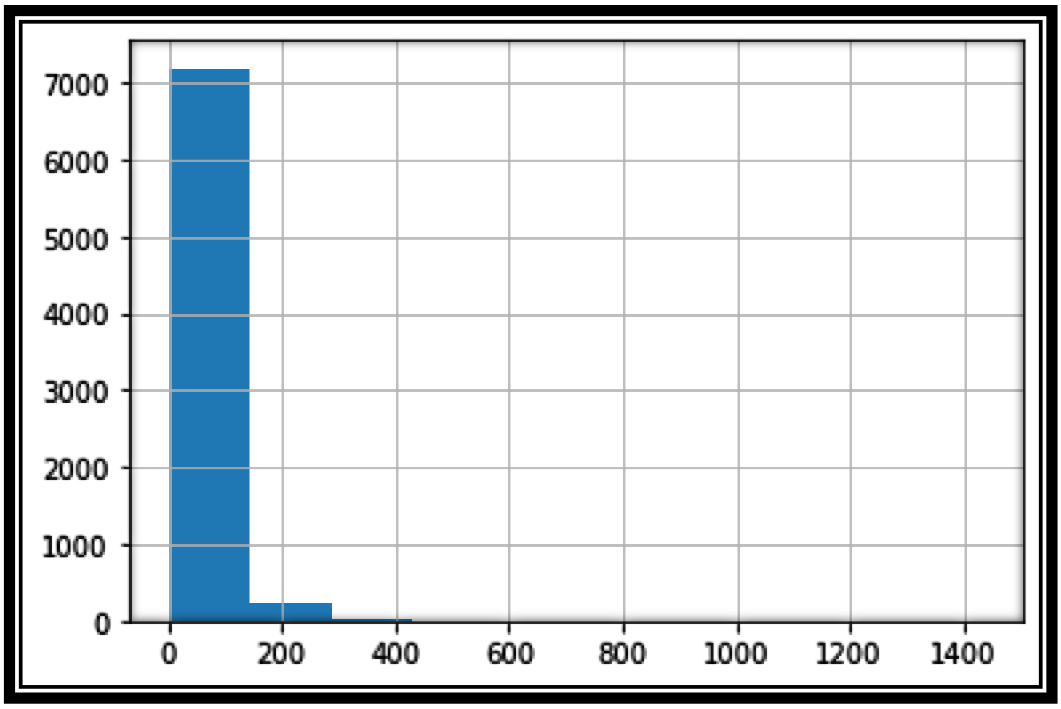
High Imbalance



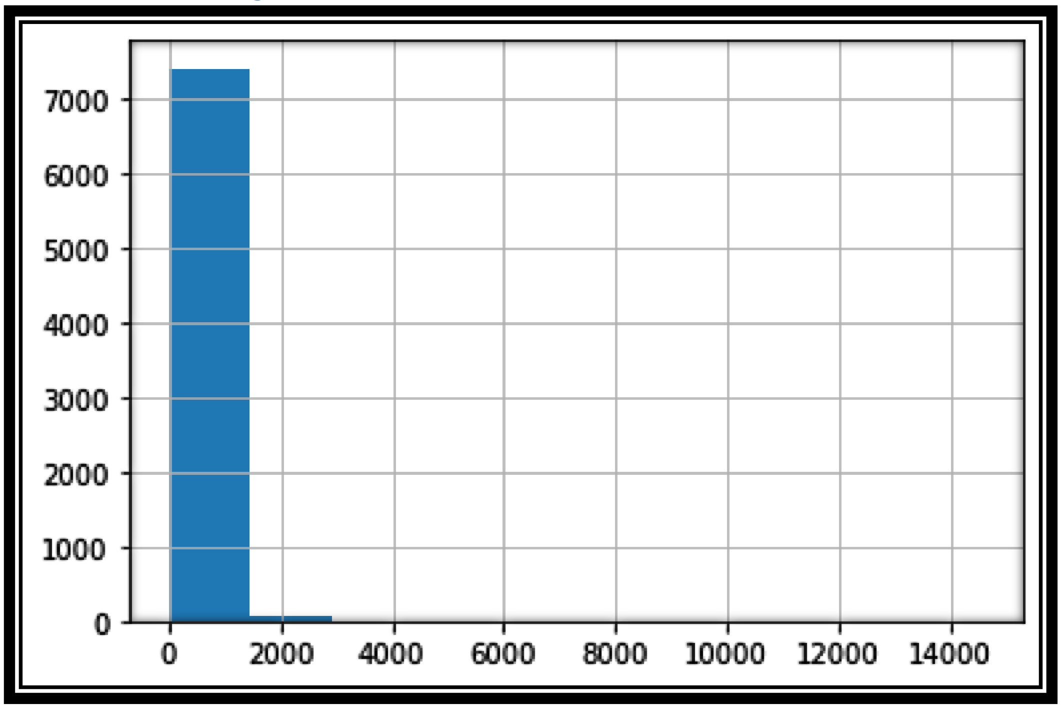
Assignment Group Distribution



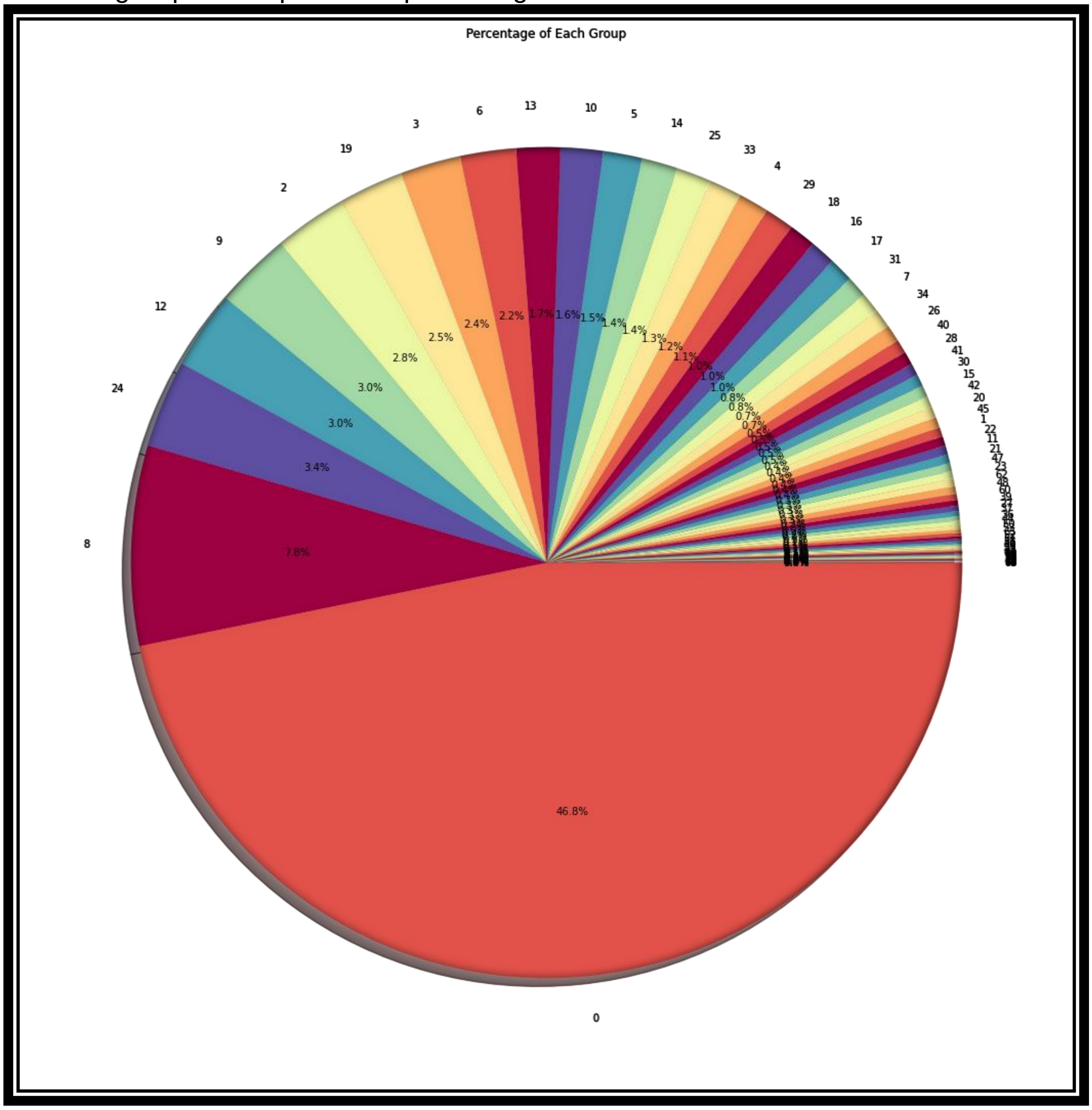
Word Length Distribution



Character Length Distribution

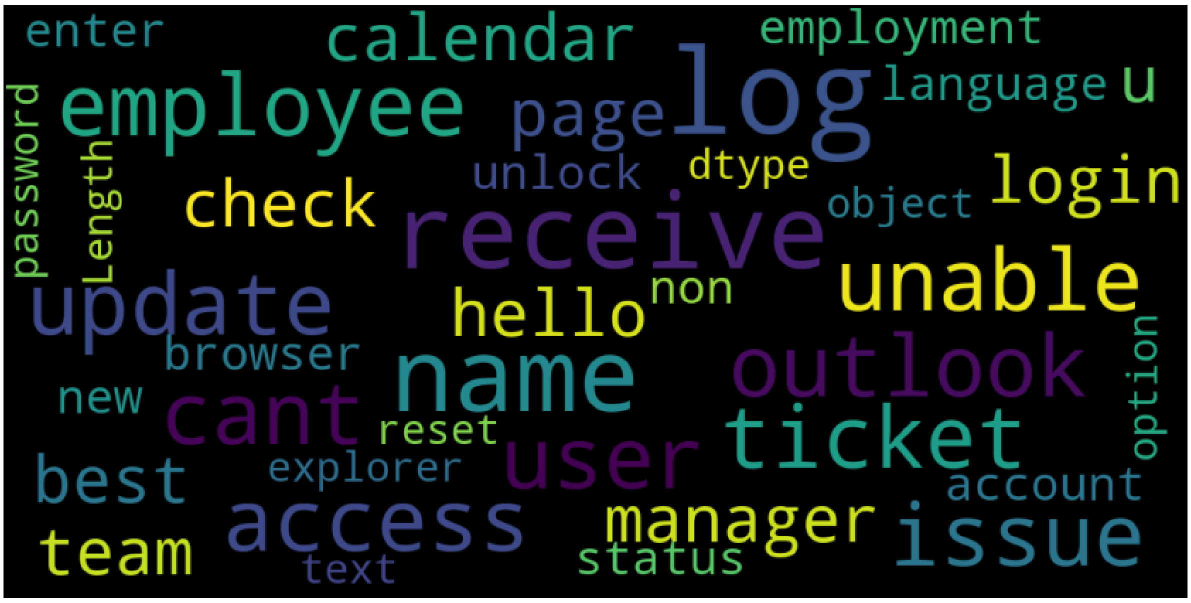


Percentage Split of Top 20 Groups Having 80+% of records

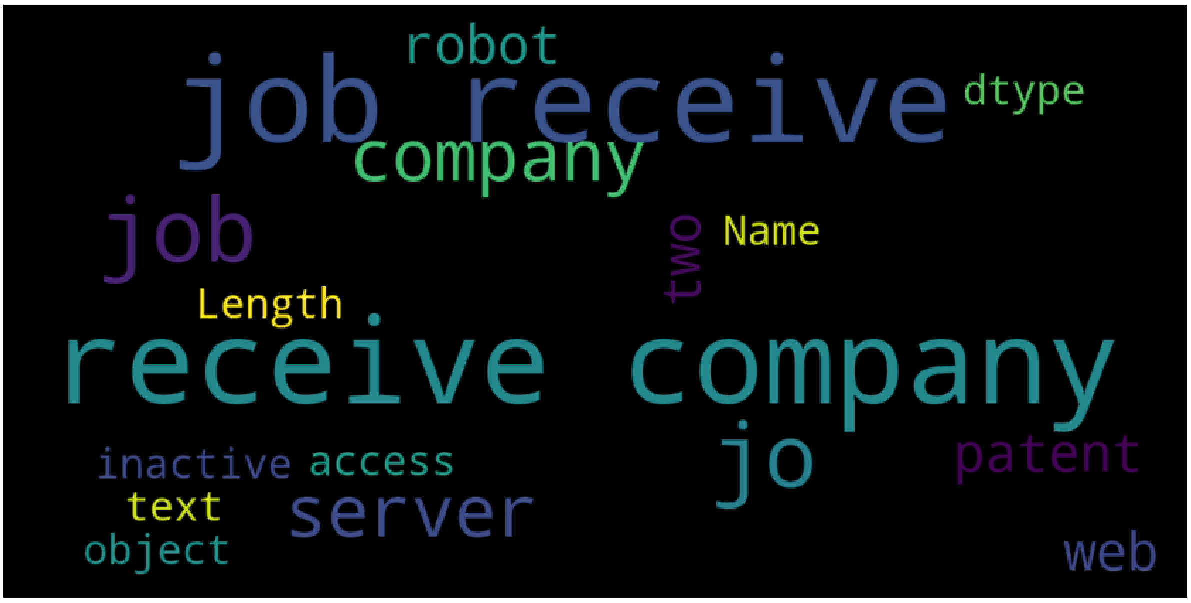


Word Cloud After Data Pre-processing and Cleaning

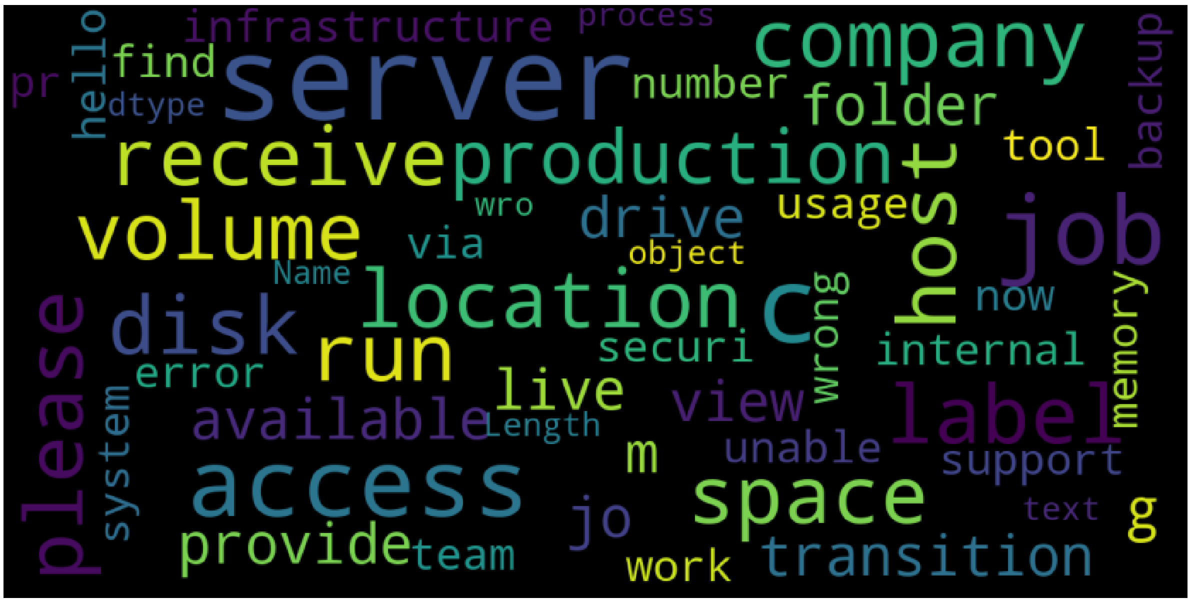
Grp_0



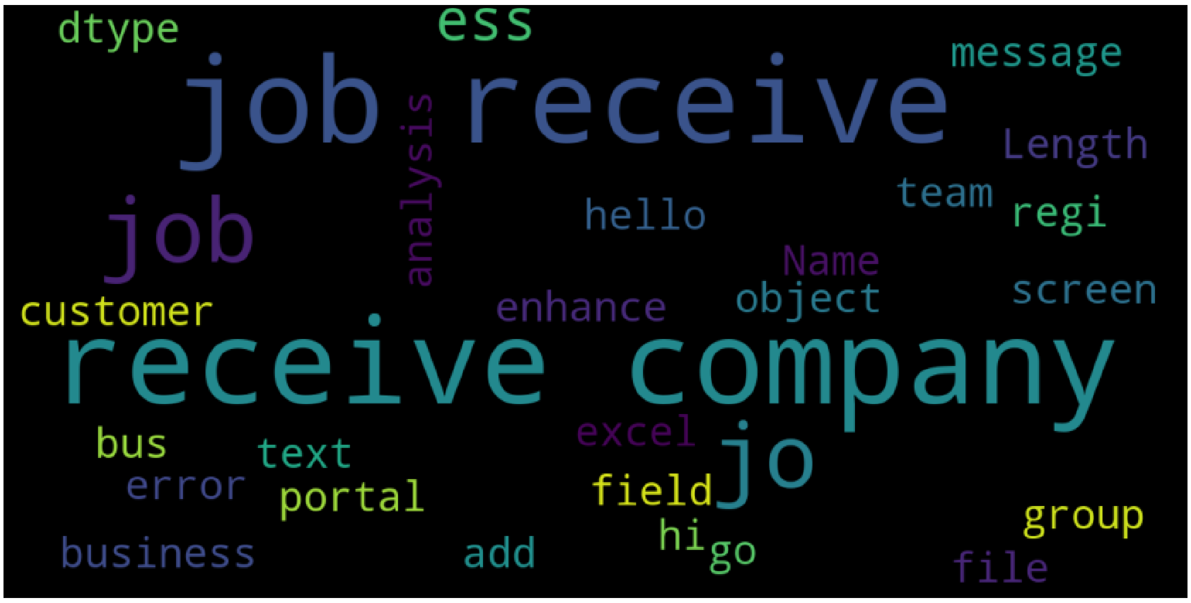
Grp_8



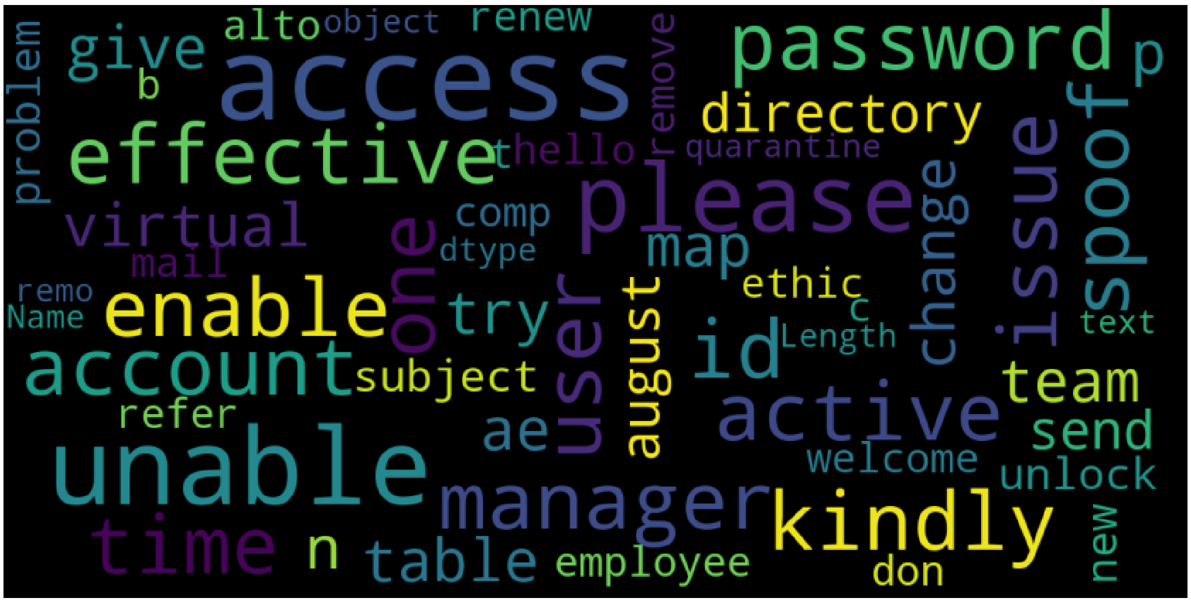
Grp_12


server
company
receive
production
volume
job
location
disk
run
host
label
please
access
space
available
provide
transition
infrastructure
folder
number
drive
tool
usage
live
view
securi
internal
error
unable
support
system
hello
find
dtype
backup
memory
wrong
now
object
via
wro
Name
Length
team
work
jo
text
process
pr
m
g
c

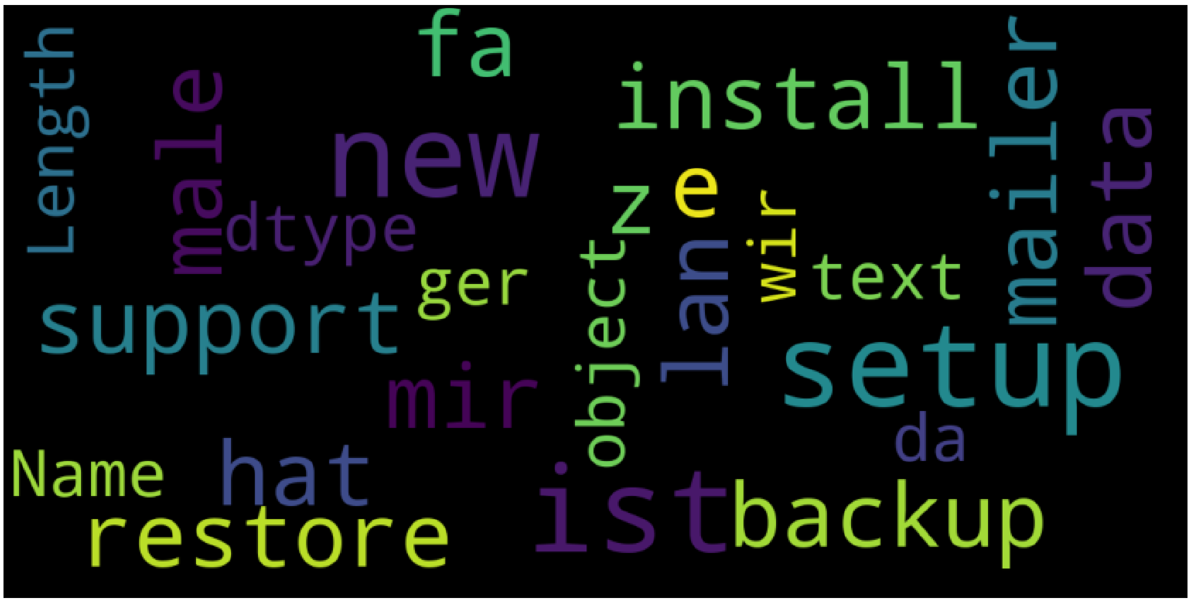
Grp_09


job receive
receive company
job
jo
dtype
ess
message
analysis
Length
hello
team
regi
Name
customer
enhance
object
screen
bus
text
excel
error
portal
field
group
business
add
hi
go
file

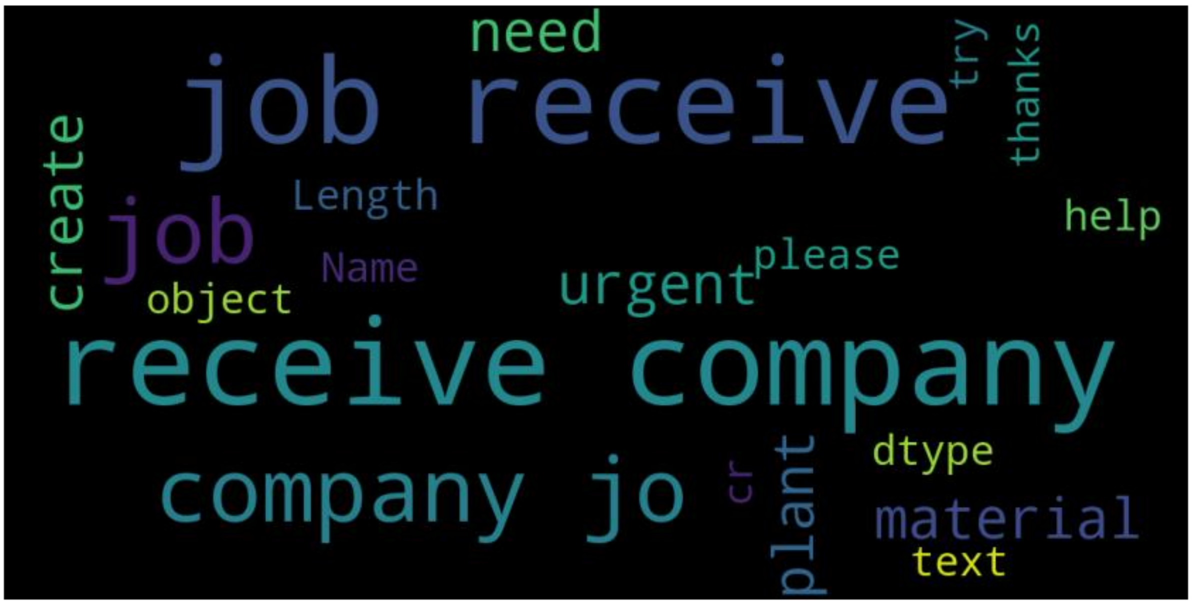
Grp_02


access
password
please
unable
effective
account
enable
manager
kindly
active
user
issue
spoof
directory
virtual
time
table
give
one
try
id
change
august
team
send
subject
unlock
welcome
map
employee
problem
alto
object
renew
remove
hello
quarantine
comp
dtype
mail
remo
Name
ethic
Length
text
refer
ae
n
don
new
b
p
c

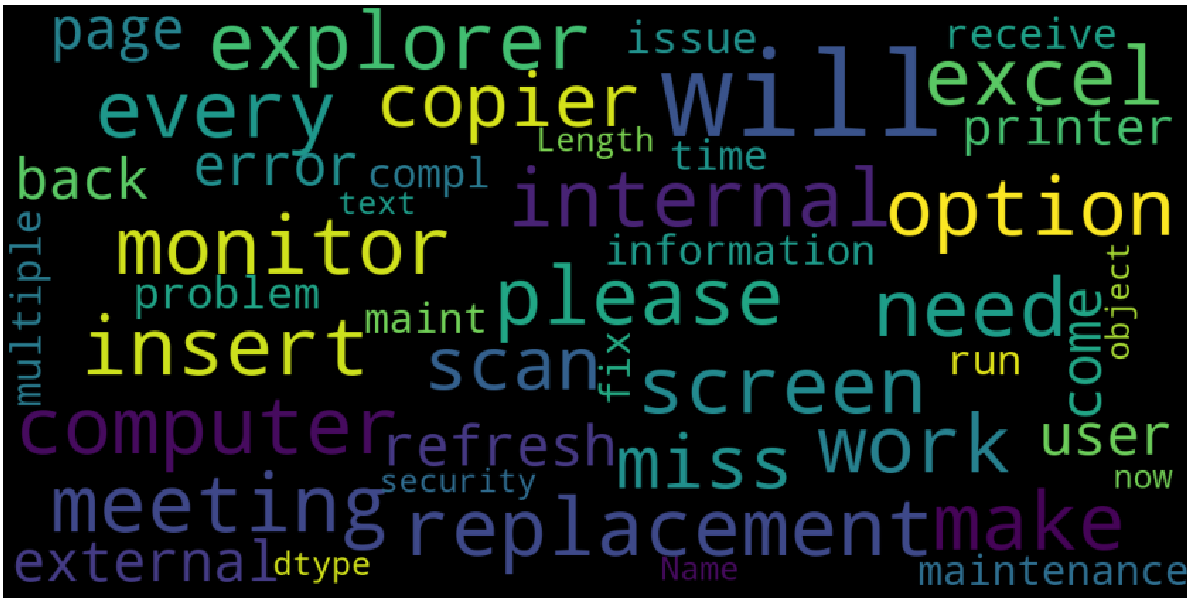
Grp_24


new
setup
install
support
restore
backup
ist
hat
mailer
data
male
Length
dtype
fa
ze
lan
object
wir
text
ger
mir
da
Name

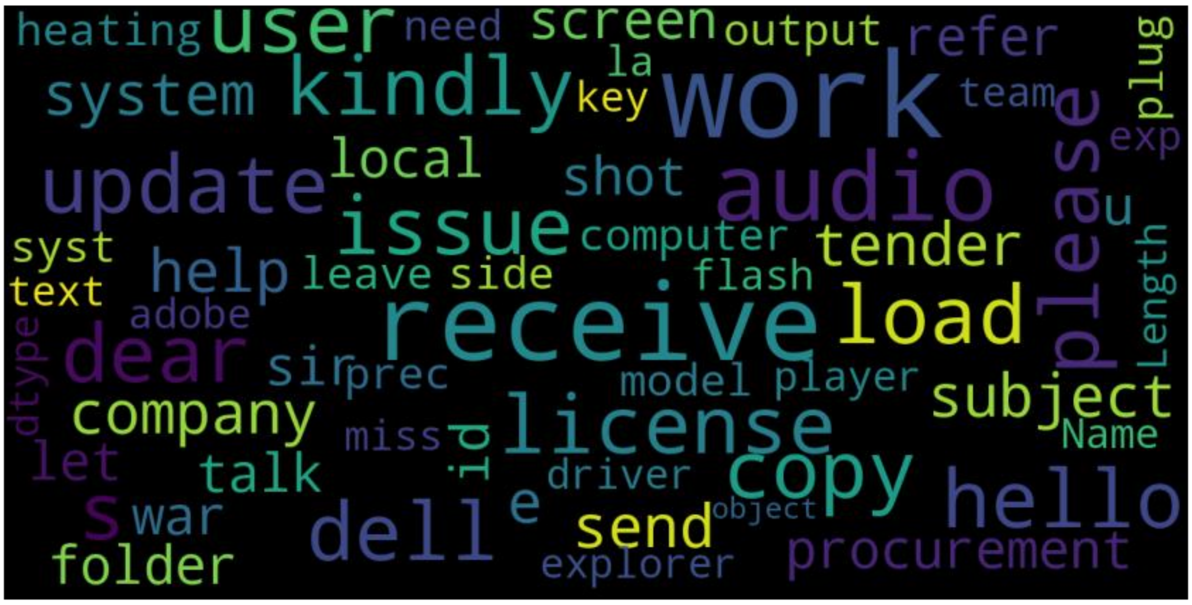
Grp_06


job receive
receive company
company jo
job
create
need
try
thanks
Length
help
Name
please
urgent
object
cr
plant
dtype
material
text

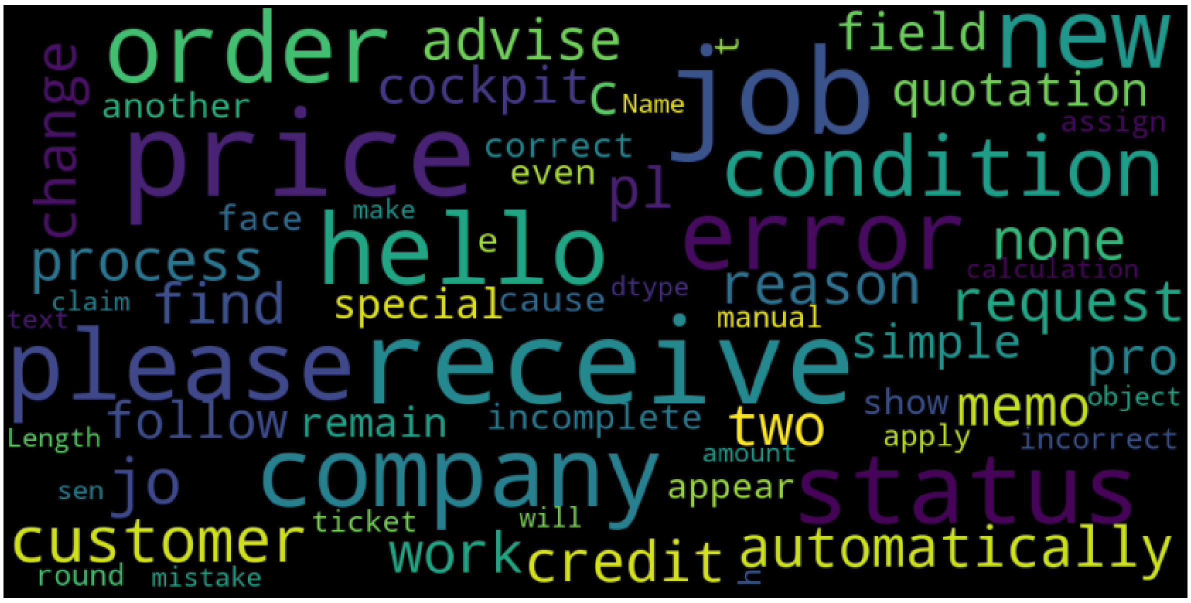
Grp_03


page
explorer
issue
receive
will
excel
every
copier
printer
Length
time
back
error
compl
text
internal
option
monitor
information
multiple
problem
please
need
object
insert
maint
scan
fix
screen
run
come
computer
refresh
miss
work
user
security
now
meeting
replacement
make
external
dtype
Name
maintenance

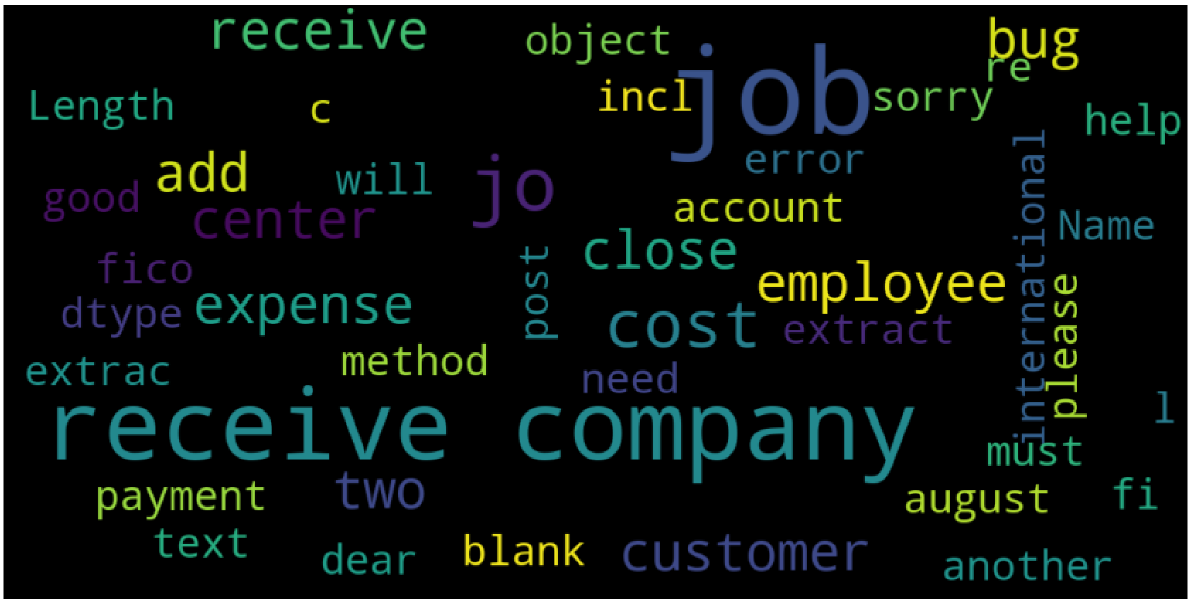
Grp_19


heating
user
need
screen
output
refer
plug
system
kindly
la
key
work
team
exp
update
local
shot
audio
please
u
issue
computer
tender
syst
help
leave
side
flash
Length
text
adobe
receive
load
dtype
dear
sir
prec
model
player
subject
company
miss
id
license
Name
let
talk
driver
copy
s
war
e
object
hello
dell
send
folder
explorer
procurement

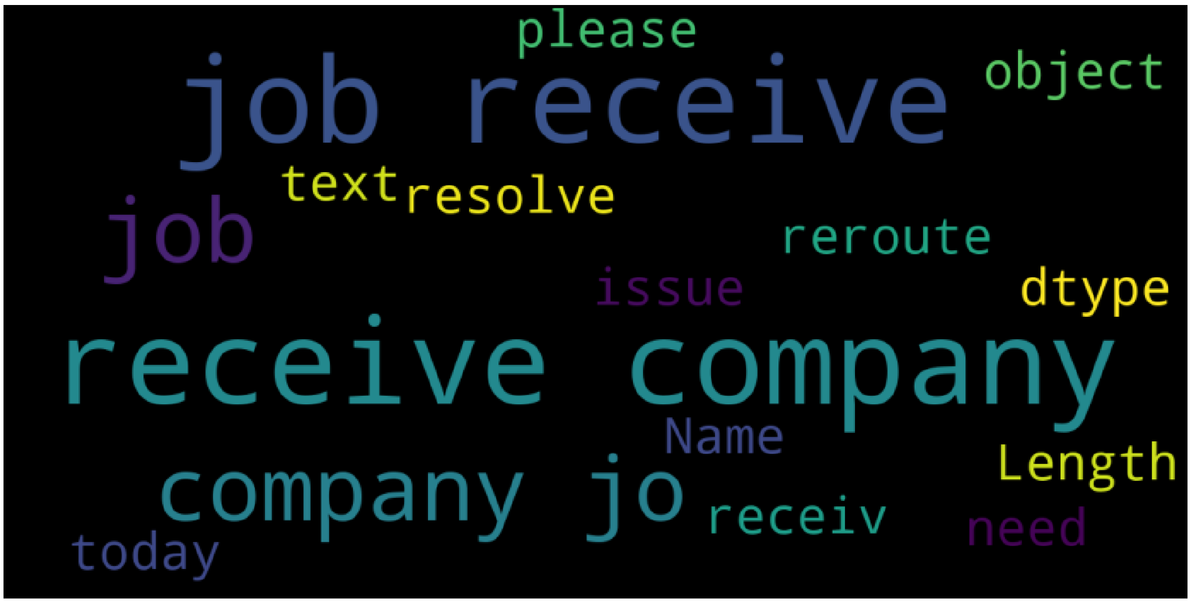
Grp_13


change
order
advise
field
new
another
cockpit
c
Name
job
quotation
assign
price
correct
even
pl
condition
face
make
process
hello
e
error
none
calculation
claim
find
special
cause
dtype
reason
request
text
manual
please
receive
simple
pro
follow
remain
incomplete
show
memo
object
Length
two
apply
incorrect
amount
sen
jo
company
appear
status
ticket
will
customer
work
credit
automatically
round
mistake

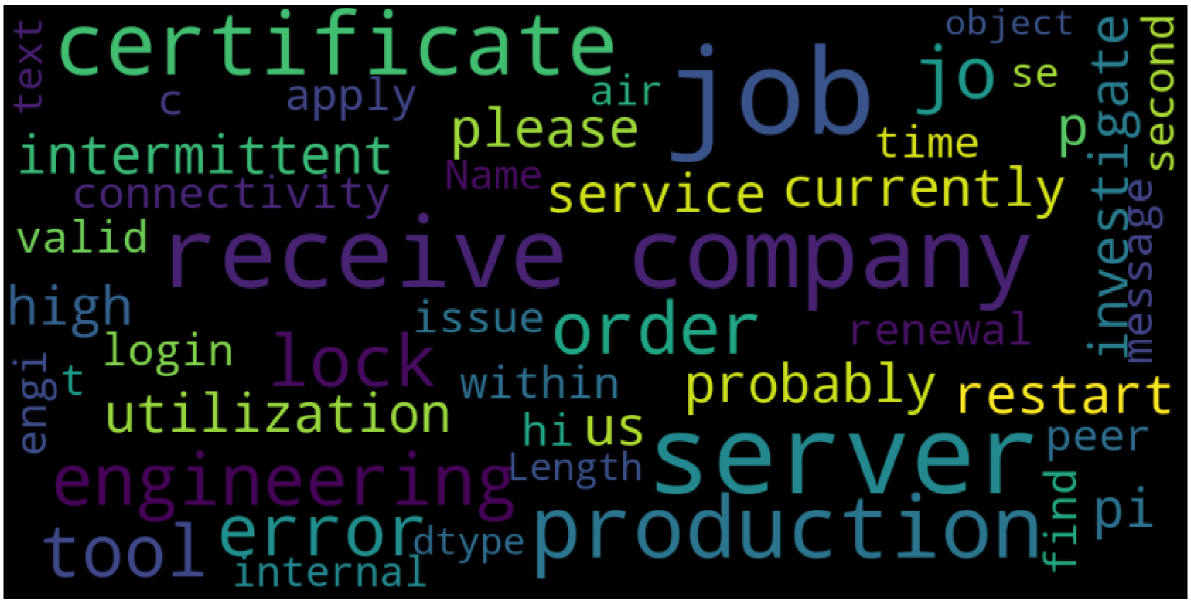
Grp_10


receive
object
job
bug
re
Length
c
incl
sorry
help
add
will
jo
error
international
good
center
account
Name
fico
close
post
employee
please
dtype
expense
cost
extract
extrac
method
need
receive company
l
must
payment
two
august
fi
text
dear
blank
customer
another

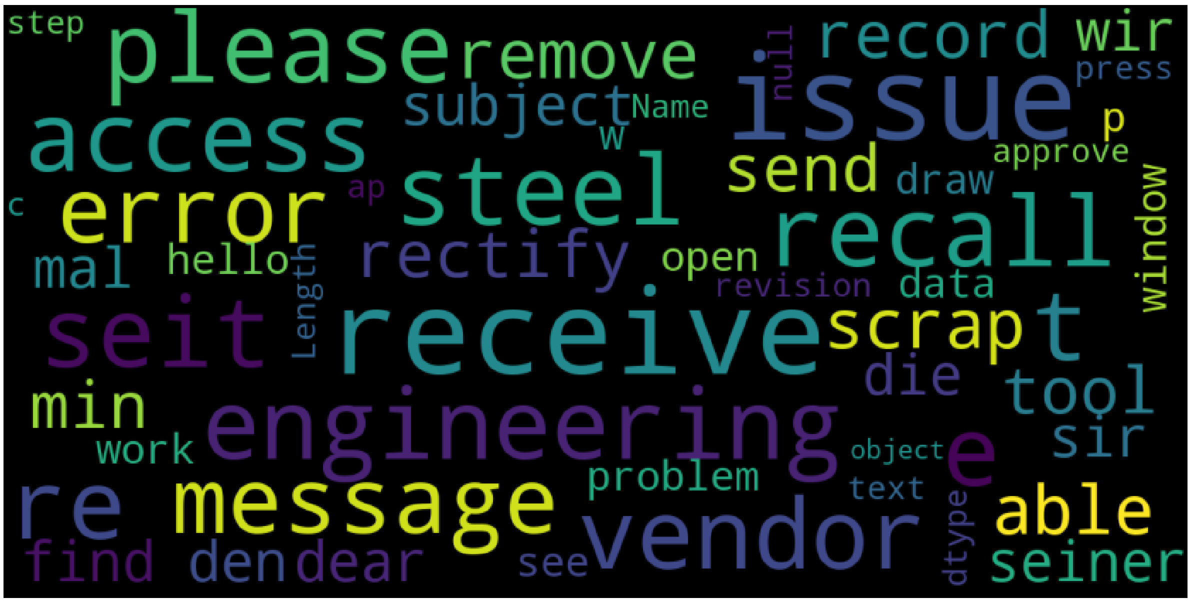
Grp_05


please
job receive
object
text
resolve
job
reroute
issue
dtype
receive company
Name
Length
company jo
receiv
need
today

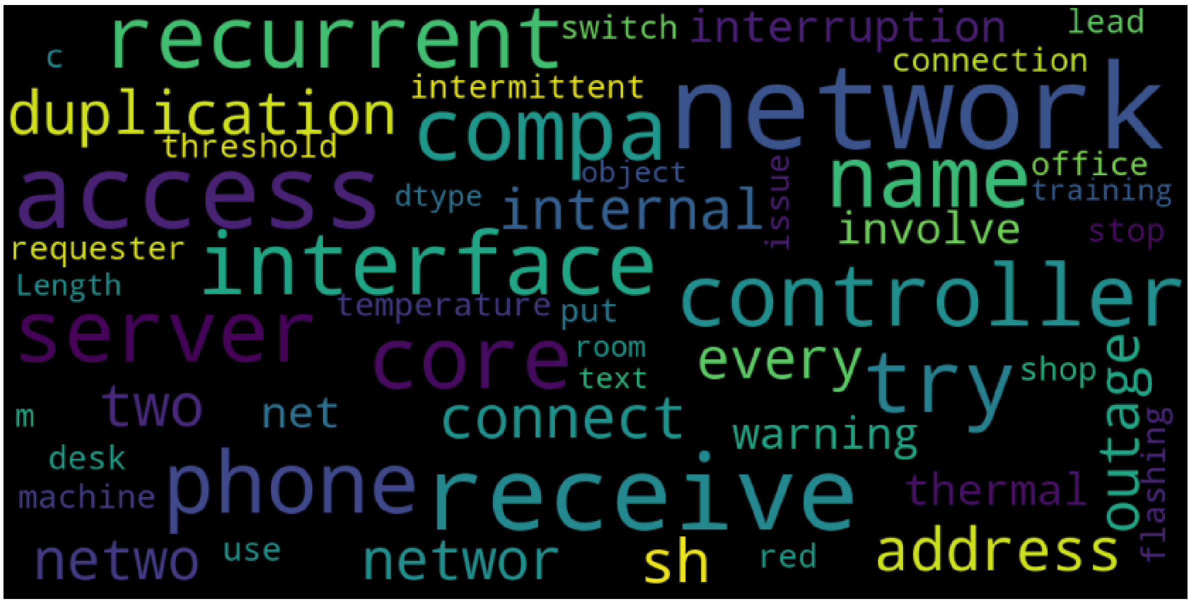
Grp_14


text
certificate
job
jo
se
object
investigate
second
message
c
apply
air
please
time
p
intermittent
Name
service
currently
connectivity
valid
receive
company
high
issue
order
renewal
login
lock
engi
t
within
probably
restart
utilization
hi
us
server
peer
engineering
Length
find
pi
tool
error
dtype
production
internal

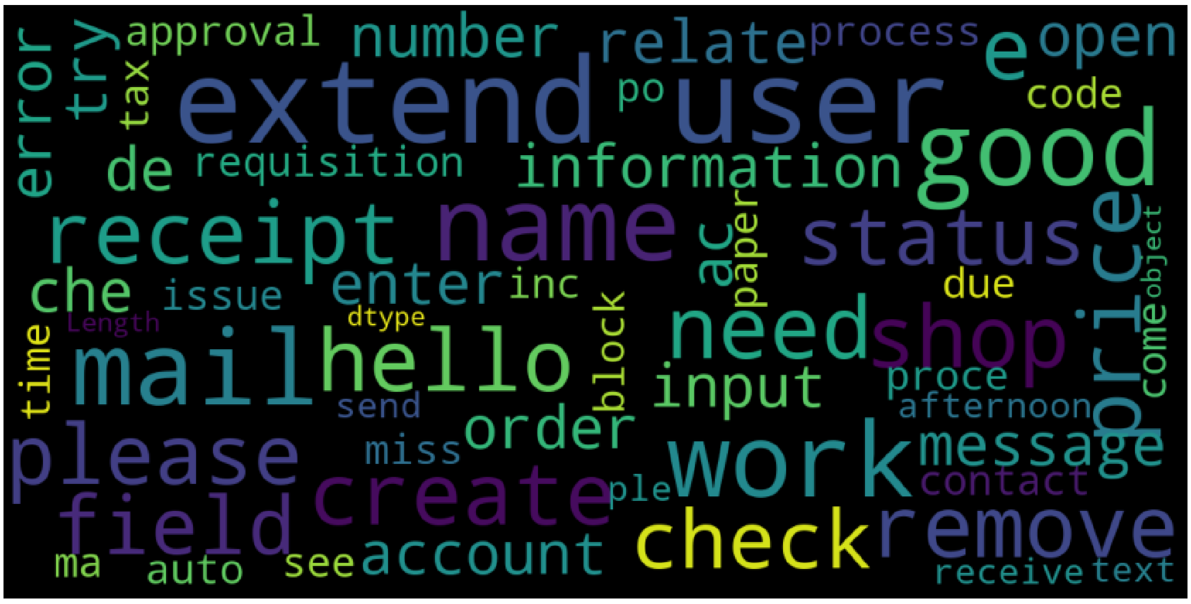
Grp_25


step
please
remove
null
record
wir
press
access
subject
Name
issue
p
W
approve
send
draw
c
ap
error
steel
recall
window
mal
hello
rectify
open
revision
data
Length
seit
receive
scrapt
die
min
tool
sir
work
engineering
object
e
problem
text
re
message
vendor
dtype
able
find
den
dear
see
seiner

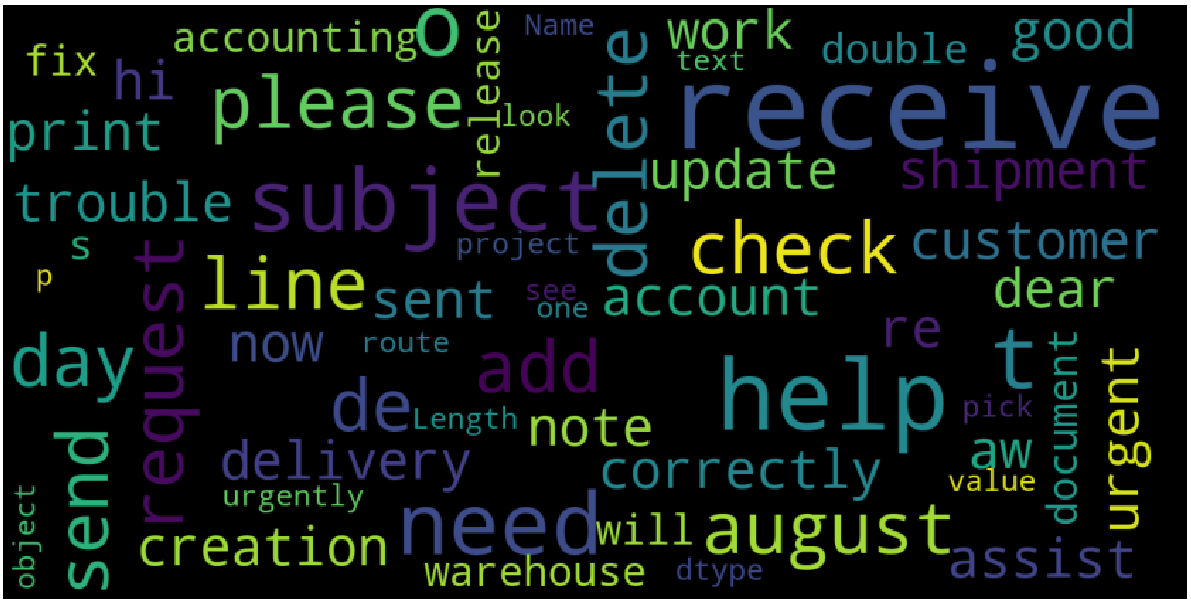
Grp_04


recurrent
network
access
compa
interface
controller
receive
phone
server
core
name
duplication
internal
try
connect
address
every
involve
two
netwo
networ
sh
outage
threshold
intermittent
interruption
connection
switch
lead
office
training
issue
dtype
object
requester
Length
temperature
put
room
text
shop
warning
thermal
flashing
stop
net
desk
machine
use
red
m
c

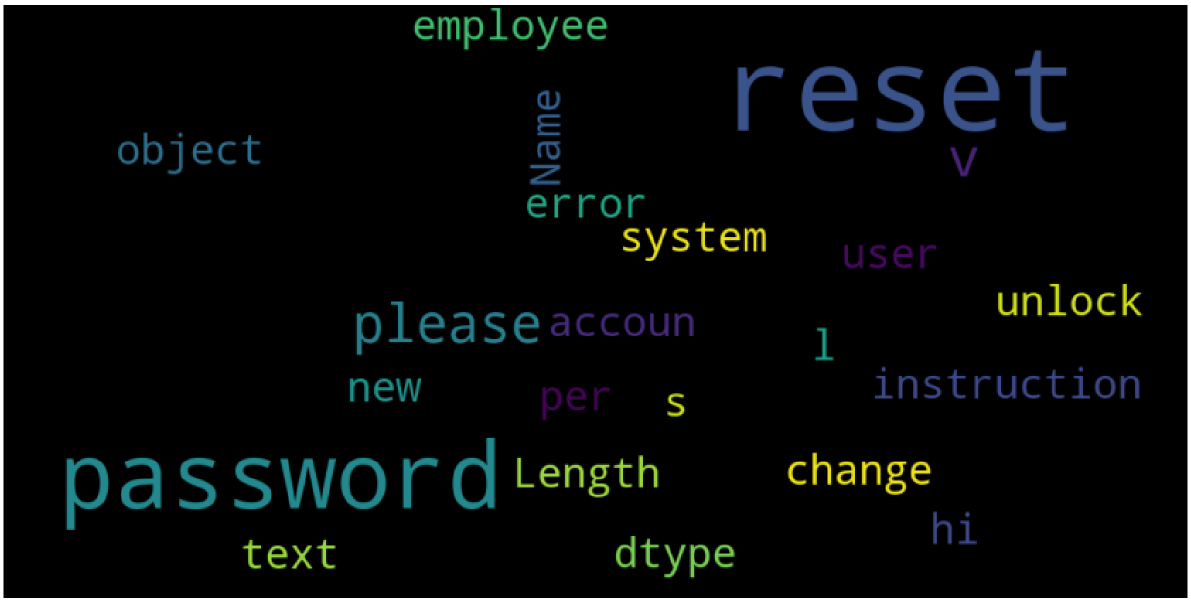
Grp_29


extend
user
good
name
mail
hello
receipt
please
create
field
check
remove
work
need
shop
price
status
information
number
relate
process
open
e
code
approval
try
tax
error
de
requisition
po
ac
paper
che
issue
enter
inc
due
object
come
block
input
proce
afternoon
message
contact
order
send
miss
ple
time
Length
dtype
ma
auto
see
account
receive
text

Grp_18


receive
subject
help
need
delete
please
check
request
send
august
line
day
customer
update
shipment
account
trouble
print
correctly
add
delivery
creation
de
note
urgent
document
warehouse
accounting
work
good
double
fix
hi
release
look
Name
text
sent
now
route
project
see
one
dear
re
t
pick
aw
value
assist
dtype
will
urgently
Length
object
s
p
o

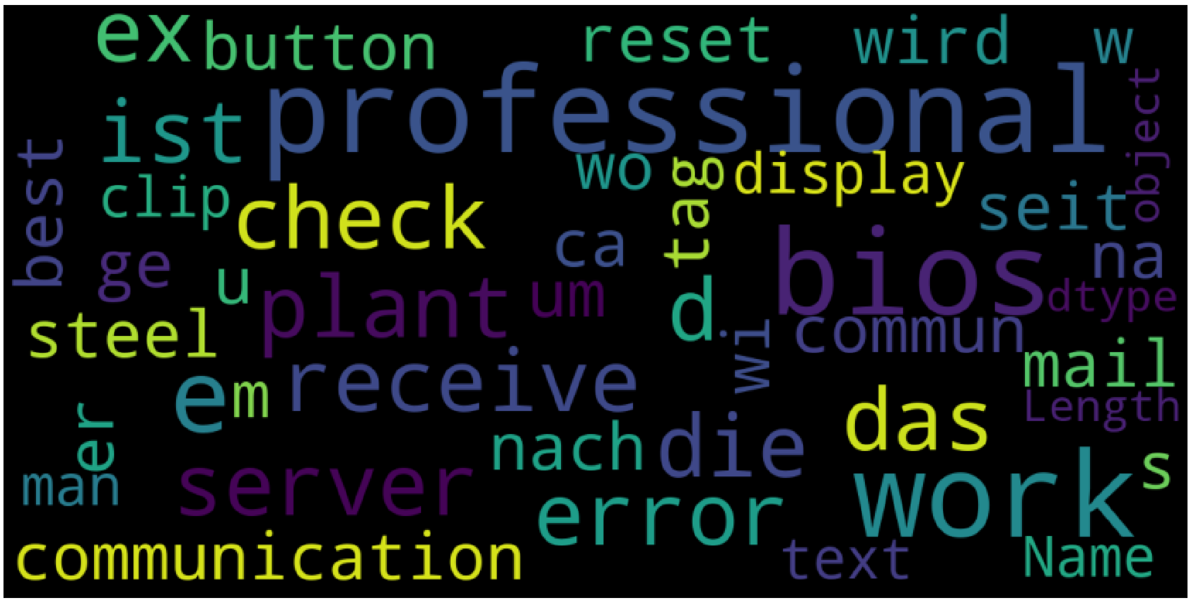
Grp_17


employee
reset
object
Name
v
error
system
user
unlock
please
accoun
l
instruction
new
per
s
password
Length
change
hi
text
dtype

Grp_16

access
retire
site
receive
need
holiday
hello
information
finance
try
manager
location
always
plant
hub
section
regional
employee
user
use
now
request
group
work
new
team
accidentally
informatio
bill
believe
send
ownership
text
message
attached
add
rebuild
language
holi
incorrect
list
modern
dtype
Name
object
someone
Length
si
see
look
image
acc
please
attach
edit
know
ta
review
ethic
recent
let
m

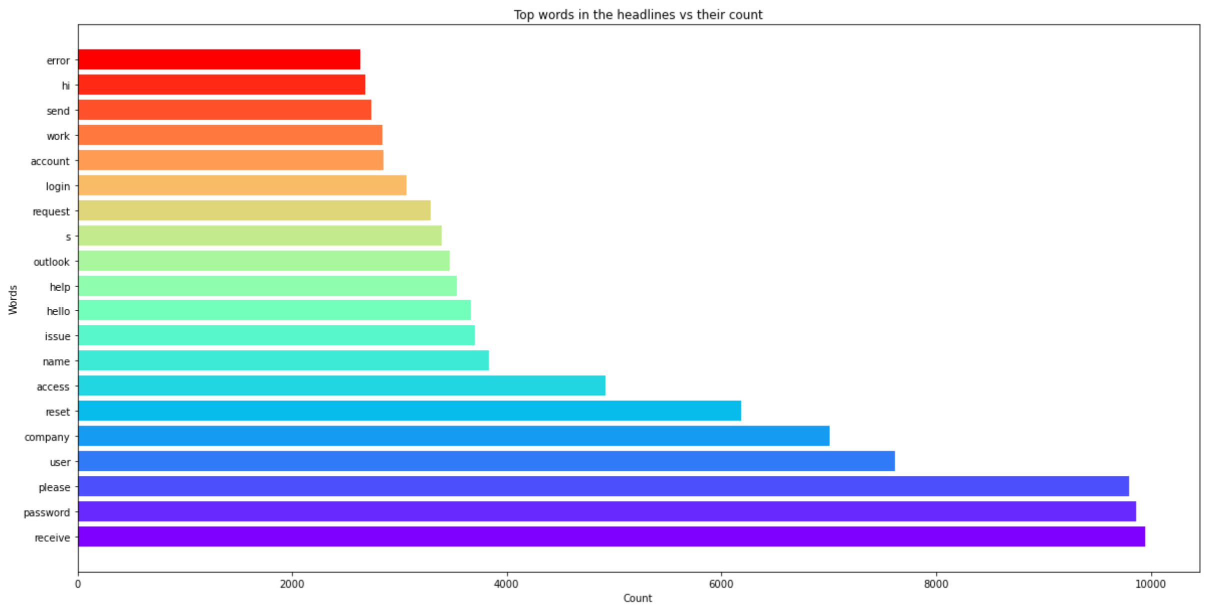
Grp_33


professional
bios
work
check
receive
plant
server
error
die
das
communication
ist
ex
button
reset
wird
w
display
wo
tag
seit
object
na
dtype
clip
best
ge
u
steel
um
ca
d
wi
commun
mail
Length
em
nach
er
man
text
Name
s

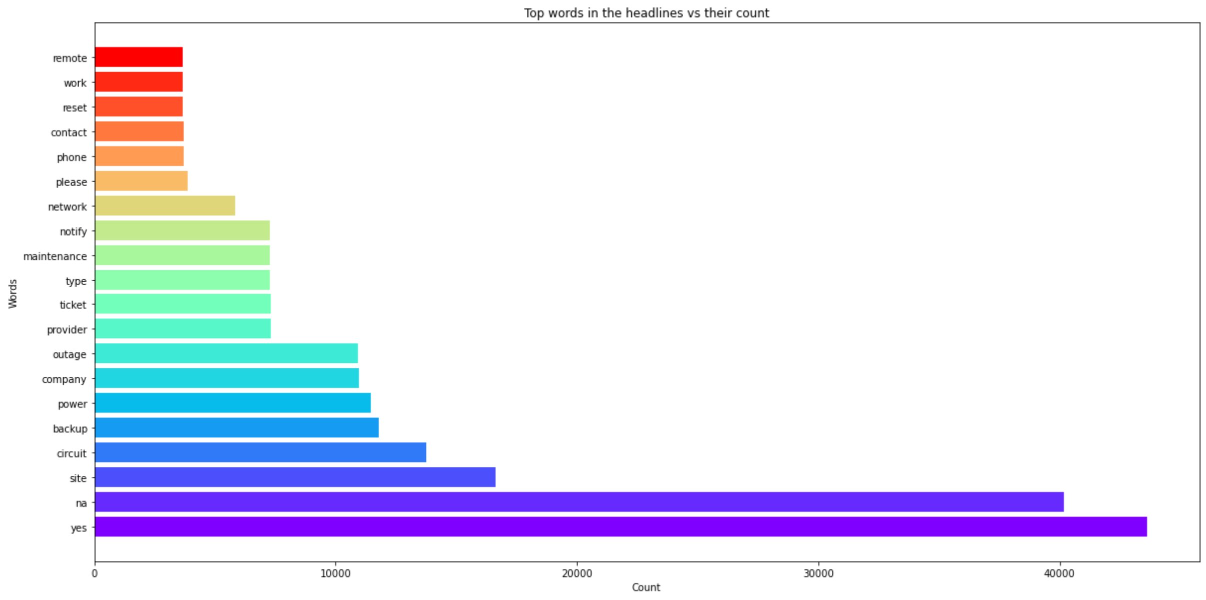
Charts

Frequent words used in Top 20 groups [Sharing charts for a few of them]

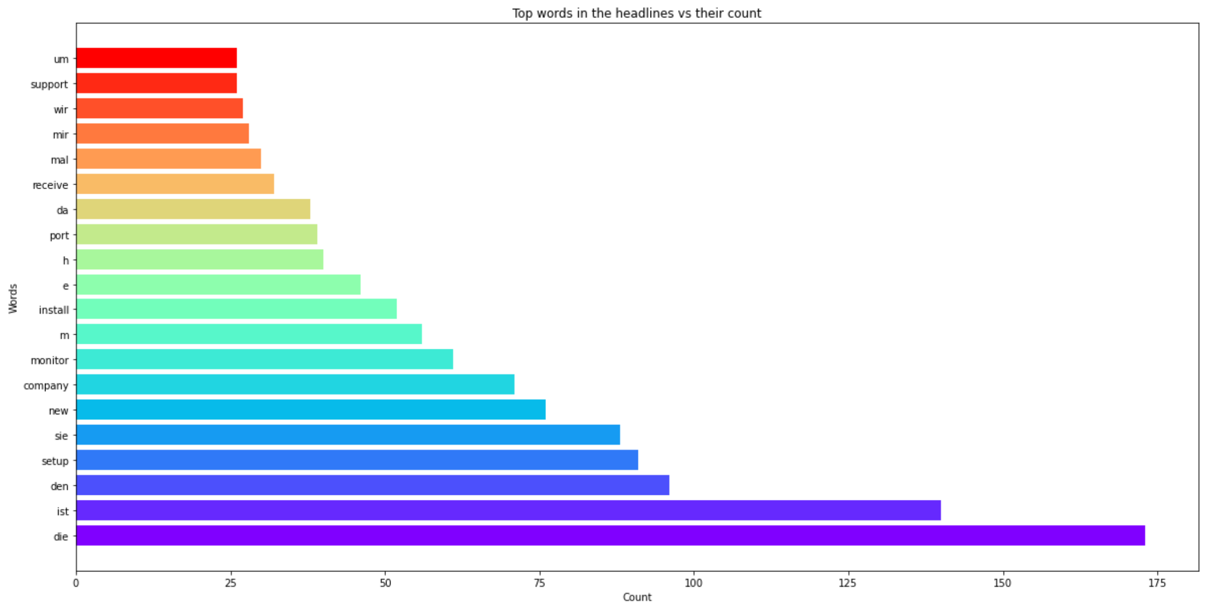
Grp_0



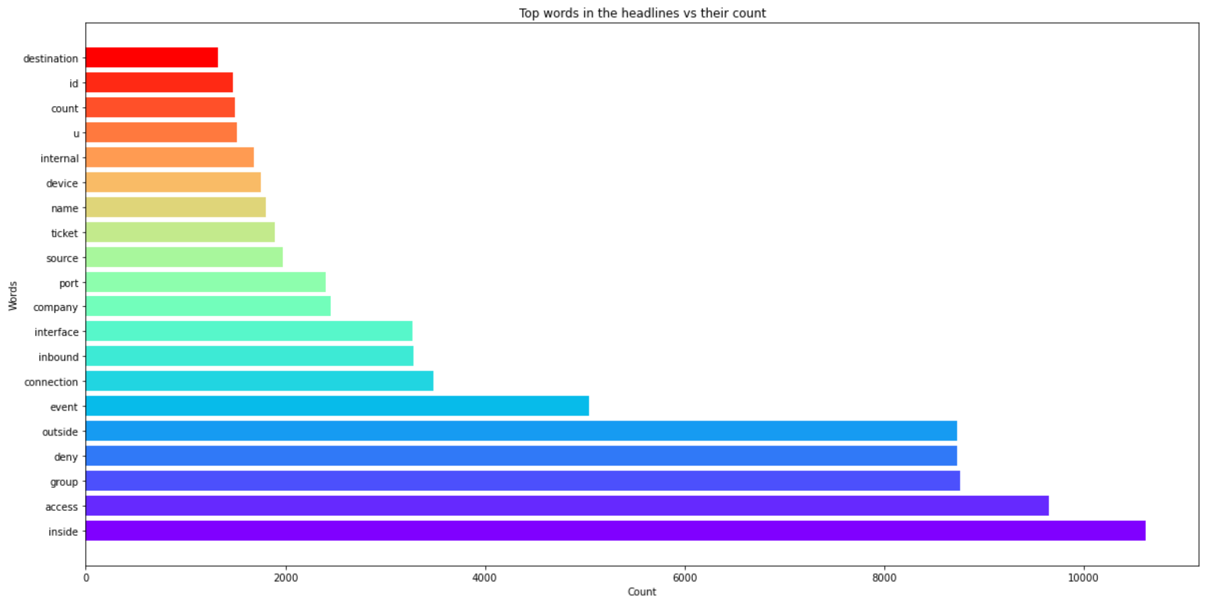
Grp_8



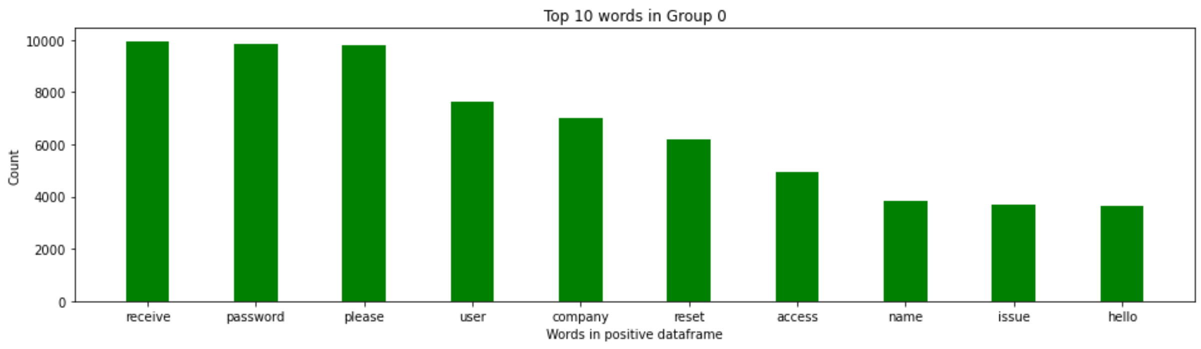
Grp_24


Top words in the headlines vs their count
Words
um
support
wir
mir
mal
receive
da
port
h
e
install
m
monitor
company
new
sie
setup
den
ist
die
0
25
50
75
100
125
150
175
Count

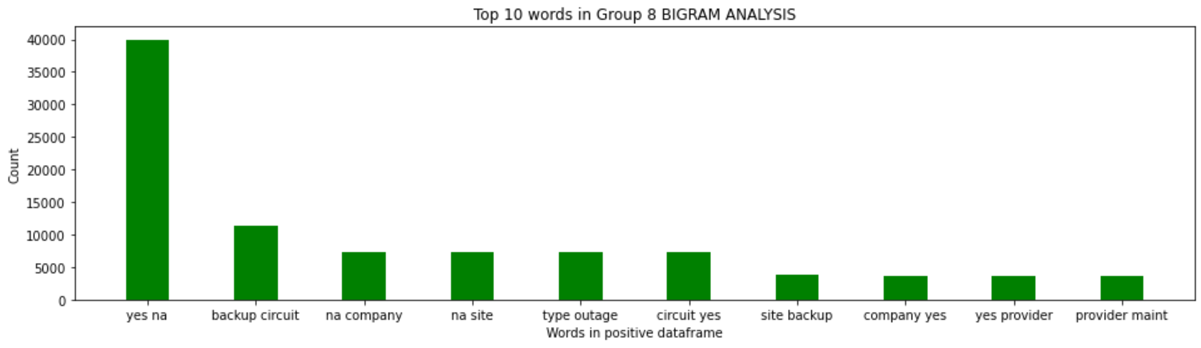
Grp_12

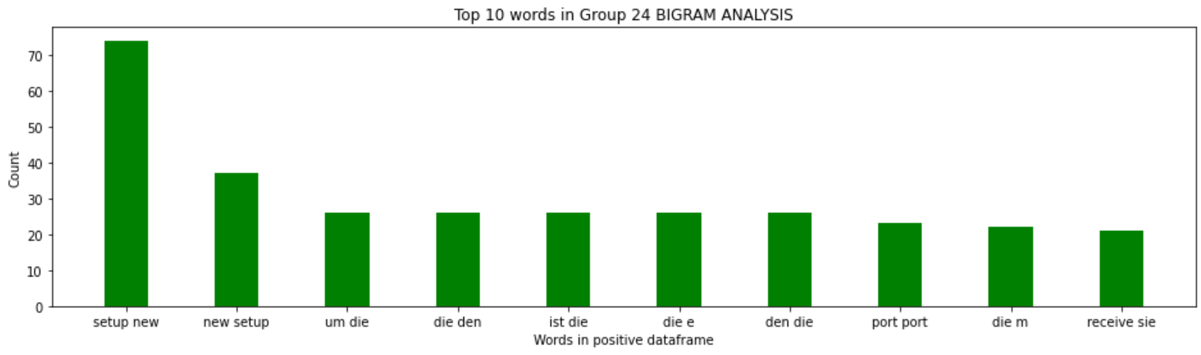

Top words in the headlines vs their count
Words
destination
id
count
u
internal
device
name
ticket
source
port
company
interface
inbound
connection
event
outside
deny
group
access
inside
0
2000
4000
6000
8000
10000
Count

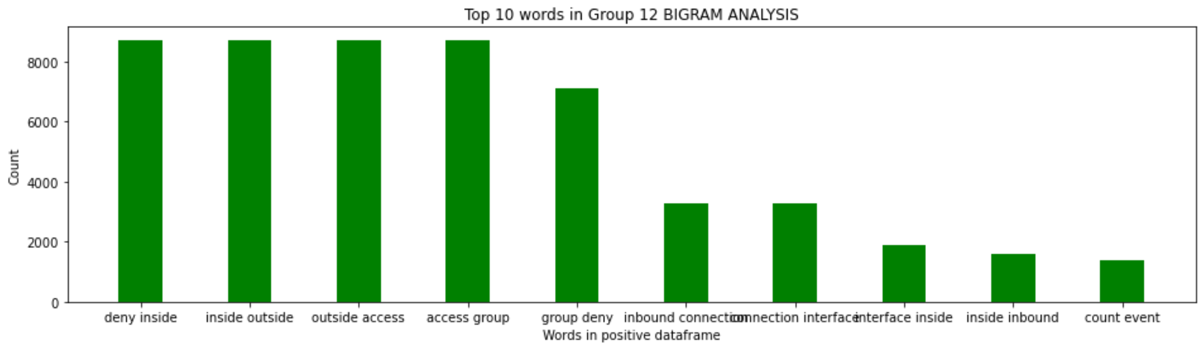
Unigram



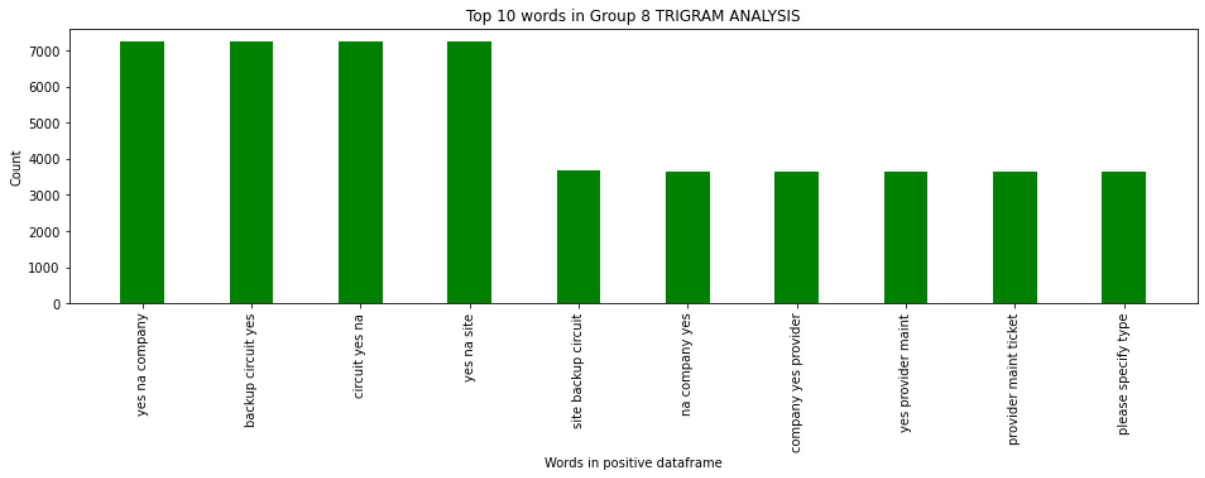
Bigram

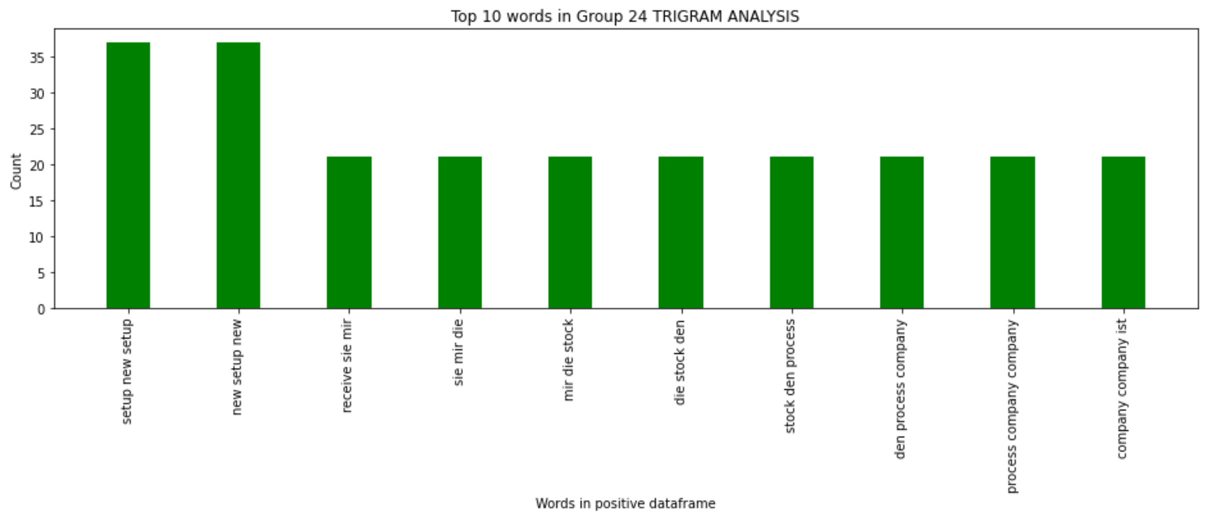


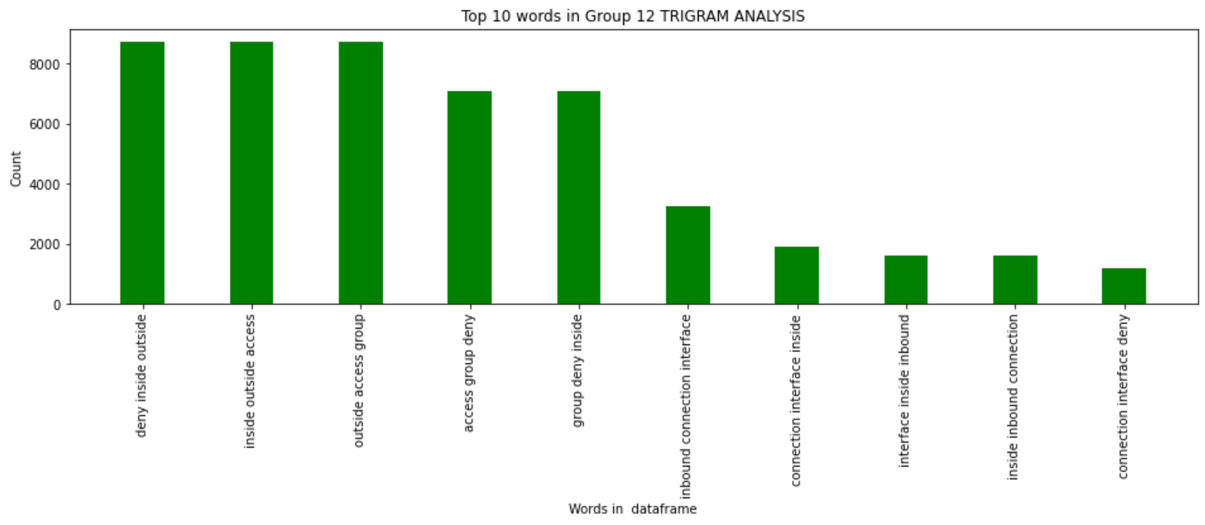




Trigram


Top 10 words in Group 8 TRIGRAM ANALYSIS
Count
0
1000
2000
3000
4000
5000
6000
7000
yes na company
backup circuit yes
circuit yes na
yes na site
site backup circuit
na company yes
company yes provider
yes provider maint
provider maint ticket
please specify type
Words in positive dataframe

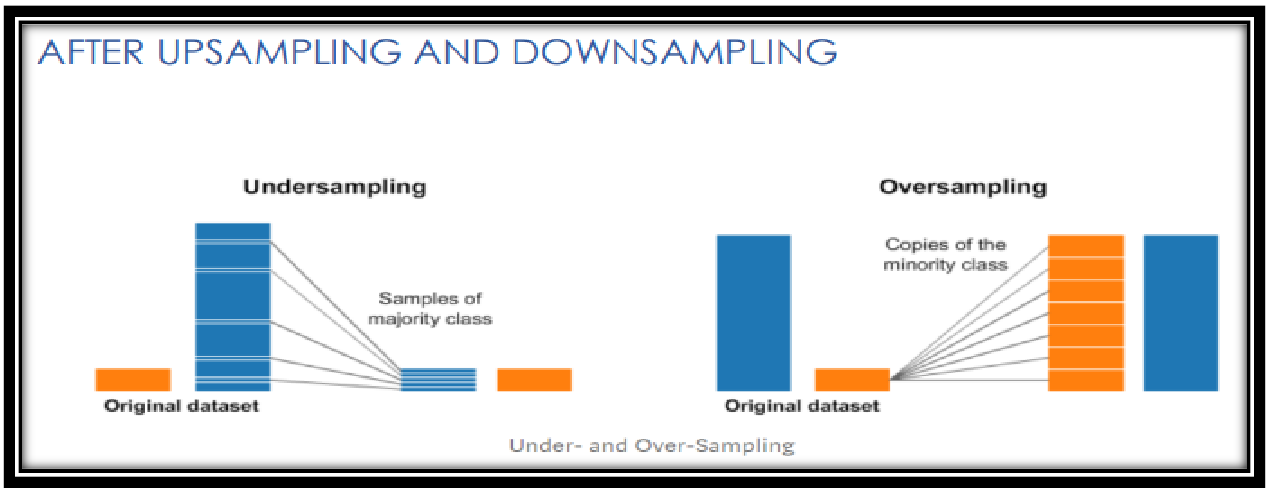

Top 10 words in Group 24 TRIGRAM ANALYSIS
Count
0
5
10
15
20
25
30
35
setup new setup
new setup new
receive sie mir
sie mir die
mir die stock
die stock den
stock den process
den process company
process company company
company company ist
Words in positive dataframe


Top 10 words in Group 12 TRIGRAM ANALYSIS
Count
0
2000
4000
6000
8000
deny inside outside
inside outside access
outside access group
access group deny
group deny inside
inbound connection interface
connection interface inside
interface inside inbound
inside inbound connection
connection interface deny
Words in dataframe

Decide Model and Model building

As the target class is completely skewed, various models have been tried with the sampled datasets to compare each performance.

1. Unigram, Bi-gram and Tri-gram models
2. Glove Embedding
3. Count Vectorization
4. TF - IDF Vectorization
5. Text Augmentation
6. Naive Bayes
7. Logistic Regression
8. XGBoost
9. GRU model
10. RNN model
11. Random Forest
12. Bidirectional LSTM


AFTER UPSAMPLING AND DOWNSAMPLING
Undersampling
Samples of majority class
Original dataset
Oversampling
Copies of the minority class
Original dataset
Under- and Over-Sampling

Word Embedding

As all our Machine Learning and Deep learning algorithms are incapable of processing strings or plain text in their raw form, word embedding has been used to convert the texts into numbers. There may be different numerical representations of the same text. It tries to map a word using a dictionary to a vector.

Tokenizing Text -> Representing each word by a number Mapping of original word to number is preserved in word index property of tokenizer Tokenized applies basic processing like changing it to lower case, explicitly setting that as False Lets keep all news to 300, add padding to news with less than 300 words and truncating long ones.

We have experimented with Glove embedding in our models.

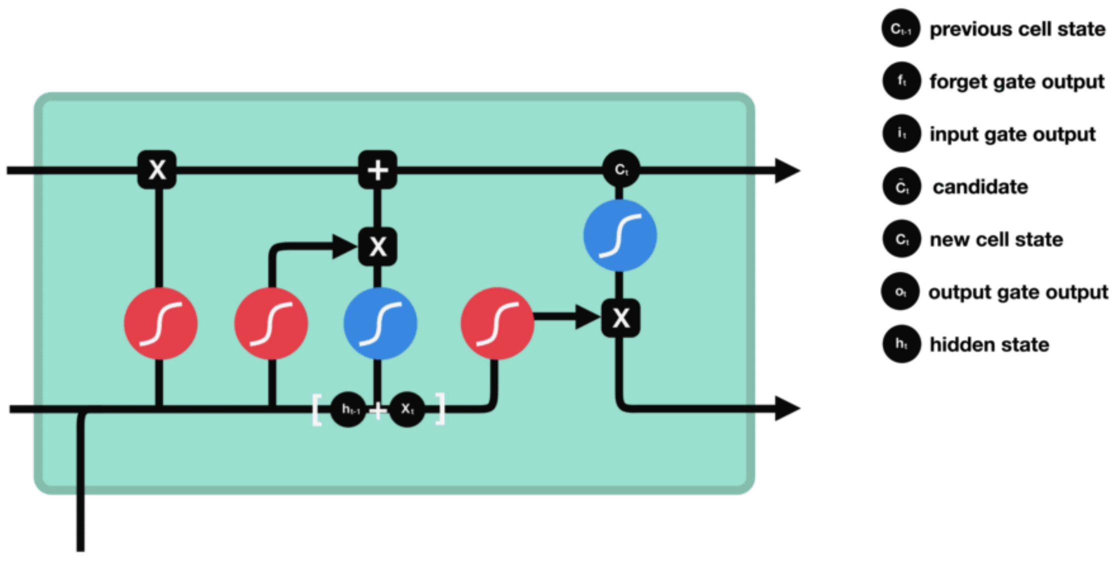
GloVe (Global Vectors) Embedding:
GloVe is an unsupervised learning algorithm for obtaining vector representations for words. Training is performed on aggregated global word-word co-occurrence statistics from a corpus, and the resulting representations showcase interesting linear substructures of the word vector space.

Text Augmentation
Data augmentation techniques are used to generate additional, synthetic data using the data you have. We used Synonym Replacement technique to up sample text Randomly choose n words from the sentence that are not stop words. Replace each of these words with one of its synonyms chosen at random.

Count Vectorization
Count Vectorization involves counting the number of occurrences each word appears in a document (i.e., distinct text such as an article, book, even a paragraph!). Python's Sci-kit learn library has a tool called Count Vectorizer to accomplish this.

Tf idf Vectorization
Tf idf Vectorizer - Transforms text to feature vectors that can be used as input to estimator.vocabulary_Is a dictionary that converts each token (word) to feature index in the matrix, each unique token gets a feature index

Bi-directional LSTM Model
Bidirectional LSTMs are an extension of traditional LSTMs that can improve model performance on classification problems. In problems where all timesteps of the input sequence are available, Bidirectional LSTMs train two instead of one LSTMs on the input sequence. The first on the input sequence as-is and the second on a reversed copy of the input sequence. This can provide additional context to the network and result in faster and even fuller learning on the problem.

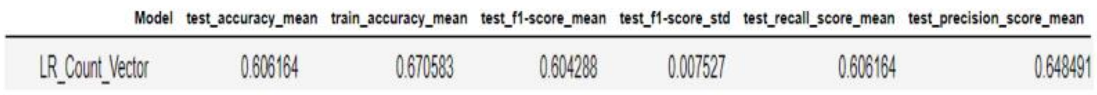


Machine Learning Models

Logistic Regression

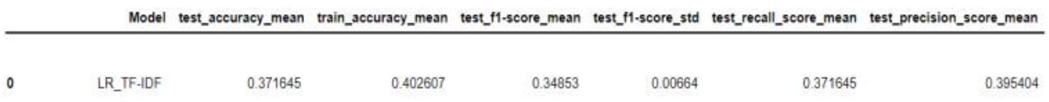
classification algorithm to predict multi class labels

- Applied logistic regression after applying count vectorization on data



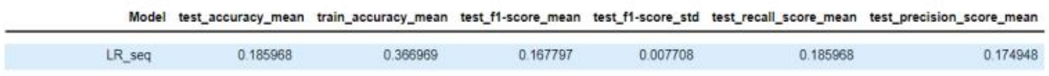
	Model	test_accuracy_mean	train_accuracy_mean	test_f1-score_mean	test_f1-score_std	test_recall_score_mean	test_precision_score_mean
	LR_Count_Vector	0.606164	0.670583	0.604288	0.007527	0.606164	0.648491

- Applied logistic regression after applying TF-IDF Vectorization



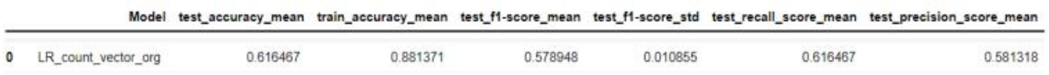
	Model	test_accuracy_mean	train_accuracy_mean	test_f1-score_mean	test_f1-score_std	test_recall_score_mean	test_precision_score_mean
0	LR_TF-IDF	0.371645	0.402607	0.34853	0.00664	0.371645	0.395404

- Applied logistic regression after applying TF-IDF sequence padding



	Model	test_accuracy_mean	train_accuracy_mean	test_f1-score_mean	test_f1-score_std	test_recall_score_mean	test_precision_score_mean
	LR_seq	0.185968	0.366969	0.167797	0.007708	0.185968	0.174948

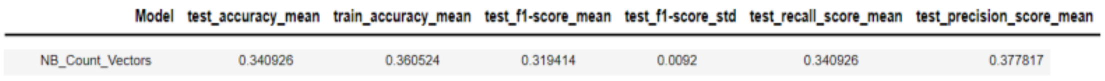
- Logistic Regression on Original data before Up sampling



	Model	test_accuracy_mean	train_accuracy_mean	test_f1-score_mean	test_f1-score_std	test_recall_score_mean	test_precision_score_mean
0	LR_count_vector_org	0.616467	0.881371	0.578948	0.010855	0.616467	0.581318

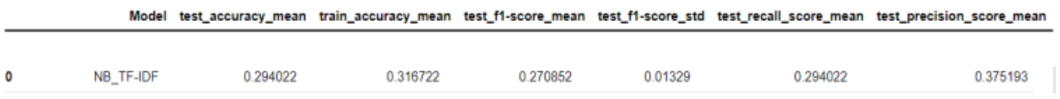
Naïve Bayes

- Applied Naïve Bayes after applying count vectorization on data



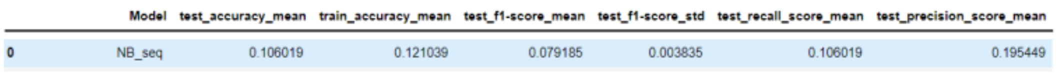
Model	test_accuracy_mean	train_accuracy_mean	test_f1-score_mean	test_f1-score_std	test_recall_score_mean	test_precision_score_mean
NB_Count_Vectors	0.340926	0.360524	0.319414	0.0092	0.340926	0.377817

- Applied Naïve Bayes after applying TF-IDF Vectorization



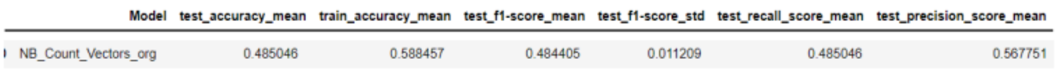
	Model	test_accuracy_mean	train_accuracy_mean	test_f1-score_mean	test_f1-score_std	test_recall_score_mean	test_precision_score_mean
0	NB_TF-IDF	0.294022	0.316722	0.270852	0.01329	0.294022	0.375193

- Applied Naïve Bayes after applying TF-IDF sequence padding



	Model	test_accuracy_mean	train_accuracy_mean	test_f1-score_mean	test_f1-score_std	test_recall_score_mean	test_precision_score_mean
0	NB_seq	0.106019	0.121039	0.079185	0.003835	0.106019	0.195449

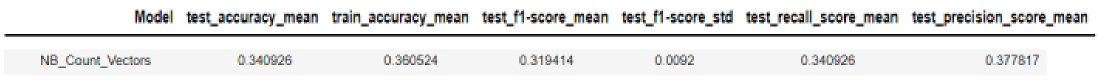
- Naïve Bayes on Orginal data before Up sampling



Model	test_accuracy_mean	train_accuracy_mean	test_f1-score_mean	test_f1-score_std	test_recall_score_mean	test_precision_score_mean
NB_Count_Vectors_org	0.485046	0.588457	0.484405	0.011209	0.485046	0.567751

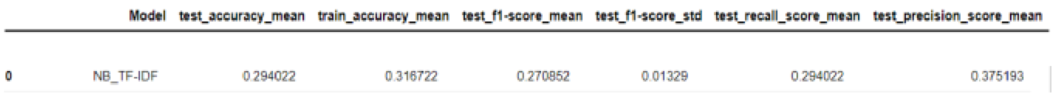
XGBoost

- Applied Xgboost after applying count vectorization on data



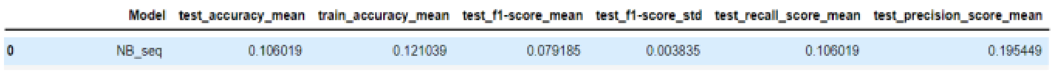
Model	test_accuracy_mean	train_accuracy_mean	test_f1-score_mean	test_f1-score_std	test_recall_score_mean	test_precision_score_mean
NB_Count_Vectors	0.340926	0.360524	0.319414	0.0092	0.340926	0.377817

- Applied Xgboost after applying TF-IDF Vectorization



	Model	test_accuracy_mean	train_accuracy_mean	test_f1-score_mean	test_f1-score_std	test_recall_score_mean	test_precision_score_mean
0	NB_TF-IDF	0.294022	0.316722	0.270852	0.01329	0.294022	0.375193

- Applied Xgboost after applying TF-IDF sequence padding

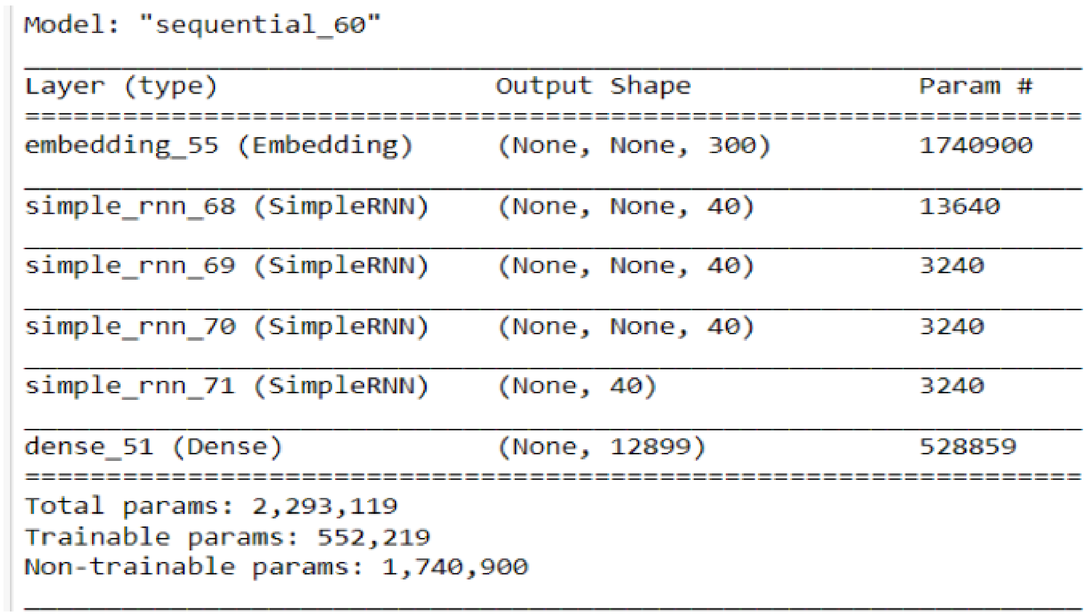


	Model	test_accuracy_mean	train_accuracy_mean	test_f1-score_mean	test_f1-score_std	test_recall_score_mean	test_precision_score_mean
0	NB_seq	0.106019	0.121039	0.079185	0.003835	0.106019	0.195449

Deep Learning Models

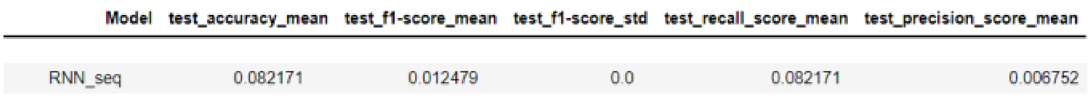
RNN Architecture

RNNs perform the same task for every element of a sequence, with the output being dependent on the previous computations. Another way to think about RNNs is that they have a “memory” which captures information about what has been calculated so far.



```
Model: "sequential_60"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_55 (Embedding)     (None, None, 300)         1740900
_________________________________________________________________
simple_rnn_68 (SimpleRNN)    (None, None, 40)          13640
_________________________________________________________________
simple_rnn_69 (SimpleRNN)    (None, None, 40)          3240
_________________________________________________________________
simple_rnn_70 (SimpleRNN)    (None, None, 40)          3240
_________________________________________________________________
simple_rnn_71 (SimpleRNN)    (None, 40)                3240
_________________________________________________________________
dense_51 (Dense)             (None, 12899)             528859
=================================================================
Total params: 2,293,119
Trainable params: 552,219
Non-trainable params: 1,740,900
_________________________________________________________________
```

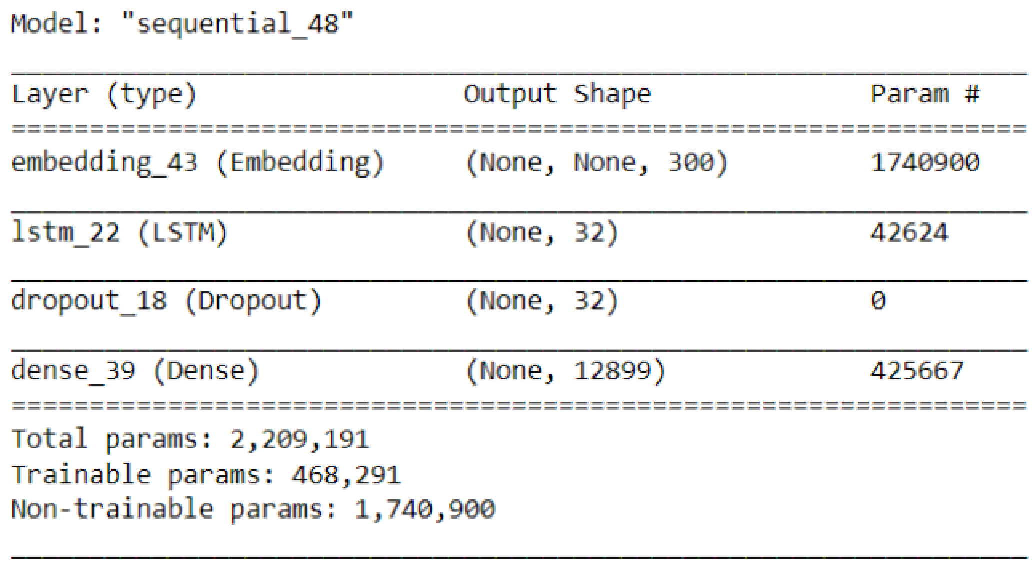
RNN Performance



Model	test_accuracy_mean	test_f1-score_mean	test_f1-score_std	test_recall_score_mean	test_precision_score_mean
RNN_seq	0.082171	0.012479	0.0	0.082171	0.006752

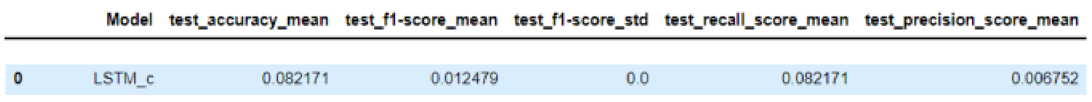
LSTM Architecture

It is special kind of recurrent neural network that is capable of learning long term dependencies in data. This is achieved because the recurring module of the model has a combination of four layers interacting with each other.



```
Model: "sequential_48"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_43 (Embedding)     (None, None, 300)         1740900
_________________________________________________________________
lstm_22 (LSTM)               (None, 32)                42624
_________________________________________________________________
dropout_18 (Dropout)         (None, 32)                0
_________________________________________________________________
dense_39 (Dense)             (None, 12899)             425667
=================================================================
Total params: 2,209,191
Trainable params: 468,291
Non-trainable params: 1,740,900
_________________________________________________________________
```

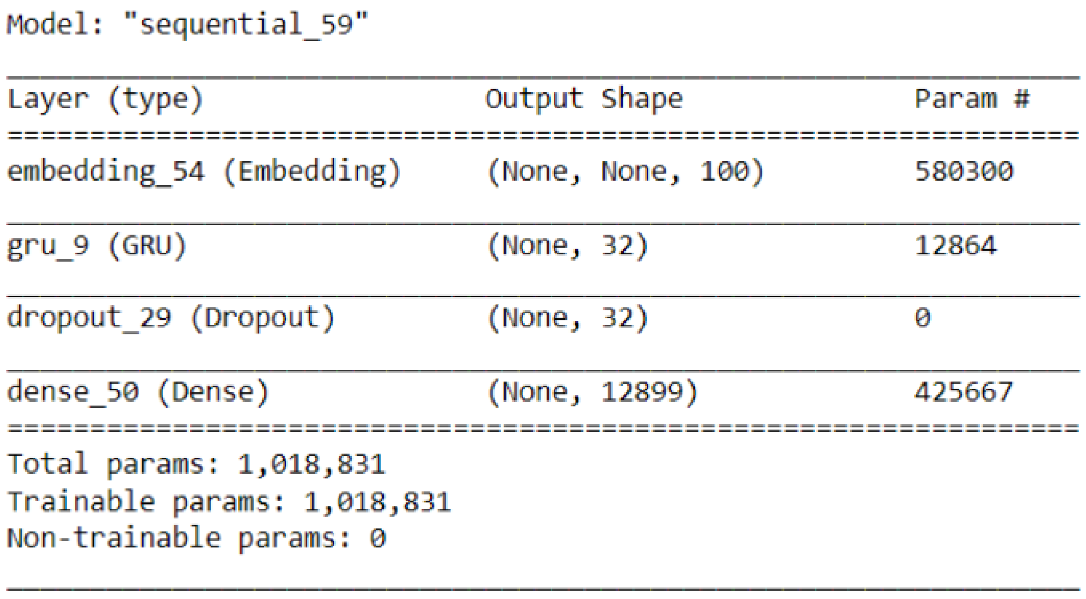
LSTM Performance



	Model	test_accuracy_mean	test_f1-score_mean	test_f1-score_std	test_recall_score_mean	test_precision_score_mean
0	LSTM_c	0.082171	0.012479	0.0	0.082171	0.006752

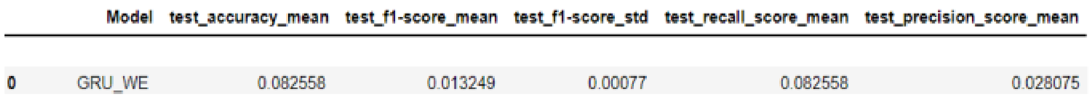
GRU Architecture

To solve the vanishing gradient problem of a standard RNN, GRU uses, so-called, **update gate and reset gate**. Basically, these are two vectors which decide what information should be passed to the output. The special thing about them is that they can be trained to keep information from long ago, without washing it through time or remove information which is irrelevant to the prediction.



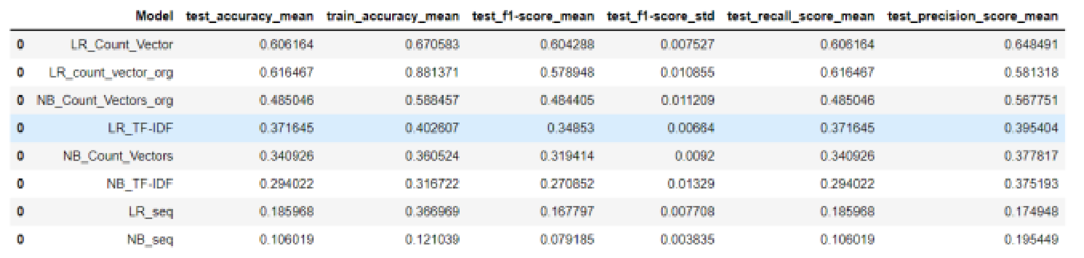
```
Model: "sequential_59"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_54 (Embedding)     (None, None, 100)         580300
_________________________________________________________________
gru_9 (GRU)                  (None, 32)                12864
_________________________________________________________________
dropout_29 (Dropout)         (None, 32)                0
_________________________________________________________________
dense_50 (Dense)             (None, 12899)             425667
=================================================================
Total params: 1,018,831
Trainable params: 1,018,831
Non-trainable params: 0
_________________________________________________________________
```

GRU Performance



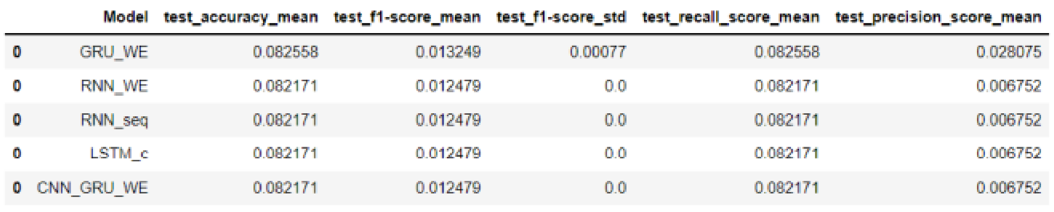
	Model	test_accuracy_mean	test_f1-score_mean	test_f1-score_std	test_recall_score_mean	test_precision_score_mean
0	GRU_WE	0.082558	0.013249	0.00077	0.082558	0.028075

ML Model Performance



	Model	test_accuracy_mean	train_accuracy_mean	test_f1-score_mean	test_f1-score_std	test_recall_score_mean	test_precision_score_mean
0	LR_Count_Vector	0.606164	0.670583	0.604288	0.007527	0.606164	0.648491
0	LR_count_vector_org	0.616467	0.881371	0.578948	0.010855	0.616467	0.581318
0	NB_Count_Vectors_org	0.485046	0.588457	0.484405	0.011209	0.485046	0.567751
0	LR_TF-IDF	0.371645	0.402607	0.34853	0.00684	0.371645	0.395404
0	NB_Count_Vectors	0.340926	0.360524	0.319414	0.0092	0.340926	0.377817
0	NB_TF-IDF	0.294022	0.316722	0.270852	0.01329	0.294022	0.375193
0	LR_seq	0.185968	0.366969	0.167797	0.007708	0.185968	0.174948
0	NB_seq	0.106019	0.121039	0.079185	0.003835	0.106019	0.195449

DL Model Performance



	Model	test_accuracy_mean	test_f1-score_mean	test_f1-score_std	test_recall_score_mean	test_precision_score_mean
0	GRU_WE	0.082558	0.013249	0.00077	0.082558	0.028075
0	RNN_WE	0.082171	0.012479	0.0	0.082171	0.006752
0	RNN_seq	0.082171	0.012479	0.0	0.082171	0.006752
0	LSTM_c	0.082171	0.012479	0.0	0.082171	0.006752
0	CNN_GRU_WE	0.082171	0.012479	0.0	0.082171	0.006752

Comparison to benchmark

From the given problem description, we could see that the current system is able to assign 75% of the tickets correctly. So, our objective here is to build an AI-based classifier model to assign the tickets to right functional groups by analysing the given description and achieve a higher degree of accuracy. From the prediction results we see that the GRU model based on is able to achieve an accuracy of 82.25% which is above than the benchmark.

Implications

Although the GRU model can classify the IT tickets with 82.25% accuracy, to achieve better accuracy in the real world it would be good if additional data is collected for each group which may help reduce the data imbalance and skewness of the data.

Limitations

As part of Data pre-processing, we have considered only the top 20 groups based on the count frequency. However, when applying this model in real world scenario, there could be quite a deviation in the model performance if the data is balanced across all assignment groups.

Closing Reflections

We found the data was present in multiple languages and in various formats such as emails, chat, etc bringing in a lot of variability in the data to be analyzed.
The Business can improve the process of raising tickets via a common unified IT Ticket Service Portal which reduces the above-mentioned variability.
By doing this, the model can perform better which can help businesses to identify the problem area for relevant clusters of topics.