Automatic Ticket Classification - EDA

In [1]:
```
%load_ext autoreload
%autoreload 2
```

In [2]:
```
# imports

import os
import math
import xlrd
import hjson
import random
import warnings
from time import time
from pathlib import Path
import pandas as pd, numpy as np
from pprint import pprint
import matplotlib.pyplot as plt
import seaborn as sns
from tqdm import tqdm
from collections import defaultdict, Counter
from wordcloud import WordCloud, STOPWORDS
from ftfy import fix_text
from sklearn.preprocessing import LabelEncoder
import tensorflow
from utils.utils import load_hjson
config = load_hjson(Path('./config/config.hjson'))

tqdm.pandas()
warnings.filterwarnings('ignore')
warnings.simplefilter(action='ignore', category=FutureWarning)
%matplotlib inline
```

```
C:\Users\surya\anaconda3\envs\full\lib\site-packages\tqdm\std.py:697: FutureWarning: The Panel
```

```
class is removed from pandas. Accessing it from the top-level namespace will also be removed in the next version
  from pandas import Panel
```

In [3]:
```
# reproducibility
seed = 7
random.seed(seed)
tensorflow.random.set_seed(seed)
```

- # Import & Analyse the data.

In [4]:
```
# check encoding
with open('./data/input_data.xlsx', 'r') as fp:
    print(fp)
```

```
<_io.TextIOWrapper name='./data/input_data.xlsx' mode='r' encoding='cp1252'>
```

In [5]:
```
wb = xlrd.open_workbook('./data/input_data.xlsx', encoding_override='cp1252')
dataset = pd.read_excel(wb)
dataset.sample(7)
```

Out[5]:

	Short description	Description	Caller	Assignment group
6770	bitte einen arbeitszeitplan erstellen fÃ¼r die...	bitte einen arbeitszeitplan erstellen fÃ¼r die...	ltxzfcgm sxvigclz	GRP_52
4976	please check the ale in detail, we have ongoin...	please have a close look on the ale interface ...	pesylifc wnyierbu	GRP_20
3273	account unlock - erp SID_34	unlocked account using password_management_too...	wgpimkle kijhcwur	GRP_0
167	user needs training to use engineering tool to...	user needs training to use engineering tool to...	rkzqjbwc juizkwpl	GRP_0
1838	unable to access erp	unable to access erp \n\nuser contacted for an...	edqylkio ykomciav	GRP_2
1340	job Job_563 failed in job_scheduler at: 10/15/...	received from: monitoring_tool@company.com\r\n...	bpctwhsn kzqsbmtp	GRP_8
7970	abended job in job_scheduler: Job_1320	received from: monitoring_tool@company.com\r\n...	ZkBogxib QsEJzdZO	GRP_9

In [6]:
```
dataset.shape  # very small dataset with only 8500 rows
```

Out[6]:
```
(8500, 4)
```

In [7]:
```
dataset.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
```

In [8]:
```
# rename column names for convenience
dataset = dataset.rename(columns={
    "Short description": "short_description",
    "Description": "description",
    "Caller": "caller",
```

```
    "Assignment group": "group"
    })
```

In [10]:
```
np.unique(dataset.group), len(np.unique(dataset.group))  # 74 different functional groups
```

Out[10]:
```
(array(['GRP_0', 'GRP_1', 'GRP_10', 'GRP_11', 'GRP_12', 'GRP_13', 'GRP_14',
        'GRP_15', 'GRP_16', 'GRP_17', 'GRP_18', 'GRP_19', 'GRP_2',
        'GRP_20', 'GRP_21', 'GRP_22', 'GRP_23', 'GRP_24', 'GRP_25',
        'GRP_26', 'GRP_27', 'GRP_28', 'GRP_29', 'GRP_3', 'GRP_30',
        'GRP_31', 'GRP_32', 'GRP_33', 'GRP_34', 'GRP_35', 'GRP_36',
        'GRP_37', 'GRP_38', 'GRP_39', 'GRP_4', 'GRP_40', 'GRP_41',
        'GRP_42', 'GRP_43', 'GRP_44', 'GRP_45', 'GRP_46', 'GRP_47',
        'GRP_48', 'GRP_49', 'GRP_5', 'GRP_50', 'GRP_51', 'GRP_52',
        'GRP_53', 'GRP_54', 'GRP_55', 'GRP_56', 'GRP_57', 'GRP_58',
        'GRP_59', 'GRP_6', 'GRP_60', 'GRP_61', 'GRP_62', 'GRP_63',
        'GRP_64', 'GRP_65', 'GRP_66', 'GRP_67', 'GRP_68', 'GRP_69',
        'GRP_7', 'GRP_70', 'GRP_71', 'GRP_72', 'GRP_73', 'GRP_8', 'GRP_9'],
       dtype=object),
 74)
```

- ## Check for Incomplete Information

In [11]:
```
dataset.isna().sum()  # Few missing values
```

Out[11]:
```
short_description    8
description          1
caller               0
group                0
dtype: int64
```

In [12]:
```
dataset[dataset.isna().any(axis=1)]  # check rows with missing values
```

Out[12]:

	short_description	description	caller	group
2604	NaN	\r\n\r\nreceived from: ohdrnswl.rezuibdt@gmail...	ohdrnswl rezuibdt	GRP_34
3383	NaN	\r\n-connected to the user system using teamvi...	qftpazns fxpnytmk	GRP_0
3906	NaN	-user unable tologin to vpn.\r\n-connected to...	awpcmsey ctdiuqwe	GRP_0
3910	NaN	-user unable tologin to vpn.\r\n-connected to...	rhwsmefo tvphyura	GRP_0
3915	NaN	-user unable tologin to vpn.\r\n-connected to...	hxripljo efzounig	GRP_0
3921	NaN	-user unable tologin to vpn.\r\n-connected to...	cziadygo veiosxby	GRP_0
3924	NaN	name:wvqgbdhm fwchqjor\nlanguage:\nbrowser:mic...	wvqgbdhm fwchqjor	GRP_0
4341	NaN	\r\n\r\nreceived from: eqmuniov.ehxkcbgj@gmail...	eqmuniov ehxkcbgj	GRP_0
4395	i am locked out of skype	NaN	viyglzfo ajtfzpkb	GRP_0

In [13]:
```
dataset.loc[dataset['description'].isna()]
```

Out[13]:

	short_description	description	caller	group
4395	i am locked out of skype	NaN	viyglzfo ajtfzpkb	GRP_0

In [14]:
```
dataset.iloc[4395]
```

Out[14]:
```
short_description    i am locked out of skype
description                               NaN
caller                        viyglzfo ajtfzpkb
group                                   GRP_0
Name: 4395, dtype: object
```

In [15]:
```
# imputing the short description by value in description and vice-versa
# However, If both columns were missing, we would have to drop the row
dataset.loc[dataset['description'].isna(), 'description']  = dataset.loc[dataset['description']
```

In [16]:
```
dataset.loc[dataset['short_description'].isna()]
```

Out[16]:

	short_description	description	caller	group
2604	NaN	\r\n\r\nreceived from: ohdrnswl.rezuibdt@gmail...	ohdrnswl rezuibdt	GRP_34
3383	NaN	\r\n-connected to the user system using teamvi...	qftpazns fxpnytmk	GRP_0
3906	NaN	-user unable tologin to vpn.\r\n-connected to...	awpcmsey ctdiuqwe	GRP_0
3910	NaN	-user unable tologin to vpn.\r\n-connected to...	rhwsmefo tvphyura	GRP_0
3915	NaN	-user unable tologin to vpn.\r\n-connected to...	hxripljo efzounig	GRP_0
3921	NaN	-user unable tologin to vpn.\r\n-connected to...	cziadygo veiosxby	GRP_0
3924	NaN	name:wvqgbdhm fwchqjor\nlanguage:\nbrowser:mic...	wvqgbdhm fwchqjor	GRP_0
4341	NaN	\r\n\r\nreceived from: eqmuniov.ehxkcbgj@gmail...	eqmuniov ehxkcbgj	GRP_0

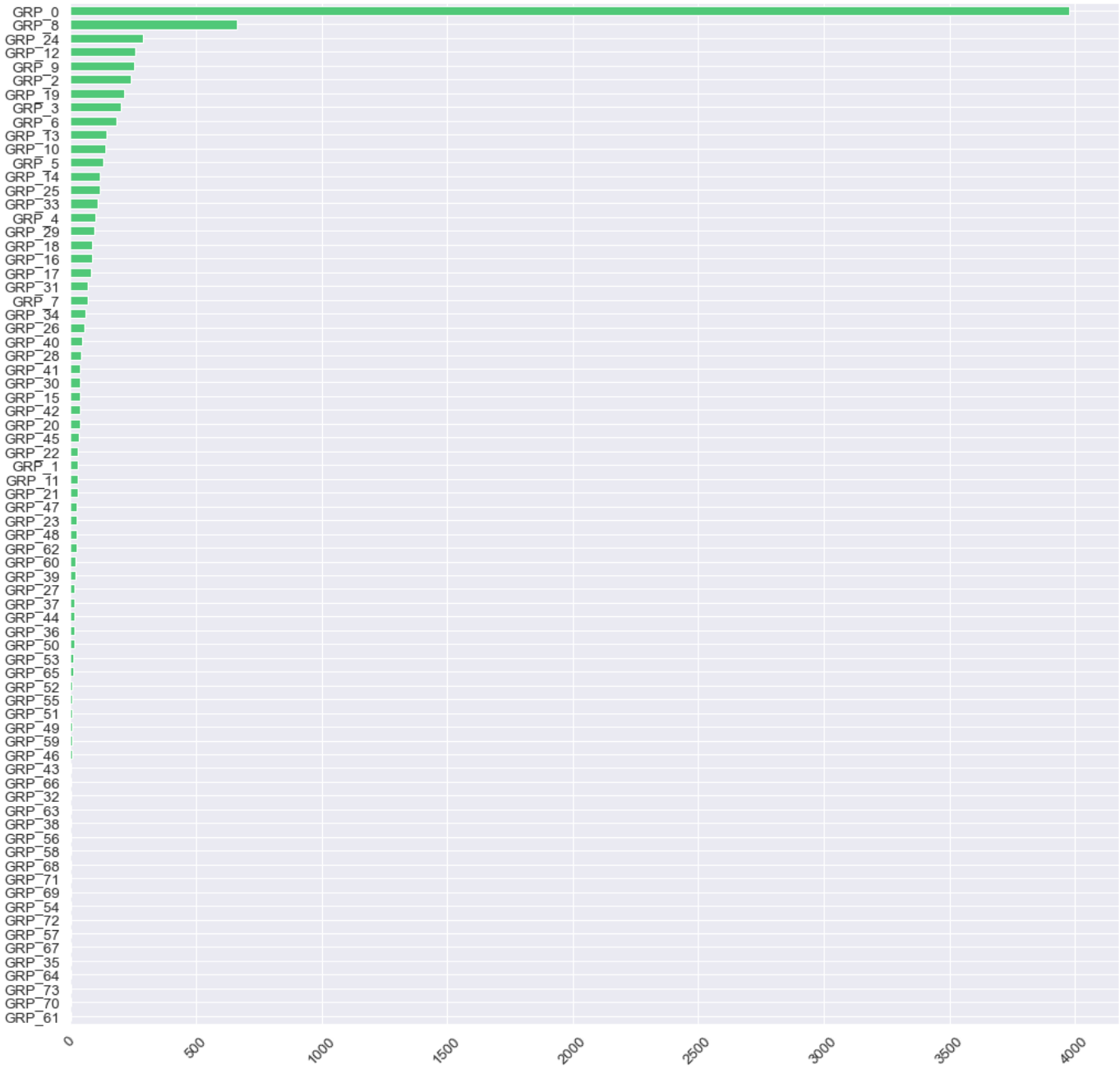
In [17]:
```
dataset.loc[dataset['short_description'].isna(), 'short_description']  = dataset.loc[dataset['s
```

In [18]:
```
dataset.isna().sum()  # all missing values imputed
```

Out[18]:
```
short_description    0
description          0
caller               0
group                0
dtype: int64
```

- ## Target Class Distribution

In [19]:
```
sns.set(font_scale=1.2)  # scale up font size
dataset.group.value_counts().sort_values(ascending=True).plot(kind='barh', width=0.65, figsize=
plt.xticks(rotation=45)
plt.show()
```



In [20]:
```
dataset.group.value_counts().sort_values(ascending=False).tail(30)  # few classes with only sir
```

Out[20]:

GRP_44	15
GRP_36	15
GRP_50	14
GRP_53	11
GRP_65	11
GRP_52	9
GRP_55	8
GRP_51	8
GRP_49	6
GRP_59	6
GRP_46	6
GRP_43	5
GRP_66	4
GRP_32	4
GRP_63	3
GRP_38	3
GRP_56	3
GRP_58	3
GRP_68	3
GRP_71	2
GRP_69	2
GRP_54	2
GRP_72	2
GRP_57	2
GRP_67	1
GRP_35	1
GRP_64	1

```
GRP_73     1
GRP_70     1
GRP_61     1
Name: group, dtype: int64
```

In [21]:
```
dataset[dataset.group == 'GRP_70']  # small groups have to merged into a separate "Others" cate
dataset[dataset.group == 'GRP_70'].description.tolist()[0]
```

Out[21]:
```
'an e-mail from  it training has email hints and tips #1.  under "create signature" it has a li
nk "company formatheywting standard" that i am forbidden to see.\n615'
```

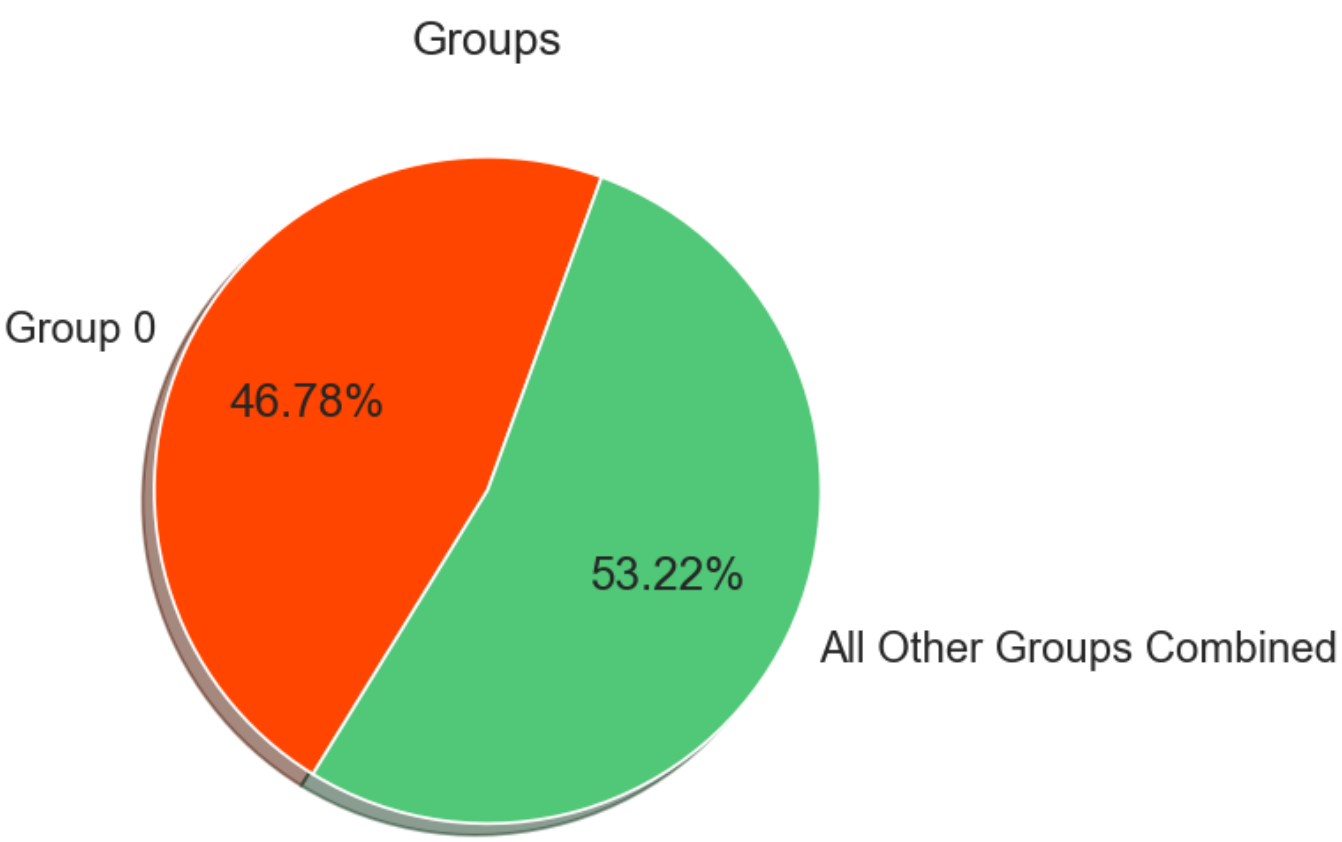
- The target class distribution is heavily imbalanced as most calls are assinged to Group 0 and exluding this as well, we find an imabalanced dataset for the rest of the groups.

In [22]:
```
temp = dataset.copy(deep=True)
temp.loc[temp["group"] != 'GRP_0', 'group'] = 'Other'
temp.loc[temp["group"] == 'GRP_0', "group"] = 'Group 0'
```

In [23]:
```
temp.group.value_counts()
```

Out[23]:
```
Other      4524
Group 0    3976
Name: group, dtype: int64
```

In [24]:
```
sns.set(font_scale=1.25)  # scale up font size

plt.figure(figsize=(5, 5), dpi=125)
group_0 = len(temp[temp['group'] == 'Group 0'])
others = len(temp[temp['group'] == 'Other'])

plt.pie(x=[group_0, others],
        explode=(0, 0),
        labels=['Group 0', 'All Other Groups Combined'],
        autopct='%1.2f%%',
        shadow=True,
        startangle=70,
        colors=['#FF4500', '#50C878'])

fig = plt.gcf()
fig.set_size_inches(5, 5)
plt.title('Groups')
plt.show()
```



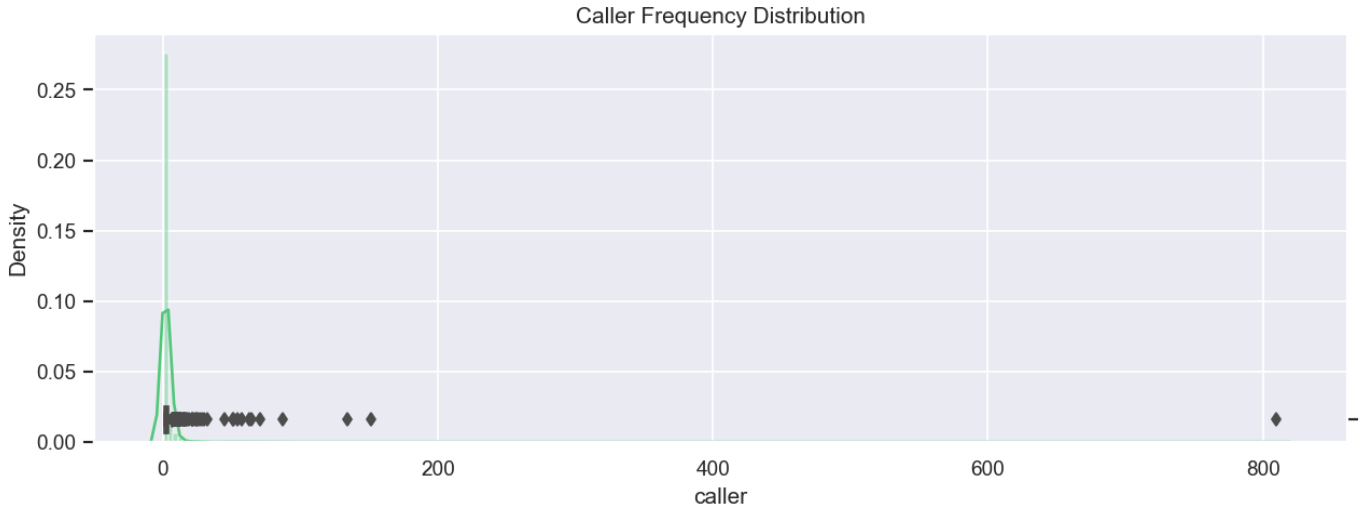
- ## Choosing a Metric to benchmark model performance

- This is a multi-class classification problem, where the machine learning model will try to predict if each row is one of the 74 possibilities.
- The majority class is GRP_0, which occurs in 46.78% of the observations.
- The most common metrics for a multi-class classification problem are AUC, F1-score and accuracy.
- Accuracy is not suitable for an imbalanced classification problem. (Note that a model that always predicts GRP_0, will get an accuracy of 46.78%)
- We would choose F1-score if the majority class is more important than the smaller classes.
- We would choose AUC if we also care about the smaller classes.

As we want to be able to classify the tickets into all functional groups and functional groups are given equal importance, we choose AUC as the final metric to score model performance.

- ## Outlier Analysis

In [25]:
```
# plotting caller frequency counts
sns.set()
plt.figure(figsize=(12, 4), dpi=125)
ax = sns.distplot(dataset.caller.value_counts(), bins=250, kde=True, color='#50C878')
ax_ = ax.twinx()
sns.boxplot(dataset.caller.value_counts(), color='#FF4500')
ax_.set(ylim=(-.7, 12))
plt.title('Caller Frequency Distribution')
plt.show()
```



In [26]:
```
dataset[dataset.caller == 'bpctwhsn kzqsbmtp'].group.value_counts()  # most frequent caller
```

Out[26]:
```
GRP_8     362
GRP_9     153
GRP_5      96
GRP_6      89
GRP_10     60
GRP_60     16
GRP_12      8
GRP_45      7
GRP_1       6
GRP_13      4
GRP_18      3
GRP_47      2
GRP_57      1
GRP_44      1
GRP_29      1
GRP_14      1
Name: group, dtype: int64
```

In [27]:
```
dataset[dataset.caller == 'bpctwhsn kzqsbmtp']  # job failure alerts
```

Out[27]:

	short_description	description	caller	group
47	job Job_1424 failed in job_scheduler at: 10/31...	received from: monitoring_tool@company.com\r\n...	bpctwhsn kzqsbmtp	GRP_6
50	job mm_zscr0099_dly_merktc3 failed in job_sche...	received from: monitoring_tool@company.com\r\n...	bpctwhsn kzqsbmtp	GRP_8
59	job mm_zscr0099_dly_merktc2 failed in job_sche...	received from: monitoring_tool@company.com\r\n...	bpctwhsn kzqsbmtp	GRP_8
60	job Job_3181 failed in job_scheduler at: 10/31...	received from: monitoring_tool@company.com\r\n...	bpctwhsn kzqsbmtp	GRP_8
67	job Job_1338 failed in job_scheduler at: 10/31...	received from: monitoring_tool@company.com\r\n...	bpctwhsn kzqsbmtp	GRP_9
...	...	...	...	...
7053	job Job_1387 failed in job_scheduler at: 08/18...	received from: monitoring_tool@company.com\r\n...	bpctwhsn kzqsbmtp	GRP_9
7059	job Job_2063b failed in job_scheduler at: 08/1...	received from: monitoring_tool@company.com\r\n...	bpctwhsn kzqsbmtp	GRP_6
7074	job HostName_1019fail failed in job_scheduler ...	received from: monitoring_tool@company.com\r\n...	bpctwhsn kzqsbmtp	GRP_8
7076	job HostName_1019fail failed in job_scheduler ...	received from: monitoring_tool@company.com\r\n...	bpctwhsn kzqsbmtp	GRP_8
7077	job HostName_1019fail failed in job_scheduler ...	\r\n\r\nreceived from: monitoring_tool@company...	bpctwhsn kzqsbmtp	GRP_60

810 rows × 4 columns

In [28]:
```
def get_length(row):
    try:
        row['char_length'] = len(row.description)
        row['word_length'] = len(row.description.split())
        row['short_char_length'] = len(row.short_description)
        row['short_word_length'] = len(row.short_description.split())
    except Exception:  # assign 0 length to missing rows if any
        row['char_length'] = 0
        row['word_length'] = 0
        row['short_char_length'] = 0
        row['short_word_length'] = 0
    return row

dataset = dataset.progress_apply(get_length, axis=1)
```

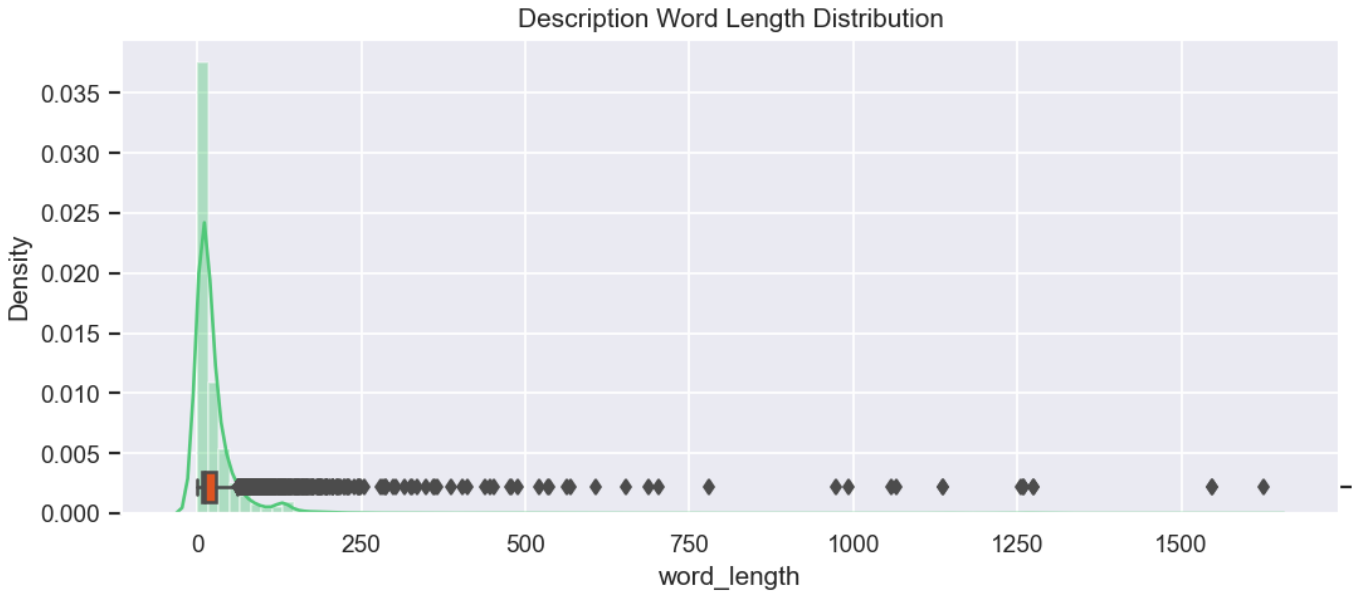
```
100%|██████████████████████████████████████████████████████████████████████| 8500/8500
[00:31<00:00, 266.93it/s]
```

In [29]:
```
dataset.sample(7)
```

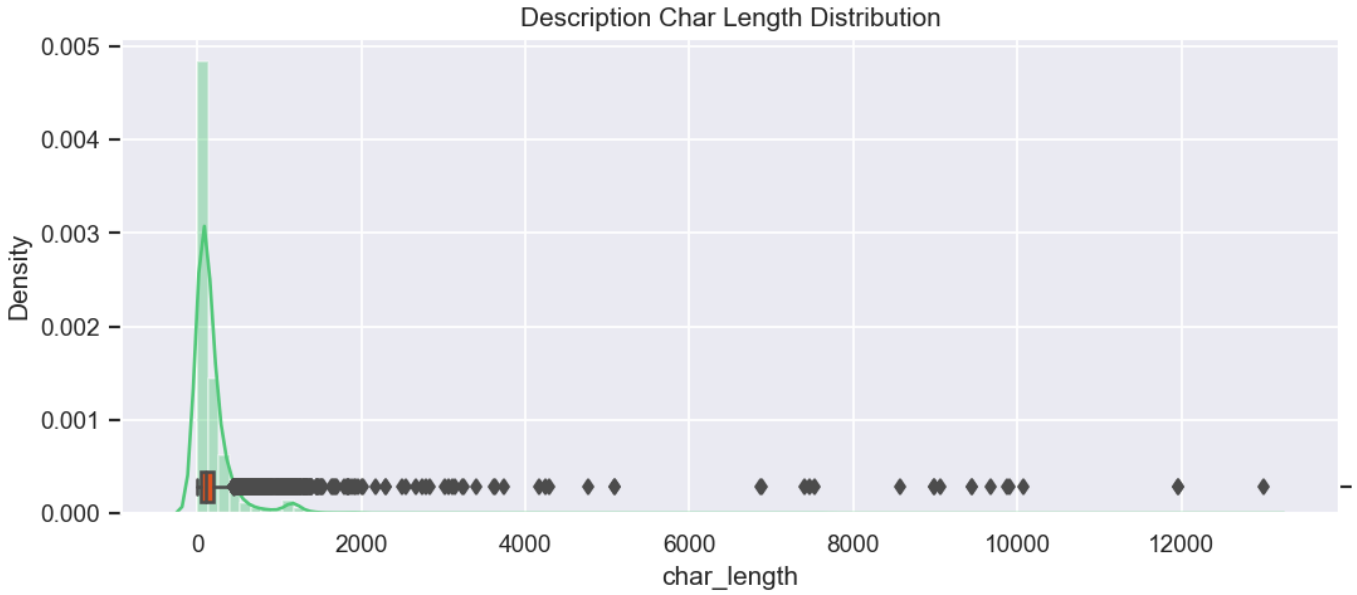
Out[29]:

	short_description	description	caller	group	char_length	word_length	short_char_le
5038	unable to connect to dv28, dv06, and dv40	unable to connect to dv28, dv06, and dv40	iwtvrhnz rxiumhfk	GRP_0	41	8	
2439	do i have to worry about this?	\r\n\r\nreceived from: hbmwlprq.ilfvyodx@gmail...	hbmwlprq ilfvyodx	GRP_0	123	9	
4304	interface: gigabitethernet1/0/47 Â· mtb gf wir...	interface: gigabitethernet1/0/47 Â· mtb gf wir...	rkupnshb gsmzfojw	GRP_4	122	11	
1825	outlook not working correctly	outlook not working correctly & freezing	hvstqfwc buvsrnze	GRP_0	41	6	
31	reset users	hi\n\nplease reset users password\n\nclient id...	qcehailo wqynckxg	GRP_0	64	11	
1697	user needs help to create delivery.	\r\n\r\nreceived from: gjtyswkb.dpvaymxr@gmail...	gjtyswkb dpvaymxr	GRP_0	92	12	
3150	alrthyu s lowe. my ee# is 6045304.it needs to ...	from: scthyott lortwe \nsent: monday, septembe...	sbvlxuwm yanbikrx	GRP_9	886	128	

In [30]:
```
sns.set()
plt.figure(figsize=(10, 4), dpi=125)
ax = sns.distplot(dataset.word_length, bins=100, kde=True, color='#50C878')
ax_ = ax.twinx()
sns.boxplot(dataset.word_length, color='#FF4500')
ax_.set(ylim=(-.7, 12))
plt.title('Description Word Length Distribution')
plt.show()
```



In [31]:
```
sns.set()
plt.figure(figsize=(10, 4), dpi=125)
ax = sns.distplot(dataset.char_length, bins=100, kde=True, color='#50C878')
ax_ = ax.twinx()
sns.boxplot(dataset.char_length, color='#FF4500')
ax_.set(ylim=(-.7, 12))
plt.title('Description Char Length Distribution')
plt.show()
```



In [32]:
```
dataset[dataset.word_length == 0]  # empty description => imputing with the corresponding short
```

Out[32]:

	short_description	description	caller	group	char_length	word_length	short_char_length	short_word_
6371	authorization add/delete members	\r\n\r\n	hpmwliog kqtnfvrl	GRP_0	5	0	33	
7397	browser issue :	\r\n	fgejnhux fnkymoht	GRP_0	2	0	16	

In [33]:
```
dataset.loc[dataset.word_length == 0, 'description'] = dataset.loc[dataset.word_length == 0]['s
dataset = dataset.progress_apply(get_length, axis=1)
```

```
100%|██████████| 8500/8500 [0
0:01<00:00, 4821.31it/s]
```

In [34]:
```
dataset[dataset.word_length == 0]  # cleaned
```

Out[34]:

short_description	description	caller	group	char_length	word_length	short_char_length	short_word_length

In [35]:
```
dataset[dataset.char_length < 4]  # description 'the' holds no information => imputed with shor
```

Out[35]:

	short_description	description	caller	group	char_length	word_length	short_char_length	short_wor
1049	reset passwords for soldfnbq uhnbsvqd using pa...	the	soldfnbq uhnbsvqd	GRP_17	3	1	84	
1054	reset passwords for fygrwuna gomcekzi using pa...	the	fygrwuna gomcekzi	GRP_17	3	1	84	
1144	reset passwords for wvdxnkhf jirecvta using pa...	the	wvdxnkhf jirecvta	GRP_17	3	1	84	
1184	reset passwords for pxvjczdt kizsjfpq using pa...	the	pxvjczdt kizsjfpq	GRP_17	3	1	84	
1292	reset passwords for cubdsrml znewqgop using pa...	the	cubdsrml znewqgop	GRP_17	3	1	84	
1476	reset passwords for bnoupaki cpeioxdz using pa...	the	bnoupaki cpeioxdz	GRP_17	3	1	84	
1558	reset passwords for usa feathers using passwor...	the	lmqysdec ljvbnpqw	GRP_17	3	1	79	
1693	reset passwords for eglavnhx uprodleq using pa...	the	eglavnhx uprodleq	GRP_17	3	1	84	
1834	reset passwords for hybiaxlk lawptzir using pa...	the	hybiaxlk lawptzir	GRP_17	3	1	84	
1850	reset passwords for fylrosuk kedgmiul using pa...	the	fylrosuk kedgmiul	GRP_17	3	1	84	
1851	reset passwords for fylrosuk kedgmiul using pa...	the	fylrosuk kedgmiul	GRP_17	3	1	84	
1860	s	s	gzjtweph mnslwfqv	GRP_0	1	1	1	
2151	reset passwords for gjisfonb odwfhmze using pa...	the	gjisfonb odwfhmze	GRP_17	3	1	84	
2532	reset passwords for qwsjptlo hnlasbed using pa...	the	goaxzsql qpjnbgsa	GRP_17	3	1	84	

	short_description	description	caller	group	char_length	word_length	short_char_length	short_word
2533	reset passwords for qwsjptlo hnlasbed using pa...	the	goaxzsql qpjnbgsa	GRP_17	3	1	84	
2553	reset passwords for bxeagsmt zrwdgsco using pa...	the	bxeagsmt zrwdgsco	GRP_17	3	1	84	
2554	reset passwords for bxeagsmt zrwdgsco using pa...	the	bxeagsmt zrwdgsco	GRP_17	3	1	84	
2572	reset passwords for prgewfly ndtfvple using pa...	the	prgewfly ndtfvple	GRP_17	3	1	84	
2602	reset passwords for wxdvjoct ckxwtoam using pa...	the	wxdvjoct ckxwtoam	GRP_17	3	1	84	
2605	reset passwords for ytzpxhql ntfxgpms using pa...	the	ytzpxhql ntfxgpms	GRP_17	3	1	84	
2749	reset passwords for fkuqjwit jgcsaqzi using pa...	the	fkuqjwit jgcsaqzi	GRP_17	3	1	84	
2788	reset passwords for hzmxwdrs tcbjyqps using pa...	the	hzmxwdrs tcbjyqps	GRP_17	3	1	84	
3000	reset passwords for knemilvx dvqtziya using pa...	the	jtwykasf elkhcjqn	GRP_17	3	1	84	
3432	dds	dss	onctqhsg cpahzsle	GRP_0	3	1	3	
3447	reset passwords for qoybxkfh dwcmxuea using pa...	the	qoybxkfh dwcmxuea	GRP_17	3	1	84	
3692	reset passwords for mvhcoqed konjdmwq using pa...	the	mvhcoqed konjdmwq	GRP_17	3	1	84	
3693	reset passwords for mvhcoqed konjdmwq using pa...	the	mvhcoqed konjdmwq	GRP_17	3	1	84	
4055	reset passwords for jerydwbn gdylnaue using pa...	the	jerydwbn gdylnaue	GRP_17	3	1	84	

	short_description	description	caller	group	char_length	word_length	short_char_length	short_word
4065	reset passwords for dmexgspl mruzqhac using pa...	the	dmexgspl mruzqhac	GRP_17	3	1	84	
4672	reset passwords for robhyertyj f duca using pa...	the	acteiqdu bferalus	GRP_17	3	1	84	
4978	reset passwords for davidthd robankm using pas...	the	zelunfcq yimdwjrp	GRP_17	3	1	83	
4984	reset passwords for cubdsrml znewqgop using pa...	the	cubdsrml znewqgop	GRP_17	3	1	84	
4991	reset passwords for davidthd robankm using pas...	the	zelunfcq yimdwjrp	GRP_17	3	1	83	
5074	reset passwords for mafgtnik -0 using password...	the	plzsntqj ujdyobsk	GRP_17	3	1	78	
5077	reset passwords for cÃ©sar abreu rghkiriuytes ...	the	btvmxdfc yfahetsc	GRP_17	3	1	92	
5182	reset passwords for yolktfas fyoxqgvh using pa...	the	yolktfas fyoxqgvh	GRP_17	3	1	84	
5228	reset passwords for ugawcoye jcfqgviy using pa...	the	ugawcoye jcfqgviy	GRP_17	3	1	84	
5305	reset passwords for qgilmtyc gmscovxa using pa...	the	qgilmtyc gmscovxa	GRP_17	3	1	84	
5317	reset passwords for bxeagsmt zrwdgsco using pa...	the	bxeagsmt zrwdgsco	GRP_17	3	1	84	
5482	reset passwords for qycgdfhz iqshzdru using pa...	the	qycgdfhz iqshzdru	GRP_17	3	1	84	
5708	reset passwords for bxeagsmt zrwdgsco using pa...	the	bxeagsmt zrwdgsco	GRP_17	3	1	84	
5839	reset passwords for cpmaidhj elbaqmtp using pa...	the	cpmaidhj elbaqmtp	GRP_17	3	1	84	
5884	reset passwords for bxeagsmt zrwdgsco using pa...	the	bxeagsmt zrwdgsco	GRP_17	3	1	84	

	short_description	description	caller	group	char_length	word_length	short_char_length	short_word
6037	reset passwords for cesgrtar abgrtyreu using p...	the	btvmxdfc yfahetsc	GRP_17	3	1	85	
6058	reset passwords for bxeagsmt zrwdgsco using pa...	the	bxeagsmt zrwdgsco	GRP_17	3	1	84	
6693	reset passwords for pzjelyxg vstyaouc using pa...	the	pzjelyxg vstyaouc	GRP_17	3	1	84	
6764	reset passwords for horeduca ogrhivnm using pa...	the	horeduca ogrhivnm	GRP_17	3	1	84	
6819	reset passwords for wvdxnkhf jirecvta using pa...	the	wvdxnkhf jirecvta	GRP_17	3	1	84	
6963	reset passwords for patrcja szpilewska using p...	the	lmsxcvoz vzhkdpfn	GRP_17	3	1	85	
7131	reset passwords for ezrsdgfc hofgvwel using pa...	the	ezrsdgfc hofgvwel	GRP_17	3	1	84	
7132	reset passwords for ezrsdgfc hofgvwel using pa...	the	ezrsdgfc hofgvwel	GRP_17	3	1	84	
7169	reset passwords for andrdgrtew p taneghrty usi...	the	tjzohmve wusgaozx	GRP_17	3	1	89	
7630	reset passwords for jcmxerol nbfyczqr using pa...	the	jcmxerol nbfyczqr	GRP_17	3	1	84	
7875	reset passwords for esias bosch using password...	the	paqrentz gcnyaxsb	GRP_17	3	1	78	
8059	reset passwords for wptbgchj jutpdcqf using pa...	the	wptbgchj jutpdcqf	GRP_17	3	1	84	
8092	reset passwords for prgthyuulla ramdntythanjes...	the	boirqctx bkijgqry	GRP_17	3	1	94	
8093	reset passwords for prgthyuulla ramdntythanjes...	the	boirqctx bkijgqry	GRP_17	3	1	94	
8168	reset passwords for kevguind l gineman using p...	the	nckihpba czrdksex	GRP_17	3	1	85	

In [36]:
```
dataset[dataset.description == 's']  # holds no actual information with just one letter, has to
```

Out[36]:

	short_description	description	caller	group	char_length	word_length	short_char_length	short_word_l
1860	s	s	gzjtweph mnslwfqv	GRP_0	1	1	1	

In [37]:
```
# drop row with description: 's'
dataset.drop(dataset[dataset.description == 's'].index, inplace=True)
dataset = dataset.progress_apply(get_length, axis=1)
```

```
100%|██████████| 8499/8499 [0
0:01<00:00, 4780.09it/s]
```

In [38]:
```
# description 'the' holds no information => imputed with corresponding short_description
dataset.loc[dataset.description == 'the', 'description'] = dataset.loc[dataset.description == '
dataset = dataset.progress_apply(get_length, axis=1)
```

```
100%|██████████| 8499/8499 [0
0:01<00:00, 5458.59it/s]
```

In [39]:
```
# Single Word descriptions
dataset[dataset.word_length == 1].shape
```

Out[39]:
```
(41, 8)
```

In [40]:
```
# dataset[dataset.word_length == 1].to_csv('./data/Single_Word_Descriptions.csv')
dataset[dataset.word_length == 1].sample(20)  # these have to be cleaned up and imputed later c
```

Out[40]:

	short_description	description	caller	group	ch
2267	urgent help required- outlook to crm mfg_toolt...	contact	gonflcmq wmptisvz	GRP_0	
1704	è¨¦æˆ·è¢«é” å®šï	ç”¨æˆ·è¨¦é” å®šï¼Œè¯·è§£é”	yvscpgax wdfxytzu	GRP_48	
1178	ç”µè¯ æœºæ²¡æœ‰å£°éŸ³	ç”µè¯ æœºæ²¡æœ‰å£°éŸ³	cyjlqdwm kywiuosn	GRP_30	
1081	ç”¨å ‹è½¯ä»¶é—®é¢˜	æ‰“å¼€å…¬ç”¨é”®ç›˜ä¸ç¬¦å•	bwstnmjh yqumwrsk	GRP_48	
3315	ç”µè„‘ç»ç»Ÿ ¯åŠ¨è" å± ã€,	ç”µè„‘ç»ç»Ÿ ¯åŠ¨è" å± ã€,æ°´ä¸ å° å¿ƒæ´'å^°ç...	hdungfsc znuhyjkx	GRP_31	
1955	æœ‰ä¸€ä¸ªé¾¾æŽ¥æ– ‡ä»¶æ‰“ä¸ å¼€	æœ‰ä¸€é¾¾æŽ¥æ– ‡ä»¶æ‰“ä¸ å¼€ï¼Œæ ç¤ºç‰ˆæœ¬ä½Ž	qsfcxzel quwykhno	GRP_30	
5149	å¼€ä¸ äºº†æœºº	å¼€ä¸ äºº†æœººï¼Œæ˜¾ç¤ºç³»ç»Ÿ äºº†ã€,	kclhqspo xvugztyc	GRP_30	
7302	skypeä¼šè®®æ—¶ä¸ åŽ»	skypeä¼šè®®ä»Žé‚®ç®±é‚€è¯·ä¸ åŽ»ã€,	rekpvblc ufysatml	GRP_30	
5491	é›»è…¦å‡ºç ¾è— å± ,ç„¡æ³•é–‹æ©Ÿ	é€£vpnæ™„ç„¡æ³•é€£åŠå¾Œ,é‡ è©¦å¾Œ,çª ç„¶å‡ºç...	zhpwcdea cboefuis	GRP_31	
7317	ç”µè„‘æ•…éšœ	·¨æŽ§éƒ¨æ‹‰å‰›è¯•éæŒæœºæŽ§å¶¶ç"µè„‘çš„æ" ä½œç...	kwpzbxvf cvuhoizx	GRP_48	
4503	ç"¨å ‹å®šæœŸå¤‡ä»½ä¸ æ^ åŠŸ	æœ åŠ¡å™¨ç«¯ç"¨å ‹ç³»ç»Ÿæœ€è¿'å Šä¸ªæœˆæ— ¥å¤‡ä...	igdnsjhz awnftgev	GRP_48	
1399	i cant see my archived emails in outlook. i a...	outlook	koiapqbg teyldpkw	GRP_0	
5761	ä¸»æœºä¸ èƒ½å¼€å ¯ï¼Œç"µæº ç ¯æ-£å„ï¼Œä¸»æœºé...	cpdilmjx jwsqpiac	GRP_48		

	short_description	description	caller	group	ch
3120	ç"µè„'ç¡¬ç›˜æ•…éšœï¼Œè¯·æ±‚,ç»´ä¿®ã€‚	ç"µè„'ç¡¬ç›˜æ•…éšœï¼Œè¯·æ±‚,ç»´ä¿®ã€‚	ruhbyzpv vlksnjti	GRP_30	
4569	i am not able to connect to my regular printer...	x5380	koiapqbg teyldpkw	GRP_0	
416	reset the password for prgewfly ndtfvple on er...	completed	prgewfly ndtfvple	GRP_0	
6106	ç"µè„'ä¸ è¯ï¼å¼€æœº	æ—©ä¸Šç ç"µè„'æ‰"ä¸ å¼€ã€‚	mzerdtop xnlytczj	GRP_30	
6534	æ¶,å±,ã€ ç®¡ä¸ è½¦é—´ç"µè¯ æ•…éšœ	æ¶,å±,ã€ ç®¡ä¸ è½¦é—´ç"µè¯ æ•…éšœï¼Œ40634943ã€...	vrmpysoz qkiucpdx	GRP_48	
4505	request to reset microsoft online services pa...	\r\n\r\nkind	rcfwnpbi kvhedyrc	GRP_0	
6253	in the inbox always show there are several ema...	+86	mqbxwpfn uclrqfxa	GRP_0	

In [41]:
```
dataset[dataset.word_length == 2].sample(20)
```

Out[41]:

	short_description	description	
4414	mii update	mii update	d e
2466	outlook error	outlook error	p t
6286	bios update	bios update	j rz
3879	erp ticket	erp ticket	ne xp
3419	office reinstall	office reinstall	)
4951	german call	german call	rb gu
4400	password reset	password reset	he r
3740	upsæ•…éšœ	åˆ¶ç²‰è½¦é—´3æ¥¼psfäº¤æ ¢æœºå¤„upsæ•…éšœï¼Œ®¾...	ag mx
5973	outlook freezes.	outlook freezes.	rv
6176	ie settings	ie settings	j
1136	system freezing	system freezing	bi kb
6625	account locked.	account locked.	vf nt
5775	account unlock	account unlock	h om
4502	é"¸é€ è½¦é—´ç"µè„'æ•…éšœ	é"¸é€ è½¦é—´è®°å½•ç"Ÿäº§æ•°æ ®çš„ç"µè„'ä¸ è¯ï¼å...	r tu
5890	iphoneä¸Šçš„skypeä¸ è¯ï¼™»å½•ä¸ è¯ï¼å ,åŠ ä¼šè®®...	iphoneä¸Šçš„skypeä¸ è¯ï¼™»å½•ä¸ è¯ï¼å ,åŠ ä¼šè®®...	h n

	short_description	description	
2141	blank call	blank call	rl gı
859	engineering_tool installation	engineering_tool installation	mc b
7824	account locked.	account locked\r\n	tf b
7694	account locked	account locked	n vr
4956	password reset	password reset	a f

In [42]:
```
dataset[dataset.short_word_length == 0]  # all have short descriptions
```

Out[42]:

short_description	description	caller	group	char_length	word_length	short_char_length	short_word_length

In [43]:
```
dataset[dataset.short_char_length < 3]
```

Out[43]:

	short_description	description	caller	group	char_length	word_length	short_char_length	short_word_l
3529	PR	create a purchase requisition with purchasing ...	ejvkzobl yijgokrn	GRP_29	201	31	2	

In [44]:
```
dataset[dataset.char_length < 4]
```

Out[44]:

	short_description	description	caller	group	char_length	word_length	short_char_length	short_word_l
3432	dds	dss	onctqhsg cpahzsle	GRP_0	3	1	3	

In [45]:
```
dataset[dataset.word_length > 800]  # security incident logs
```

Out[45]:

	short_description	description	caller	group	char_length	word_length	short_char_length
3530	security incidents - (#in33944691) : possibl...	source ip: 195.272.28.222\nsource port: 80\nso...	gzhapcld fdigznbk	GRP_2	7524	974	111
3965	security incidents - (#in33809307) : possibl...	source ip :195.22.28.222 \nsystem name :androi...	gzhapcld fdigznbk	GRP_2	8988	1255	116
4087	security incidents - (sw #in33895560) : mage...	source ip : 172.20.10.37 , 208.211.136.158\nsy...	ugyothfz ugrmkdhx	GRP_39	11968	1547	63
4089	security incidents - (sw #in33895560) : mage...	source ip : 172.20.10.37 , 208.211.136.158\nsy...	ugyothfz ugrmkdhx	GRP_2	11968	1547	63

	short_description	description	caller	group	char_length	word_length	short_char_length
5092	security incidents - (#in33578632) : suspicio...	source ip: 29.26.13.3095\r\nsource hostname: H...	gzhapcld fdigznbk	GRP_3	9063	1066	92
5433	security incidents - (#in33765965) : possibl...	source ip :10.40.6.221\nsystem name :rqxl85172...	gzhapcld fdigznbk	GRP_2	8575	1057	83
7345	security incidents - (sw #in33501789) : broa...	we are seeing activity indicating the host at ...	ugyothfz ugrmkdhx	GRP_2	13001	1625	102
7647	security incidents - (#in33578632) : suspicio...	source ip :\r\nsystem name :\r\nuser name:\r\...	gzhapcld fdigznbk	GRP_2	8991	993	92
7982	security incidents - (dsw #in33390850) : sus...	source ip : 78.83.16.293\nsystem name : HostNa...	ugyothfz ugrmkdhx	GRP_2	9881	1137	118
7984	security incidents - (dsw #in33390850) : sus...	source ip : 78.83.16.293\r\nsystem name : Host...	ugyothfz ugrmkdhx	GRP_12	10077	1137	118
7989	security incidents - (dsw #in33407676) : tra...	source ip : 61.01.52.02617\r\nsystem name : lp...	ugyothfz ugrmkdhx	GRP_2	9440	1275	109
7995	security incidents - (dsw #in33407676) : tra...	source ip : 61.01.52.02617\r\nsystem name : lp...	ugyothfz ugrmkdhx	GRP_62	9440	1275	109
7997	security incidents - (sw #in33544563) : poss...	source ip : 45.25.35.0499\nsystem name : lpal9...	ugyothfz ugrmkdhx	GRP_2	9678	1260	107
8002	security incidents - (sw #in33544563) : poss...	source ip : 45.25.35.0499\r\nsystem name : lpa...	ugyothfz ugrmkdhx	GRP_62	9912	1260	107

In [46]:
```
dataset[dataset.word_length > 200][dataset.word_length < 800]
```

Out[46]:

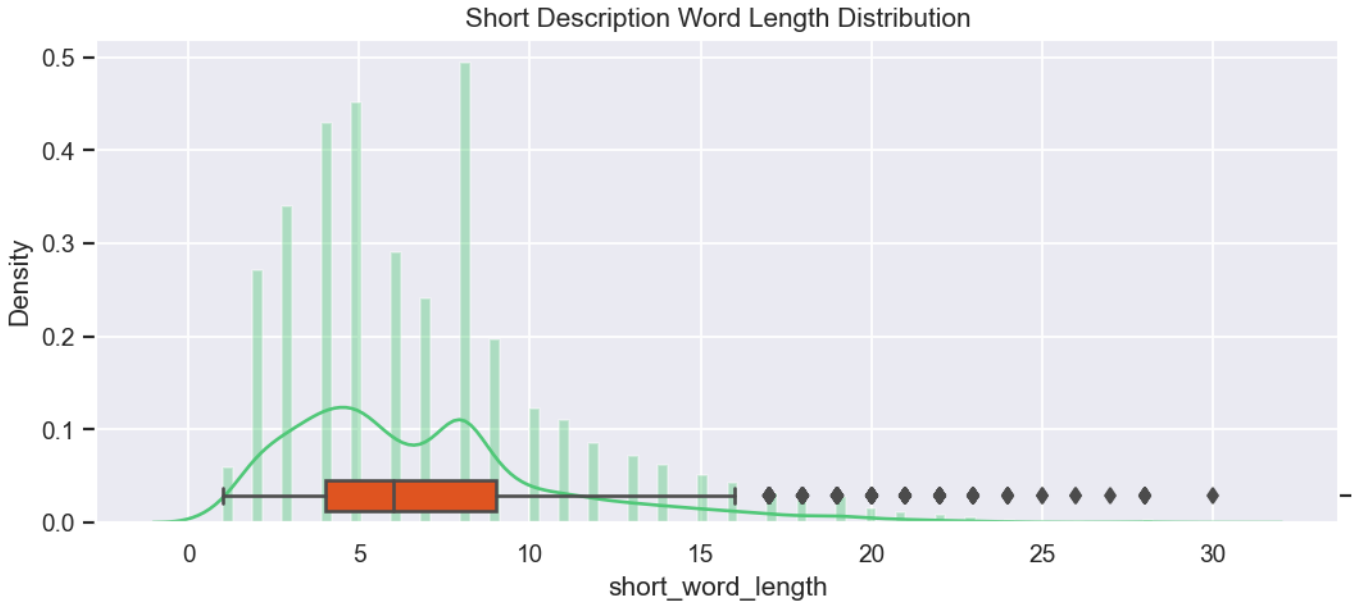
	short_description	description	caller	group	char_length	word
238	erp pi and msd crm connectivity issue-serirtc...	hi all\n\nwe have a connectivity issue between...	kgytujhe bonhwzrx	GRP_14	2007	
239	printer problem / issue information	please complete all required questions below. ...	dzjxrkae grqczsmx	GRP_3	1294	
981	employment status - new non-employee ycgkinov ...	*page down to ensure that all required data fi...	lfikjasz tjbqcmvl	GRP_2	1324	
1175	bitte das iphone-6 001 freischalten fÃ¼r mail-...	\n\nreceived from: rtnyumbg.yzemkhbq@gmail.com...	rtnyumbg yzemkhbq	GRP_0	1828	
1179	media server disconnect lpapr952(south_amerirt...	received below email from inin tried to ping t...	jloygrwh acvztedi	GRP_8	1839	

	short_description	description	caller	group	char_length	word
1577	the printer is defaulting to the usa printer f...	from: kryuisti turleythy \nsent: wednesday, oc...	hybiaxlk lawptzir	GRP_18	2172	
1696	mm:pur_req_ko assign for user: yeyhtung kimthy...	mm:pur_req_ko assign for user: yeyhtung kimthy...	kmnsvzuq euyvihzc	GRP_29	1379	
1812	sales orders are not updating with correct del...	this is in reference to so# 35086652\r\n\r\nth...	amunklhx bvrachko	GRP_6	1330	
1855	printer problem / issue information -- zebra l...	please complete all required questions below. ...	okfmbqur efzukjsa	GRP_0	1263	
2082	printing language sa38 (reporting rfumsv00)	please complete all required questions below. ...	ojhiaubp lovgirtm	GRP_33	1335	
2370	update inwarehouse_tool documents from list fo...	with the germany office move the main inwareho...	mynfoicj riuvxdas	GRP_13	1460	
2445	vh 27 - werk germany - fehlende druckauftrÃ¤ge...	\r\nbei drucker vh 27 keine ausgabe der drucka...	ucawbivs ountxzir	GRP_0	1321	
2492	printing request - request transaction print t...	please complete all required questions below. ...	omiwzbue auvolfhp	GRP_45	1297	
2741	i have lost my access to reporting_tool in crm...	from: dthyan matheywtyuews \nsent: thursday, s...	oetlgbfw bsctrnwp	GRP_0	1877	
2879	mobile device activation	from: tsbnfixp numwqahj \nsent: wednesday, sep...	tsbnfixp numwqahj	GRP_0	3141	
2978	security incidents - (#in33987594) : 29866 vi...	source ip :\r\nsystem name :\r\nuser name:\r\...	gzhapcld fdigznbk	GRP_3	3249	
3097	security incidents - (#in33976733) : suspicio...	source ip: 10.16.90.249\r\nsource hostname: an...	gzhapcld fdigznbk	GRP_56	6887	
3098	security incidents - (#in33984033) : internal...	source ip :\r\nsystem name :\r\nuser name:\r\...	gzhapcld fdigznbk	GRP_19	6868	
3165	partial confirmation info sent to erp but mach...	usa go-live week. issue reported on 9/21\r\n\r...	entuakhp xrnhtdmk	GRP_41	1314	
3325	it help	\n\nreceived from: notwkdgr.zvmesjpt@gmail.com...	notwkdgr zvmesjpt	GRP_26	7467	
3382	support with	\n\nreceived from: jogtse.mhytusa@company.com\...	kwyozxgd gasxctph	GRP_25	1526	
3532	security incidents - (#in33944327) :possible ...	source ip :\r\nsystem name :\r\nuser name:\r\...	gzhapcld fdigznbk	GRP_2	3628	
3705	security incidents - (#in33932723) : possibl...	source ip: 10.44.63.52\r\nsource hostname: lee...	gzhapcld fdigznbk	GRP_48	3235	
3706	security incidents - (#in33924718) : possibl...	source ip :195.22.28.222\r\ndestination ip: 12...	gzhapcld fdigznbk	GRP_2	4286	
3718	re: need a little help--please	\n\nreceived from: bcefayom.lzhwcgvb@gmail.com...	bcefayom lzhwcgvb	GRP_18	2292	
3961	security incidents - (#in33805815) : possible...	==========\r\nevent data\r\n===...	gzhapcld fdigznbk	GRP_2	3734	

	short_description	description	caller	group	char_length	word
4382	printer problem / issue information	please complete all required questions below. ...	kpogxqvnsfzjbhet	GRP_3	1692	
4730	security incidents - (#in33847938) : possibl...	source ip :195.22.28.222\r\nsource port: 80\r\...	gzhapcldfdigznbk	GRP_31	4169	
4825	incident #in33541962 - phishing form submit -...	source ip: 10.38.93.30\nsource hostname: dane-...	ugyothfzugrmkdhx	GRP_2	2494	
4853	bahdqrcs xvgzdtqj's onbankrding experience	\r\n\r\nreceived from: xzupryaf.vlbikhsm@gmail...	xzupryafvlbikhsm	GRP_0	2548	
4886	security incidents - (#in33826812) : possibl...	source ip :83.54.03.93209 \nsystem name :rgtw8...	gzhapcldfdigznbk	GRP_3	1838	
4893	security incidents - (#in33826812) : possibl...	source ip :83.54.03.93209 \nsystem name :rgtw8...	gzhapcldfdigznbk	GRP_2	1837	
5072	erp-step interface programdnty not sending all...	erp-step interface programdnty is not generati...	rcivkdxohlyrieck	GRP_11	2786	
5204	employment status - three new non-employee [en...	*page down to ensure that all required data fi...	lbqgystkuezmfhsn	GRP_2	3620	
5394	hana	\n\nreceived from: nealxjbc.owjduxai@gmail.com...	nealxjbcowjduxai	GRP_9	1156	
5485	printer problem / issue information	please complete all required questions below. ...	mfixrouydyifhcjt	GRP_0	1347	
5503	dsw in22210104	we are seeing your 10.16.4.16/isensor04.compan...	afkstcevutbnkyop	GRP_2	2013	
5504	incident #in33541962 - phishing form submit -...	we are seeing your 18.79.63.203/company-intern...	afkstcevutbnkyop	GRP_2	2293	
5506	dsw in22457494	dsw in33568505\r\n\r\nwe are seeing your 172.2...	afkstcevutbnkyop	GRP_2	1495	
5507	possible vulnerability scan from host.my-tss.c...	dsw in33568733\r\n\r\nwe are seeing your 208.2...	afkstcevutbnkyop	GRP_2	2833	
5697	printer problem / issue information	please complete all required questions below. ...	gljrdmnuyfnbkcmp	GRP_0	1397	
5787	windows asks to install driver and then won't ...	please complete all required questions below. ...	rxqtvanckthqwxvb	GRP_0	1398	
6017	open order schedule lines_p2016-08-28-22-03-54	hallo ruchitgrr, hallo frau haug,\n\nleider en...	anivdcorrbmfhiox	GRP_9	1952	
6734	security incidents - (dsw incident no) : sus...	==========\nincident overview\n=...	gzhapcldfdigznbk	GRP_12	5084	
6888	security incidents - (#in33655554) : errata se...	==========\nincident overview\n...	gzhapcldfdigznbk	GRP_2	2744	
6931	'51551 vid67965 microsoft windows httpsys rce ...	dsw in33568767\n\nincident overview\n=========...	afkstcevutbnkyop	GRP_12	2672	
7163	symantec endpoint encryption (see) agent roll ...	\n\nreceived from: yqlvfkih.folbpugd@gmail.com...	yqlvfkihfolbpugd	GRP_0	3062	
7331	security incidents - (#in33505432) : repeat ...	source ip :10.16.140.231\nsystem name :evhl811...	gzhapcldfdigznbk	GRP_2	4245	

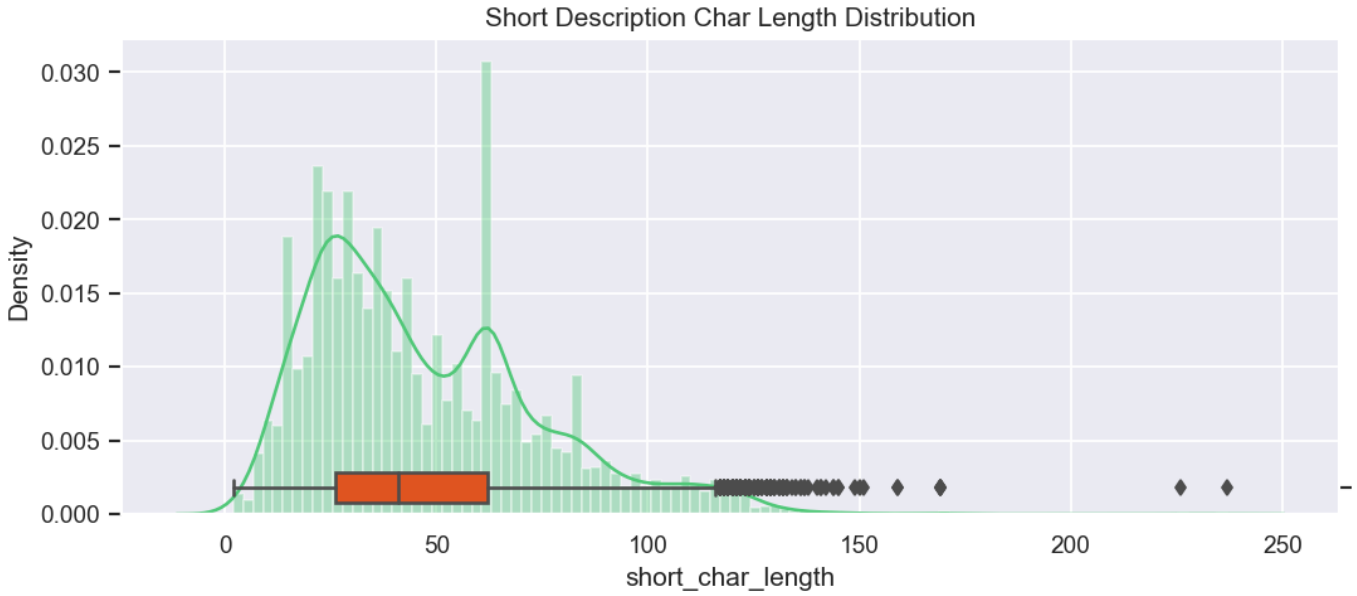
	short_description	description	caller	group	char_length	word
7338	security incidents - (#in33505432) : repeat ...	source ip :10.16.140.231\r\nsystem name :evhl8...	gzhapcld fdigznbk	GRP_2	4766	
7433	zpdist_programdnty not allowing to distribute ...	\r\nhello chandruhdty, ebi,\r\n\r\niÂ´ve creat...	cfajzero vlygoksi	GRP_18	1795	
7553	wifi slow speed-company (apac)	it team,\n\nplease kindly check internet for u...	przndfbo pldqbhtn	GRP_4	1659	
7981	as per inc1530176::security incidents - (in335...	\nfrom: gzhapcld fdigznbk \nsent: wednesday, a...	gzhapcld fdigznbk	GRP_2	1634	
7987	security incidents - (in33536629) : possible t...	source ip :10.44.94.214\r\ndest ip : 183.91.33...	gzhapcld fdigznbk	GRP_30	3403	
7991	as per inc1530161::security incidents - (in33...	\r\nfrom: gzhapcld fdigznbk \r\nsent: wednesda...	gzhapcld fdigznbk	GRP_2	5087	
7996	security incidents - (in33490582) : suspicio...	source ip : 29.26.13.3095\r\nsystem name :Host...	gzhapcld fdigznbk	GRP_12	7403	
8160	release of device	\r\n\r\nreceived from: qpixeudn.rjlziysd@gmail...	qpixeudn rjlziysd	GRP_0	3019	
8232	stepfhryhan needs access to below collaboratio...	stepfhryhan needs access to below collaboratio...	nizholae bjnqikym	GRP_0	3116	
8339	unlock account email in cell phone the users	hello team,\n\ncould you please unlock account...	qasdhyzm yuglsrwx	GRP_0	1521	

In [47]:
```
sns.set()
plt.figure(figsize=(10, 4), dpi=125)
ax = sns.distplot(dataset.short_word_length, bins=100, kde=True, color='#50C878')
ax_ = ax.twinx()
sns.boxplot(dataset.short_word_length, color='#FF4500')
ax_.set(ylim=(-.7, 12))
plt.title('Short Description Word Length Distribution')
plt.show()
```



In [48]:
```
sns.set()
plt.figure(figsize=(10, 4), dpi=125)
ax = sns.distplot(dataset.short_char_length, bins=100, kde=True, color='#50C878')
ax_ = ax.twinx()
```

```
sns.boxplot(dataset.short_char_length, color='#FF4500')
ax_.set(ylim=(-.7, 12))
plt.title('Short Description Char Length Distribution')
plt.show()
```



In [49]:
```
dataset.describe()
```

Out[49]:

	char_length	word_length	short_char_length	short_word_length
count	8499.000000	8499.000000	8499.000000	8499.000000
mean	204.595246	27.331804	47.386751	6.951759
std	519.530803	65.026307	27.323700	4.158631
min	3.000000	1.000000	2.000000	1.000000
25%	42.000000	6.000000	26.000000	4.000000
50%	106.000000	11.000000	41.000000	6.000000
75%	201.000000	28.000000	62.000000	9.000000
max	13001.000000	1625.000000	237.000000	30.000000

- Most descriptions have between 6 and 28 words long with median at 41 (106 characters) and mean at 27.2 with relatively few outliers ranging till 1625 words!
- Most Short descriptions have between 4 and 9 words long with median at 6 (41 characters) and mean at 6.92 with relatively few outliers ranging till 28 words.

In [50]:
```
def clean_nonsensible_outliers():
    # impute the zero length description with corresponding short description
    dataset.loc[dataset.word_length == 0, 'description'] = dataset.loc[dataset.word_length == 0
     # drop row with description: 's'
    dataset.drop(dataset[dataset.description == 's'].index, inplace=True)
    # description 'the' holds no information => imputed with corresponding short_description
    dataset.loc[dataset.description == 'the', 'description'] = dataset.loc[dataset.description

# clean_nonsensible_outliers()
```

• Fix Text Encoding

In [51]:
```
dataset[dataset.word_length == 1].sample(20)
```

Out[51]:

	short_description	description	caller

	short_description	description	caller	
3432	dds	dss	onctqhsg cpahzsle	
8266	erpæ— æ³•è¿›è¡Œéžï¼ˆè½¬ç»™è´¹æ£åˆ³ï¼‰	è¿›è¡Œéžæ—¶æ˜¾ç¤ºçš„æ‰¾å›žå¯†ç ...	kyagjxdh dmtjpbnz	
1081	ç”¨å ‹è½¯ä»¶é—®é¢˜	æ‰“å¼€å·²å…³é—çš„é”®è®¢å •æ—¶ï¼Œæ˜¾ç¤ºçš„ä¸ ...	bwstnmjh yqumwrsk	
3738	ç”µè¯ æ•…éšœ	é¨æ£'è½¦é—´ç”µè¯ æ•…éšœï¼Œ39523850	sbkhjigv pbvlfcse	
3120	ç”µè„‘ç¡¬ç›˜æ•…éšœï¼Œè¯·æ±‚ ,ç»´ä¿®ã€,	ç”µè„‘ç¡¬ç›˜æ•…éšœï¼Œè¯·æ±‚,ç»´ä¿®ã€,	ruhbyzpv vlksnjti	
6534	æ¶å±,ã€ ç®¡ä¸ è½¦é—´ç”µè¯ æ•…éšœ	æ¶å±,ã€ ç®¡ä¸ è½¦é—´ç”µè¯ æ•…éšœï¼Œ40634943ã€...	vrmpysoz qkiucpdx	
5761	ä¸»æœºä¸ è¦ï½å¼€å ¯	ä¸»æœºä¸ è¦ï½å¼€å ¯ï¼Œç ¯æ£å„ï¼Œä¸»æœºé...	cpdilmjx jwsqpiac	
1452	è€ƒå‹¤ç»Ÿè¿›ä¸ åŽ»è¯·å¤„ç †è°¢è°¢ï¼	è€ƒå‹¤ç»Ÿè¿›ä¸ åŽ»è¯·å¤„ç †è°¢è°¢ï¼	spgdcvhb ocagnpmj	
1704	è´¦æˆ·é¢ å®š	ç”¨æˆ·è´¦æˆ·é¢ å®šï¼Œè¯·é¢	yvscpgax wdfxytzu	
4569	i am not able to connect to my regular printer...	x5380	koiapqbg teyldpkw	
6253	in the inbox always show there are several ema...	+86	mqbxwpfn uclrqfxa	
4501	åˆ¶ç‰‰ä¸‰æ¥¼æŽ§åˆ¶å®¤ç”µè„‘æ•…éšœ	åˆ¶ç‰‰ä¸‰æ¥¼æŽ§åˆ¶å®¤ç”µè„‘ä¸ è¦ï½å¼€å ¯ï¼Œç”æ...	agyvbnwz mxsonkdc	
5891	vpnä¸ è¦ï½ä½¿ç”¨ï¼Œè¯·ç»™å° è´	vpnä¸ è¦ï½ä½¿ç”¨ï¼Œè¯·ç»™å° è´	ehfvwltg eakjbtoi	
4503	ç”¨å ‹å®šæœŸå¤‡ä»½ä¸ æˆ åŠŸ	æœ åŠ¡å™¨ç«¯ç”¨ å ‹ä¸ªæœˆæˆæ—¥å¤‡...	igdnsjhz awnftgev	
7302	skypeä¼šè® ®æ—ä¸ åŽ»	skypeä¼šè® ®ä»Žé,®ç® ±é‡Œçš„é¾¾æŽ¥è¿›ä¸ åŽ»ã€,	rekpvblc ufysatml	
7317	ç”µè„‘æ•…éšœ	è´¨æŽ§éƒ¨æ‹›è¯•éªŒæœºæŽ§åˆ¶ç”µè„‘çš„æ ä½œæ...	kwpzbxvf cvuhoizx	
608	etiketten drucker im bereich endkontrolle germ...	funktionsstÃ¶rung	tzmewbdv zjbuwmkn	
5146	walkmeåŠ è½½æ•…éšœ	walkmeä¸è½½å®‰è£... å Žï¼Œæ‰€é'®ä¸ è¦ï½åœ¨æµ è§ˆå...	whflryeb fatgdzhq	
416	reset the password for prgewfly ndtfvple on er...	completed	prgewfly ndtfvple	
5762	æ‰“å¼€office 2013æ˜¾ç¤ºæ¯ æŽ¨æ ƒä°§å"	æ‰“å¼€outlookã€ pptæ˜¾ç¤ºæ¯ æœªç» æŽ¨æ ƒä°§å" ...	hbvwqine eakqyovu	

In [52]:
```
def fix_text_encoding(row):
    row['description'] = fix_text(row.description)
    row['short_description'] = fix_text(row.short_description)
    return row

dataset = dataset.progress_apply(fix_text_encoding, axis=1)
dataset[dataset.word_length == 1].sample(20)  # translated to proper unicode text in chineese
```

```
100%|██████████| 8499/8499 [00:05<00:00, 1479.63it/s]
```

Out[52]:

	short_description	description	caller	group	char_length	word_length	short_char_len
5491	電腦出現藍屏,無法開機	連vpn時,無法連上後,重試後,突然出現藍屏,無法開機	zhpwcdeacboefuis	GRP_31	67	1	
3315	电脑系统启动蓝屏。	电脑系统启动蓝屏。水不小心洒到电脑里面。	hdungfsczhuyjkx	GRP_31	60	1	
5833	new cpp id can not request initiative. see im...	cphlme01\n	pfzxecboptygkvzl	GRP_21	10	1	
5762	打开office 2013显示是未经授权产品	打开outlook、ppt显示是未经授权产品,望解决。	hbvwqineeakqyovu	GRP_48	59	1	
5311	系统故障,启动蓝屏.	系统故障,启动蓝屏.	lhkqbmnaekhtivsd	GRP_31	29	1	
5149	开不了机	开不了机,显示系统坏了。	kclhqspoxvugztyc	GRP_30	36	1	
8266	erp无法进行采购(转给贺正平)	进行采购时显示"找不到员工1111154833的数据,请通知系统管理员"	kyagjxdhdmtjpbnz	GRP_30	84	1	
6534	涂层、管丝车间电话故障	涂层、管丝车间电话故障,40634943、39523835	vrmpysozqkiucpdx	GRP_48	55	1	
415	reset passwords for prgewflyndtfvple using pa...	complete	prgewflyndtfvple	GRP_17	8	1	
5891	vpn不能使用,请转给小贺	vpn不能使用,请转给小贺	ehfvwltgeakjbtoi	GRP_0	33	1	
1955	有一个链接文件打不开	有一链接文件打不开,提示版本低	qsfcxzelquwykhno	GRP_30	45	1	
2915	websites not loading on company center	companycenter.company.com	qcfmxgidjvxanwre	GRP_0	25	1	
4501	制粉三楼控制室电脑故障	制粉三楼控制室电脑不能开启,电源指示灯桔色频闪。	agyvbnwzmxsonkdc	GRP_48	72	1	
4098	电脑意外进水,帮助处理!请交小贺,谢谢	电脑意外进水,帮助处理!请交小贺,谢谢	pvfclkmngebyipwr	GRP_30	57	1	
7969	客户提供的在线系统打不开	客户提供的在线送货单生成系统打不开,需尽快解决	fupikdoagjkytoeh	GRP_48	69	1	
1399	i cant see my archived emails in outlook. i a...	outlook	koiapqbgteyldpkw	GRP_0	7	1	
1178	电话机没有声音	电话机没有声音	cyjlqdwmkywiuosn	GRP_30	21	1	
276	outlook收到箱中folder变为每天一个folder,office提示更新。	outlook收到箱中folder变为每天一个folder,office提示更新。	bxfdkiolmdqlszvc	GRP_30	73	1	
3738	电话故障	铸棒车间电话故障,39523850	sbkhjigvpbvlfcse	GRP_48	35	1	
3120	电脑硬盘故障,请求维修。	电脑硬盘故障,请求维修。	ruhbyzpvvlksnjti	GRP_30	36	1	

- **Keyword Extraction**

In [168…]
```
print(test)
```

```
circuit outage: vogelfontein, south africa mpls circuit is down at 8:14 am et on 08/08
```

In [169…]
```
# !pip -q install yake
import yake

language = "en"
max_ngram_size = 5
duplication_threshold = 0.9
numOfKeywords = 1

custom_kw_extractor = yake.KeywordExtractor(lan=language,
                                            n=max_ngram_size,
                                            dedupLim=duplication_threshold,
                                            top=numOfKeywords,
                                            features=None)
```

In [170…]
```
k = custom_kw_extractor.extract_keywords(test)
k[0][0]
```

Out[170…]
```
'south africa mpls circuit'
```

In [54]:
```
def get_keywords(row):
    '''Keyword extraction on Keywords and Short Keywords'''
    description_keywords = custom_kw_extractor.extract_keywords(row.description)
    if len(description_keywords) == 0:
        description_keywords = ''
    elif len(description_keywords) == 1:
        description_keywords = description_keywords[0][0]
    else:
        description_keywords = ' '.join([i[0] for i in description_keywords])
        # print(row)
    row['description_keywords'] = description_keywords
    description_keywords = custom_kw_extractor.extract_keywords(row.short_description)
    if len(description_keywords) == 0:
        description_keywords = ''
    elif len(description_keywords) == 1:
        description_keywords = description_keywords[0][0]
    else:
        description_keywords = ' '.join([i[0] for i in description_keywords])
        # print(row)
    row['short_description_keywords'] = description_keywords
    return row

dataset = dataset.progress_apply(get_keywords, axis=1)
```

```
100%|██████████████████████████████████████████████████████████████████████████| 8499/8499
[01:21<00:00, 104.85it/s]
```

In [55]:
```
# TODO: clean caller ids, ..  from descriptions before keyword extraction?
dataset.sample(20)
```

Out[55]:

	short_description	description	caller	group	cha
3749	unable to log in to erp	unable to log in to erp	uerghtyi erzatqpy	GRP_0	
1057	verbindung zwischen drucker em98 und pc eemw81...	verbindung zwischen drucker em98 und pc eemw81...	nemzycxb xpsgkahw	GRP_0	
7289	connection to 'admin2-datacenter-switch04' ;sw...	connection to 'admin2-datacenter-switch04' (pi...	uvrbhlnt bjrmalzi	GRP_53	
5871	HostName_1010: erpstartsrv.exe service is down.	HostName_1010: erpstartsrv.exe service is down.	rkupnshb gsmzfojw	GRP_14	

	short_description	description	caller	group	cha
5855	outlook not loading	outlook not loading	cighytol yjurztgd	GRP_0	
3247	getting error while accessing it support revie...	pl refer attachment	ginjmaxk zumkvfeb	GRP_19	
3423	xerox in our office will not turn on, no power...	xerox in our office will not turn on, no power...	xzcwlqrv fjrdhiqt	GRP_3	
6935	114 occurrences of your firewall company-europ...	dsw in32075851\n\n==========\ni...	afkstcev utbnkyop	GRP_12	
3413	k100sfs.company.company.com is reporting a : ...	observing below alert in monitoring_tool since...	jloygrwh acvztedi	GRP_14	
127	update on inplant_874269	update on inplant_874269	rbozivdq gmlhrtvp	GRP_0	
3193	skype issue : personal certificate error.	\nsummary:i am unable to sign into skype - get...	grtaoivq dwjvfkqe	GRP_0	
4121	not able to submit reports in engineering_tool	not able to submit reports in engineering_tool...	efbwiadp dicafxhv	GRP_0	
1372	windows password reset	windows password reset	mxifcasu cxsembup	GRP_0	
541	unable to login to skype	unable to login to skype	ctepaurs igrazhwo	GRP_0	
6513	account lock out ; ad	account lock out ; ad	hzetqwba tmsbnfkh	GRP_0	
4645	job Job_2063c failed in job_scheduler at: 09/1...	received from: monitoring_tool@company.com\n\n...	bpctwhsn kzqsbmtp	GRP_6	
5146	walkme加载故障	walkme下载安装后,按钮不能在浏览器界面显现	whflryeb fatgdzhq	GRP_48	
585	activation of outlook email access on samsung ...	\n\nreceived from: kbclinop.vsczklfp@gmail.com...	kbclinop vsczklfp	GRP_0	
8119	company\orshop_floor_app locked out for too ma...	multiple computers use the same log in. compan...	xdswhrif ludsxncq	GRP_0	
7182	mobile device activation	have a new phone, my exchange server is blocke...	jvxtfhkg heptuizn	GRP_0	

- # Word Frequency Distributions & WordClouds

In [56]:
```
# top 50 most frequent words in text
top_N = 50

words = (dataset.description.str.cat(sep=' ').split())
rslt = pd.DataFrame(Counter(words).most_common(top_N),
                    columns=['Word', 'Frequency']).set_index('Word')
```

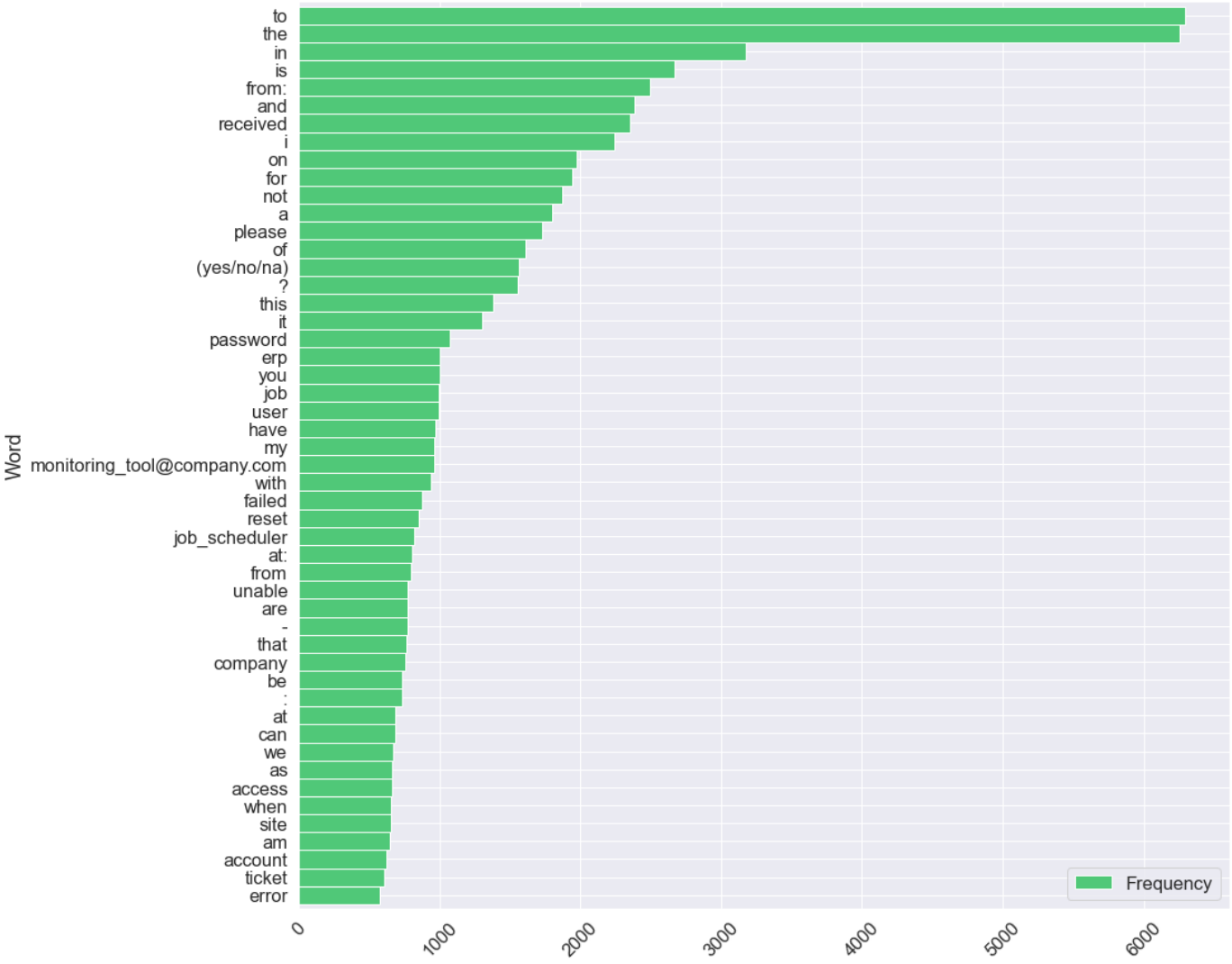
In [57]:
```
rslt[:50].transpose()
```

Out[57]:

Word	to	the	in	is	from:	and	received	i	on	for	...	can	we	as	access	when
Frequency	6296	6254	3172	2670	2499	2384	2354	2244	1977	1946	...	684	669	665	664	656

1 rows × 50 columns

In [58]:
```
sns.set(font_scale=1.5)  # scale up font size
rslt.sort_values(by='Frequency', ascending=True).plot(kind='barh', width=1, figsize=(15, 15), c
plt.xticks(rotation=45)
plt.show()
```



In [59]:
```
pprint(rslt.index.tolist(), compact=True)
```

```
['to', 'the', 'in', 'is', 'from:', 'and', 'received', 'i', 'on', 'for', 'not',
 'a', 'please', 'of', '(yes/no/na)', '?', 'this', 'it', 'password', 'erp',
 'you', 'job', 'user', 'have', 'my', 'monitoring_tool@company.com', 'with',
 'failed', 'reset', 'job_scheduler', 'at:', 'from', 'unable', 'are', '-',
 'that', 'company', 'be', ':', 'at', 'can', 'we', 'as', 'access', 'when',
 'site', 'am', 'account', 'ticket', 'error']
```

- Stopwords and Anchor words like 'From:', 'Recieved' have to be stripped out

In [60]:
```
# top 50 most frequent words in text
top_N = 50

words = (dataset.short_description.str.cat(sep=' ').split())
rslt = pd.DataFrame(Counter(words).most_common(top_N),
                    columns=['Word', 'Frequency']).set_index('Word')
```

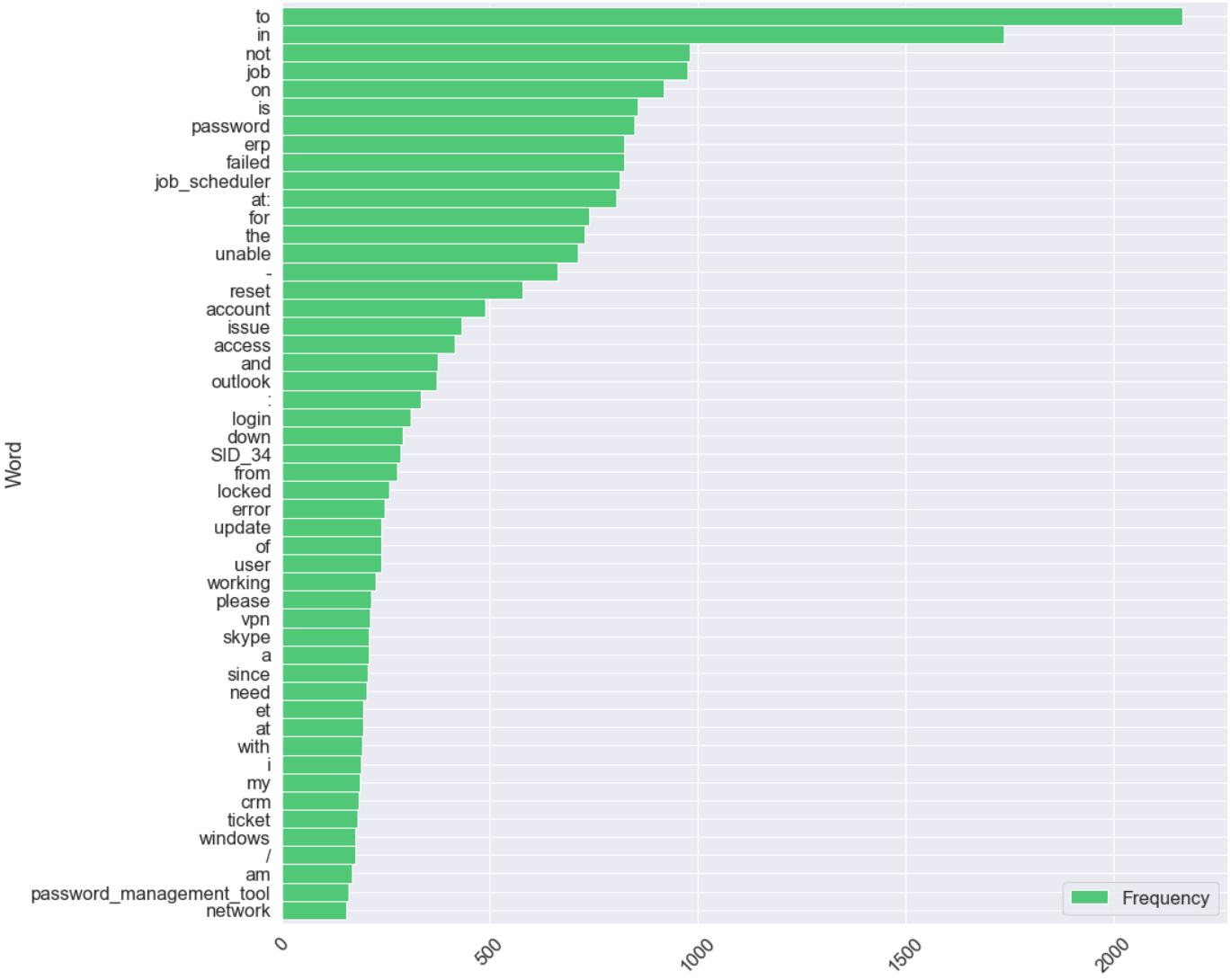
In [61]:
```
rslt[:50].transpose()
```

Out[61]:

Word	to	in	not	job	on	is	password	erp	failed	job_scheduler	...	with	i	my	crm	tick
Frequency	2167	1737	981	976	917	856	847	822	822	811	...	193	191	187	185	1

1 rows × 50 columns

In [62]:
```
sns.set(font_scale=1.5)  # scale up font size
rslt.sort_values(by='Frequency', ascending=True).plot(kind='barh', width=1, figsize=(15, 15), c
plt.xticks(rotation=45)
plt.show()
```



In [63]:
```
pprint(rslt.index.tolist(), compact=True)
```

```
['to', 'in', 'not', 'job', 'on', 'is', 'password', 'erp', 'failed',
 'job_scheduler', 'at:', 'for', 'the', 'unable', '-', 'reset', 'account',
 'issue', 'access', 'and', 'outlook', ':', 'login', 'down', 'SID_34', 'from',
 'locked', 'error', 'update', 'of', 'user', 'working', 'please', 'vpn', 'skype',
 'a', 'since', 'need', 'et', 'at', 'with', 'i', 'my', 'crm', 'ticket',
 'windows', '/', 'am', 'password_management_tool', 'network']
```

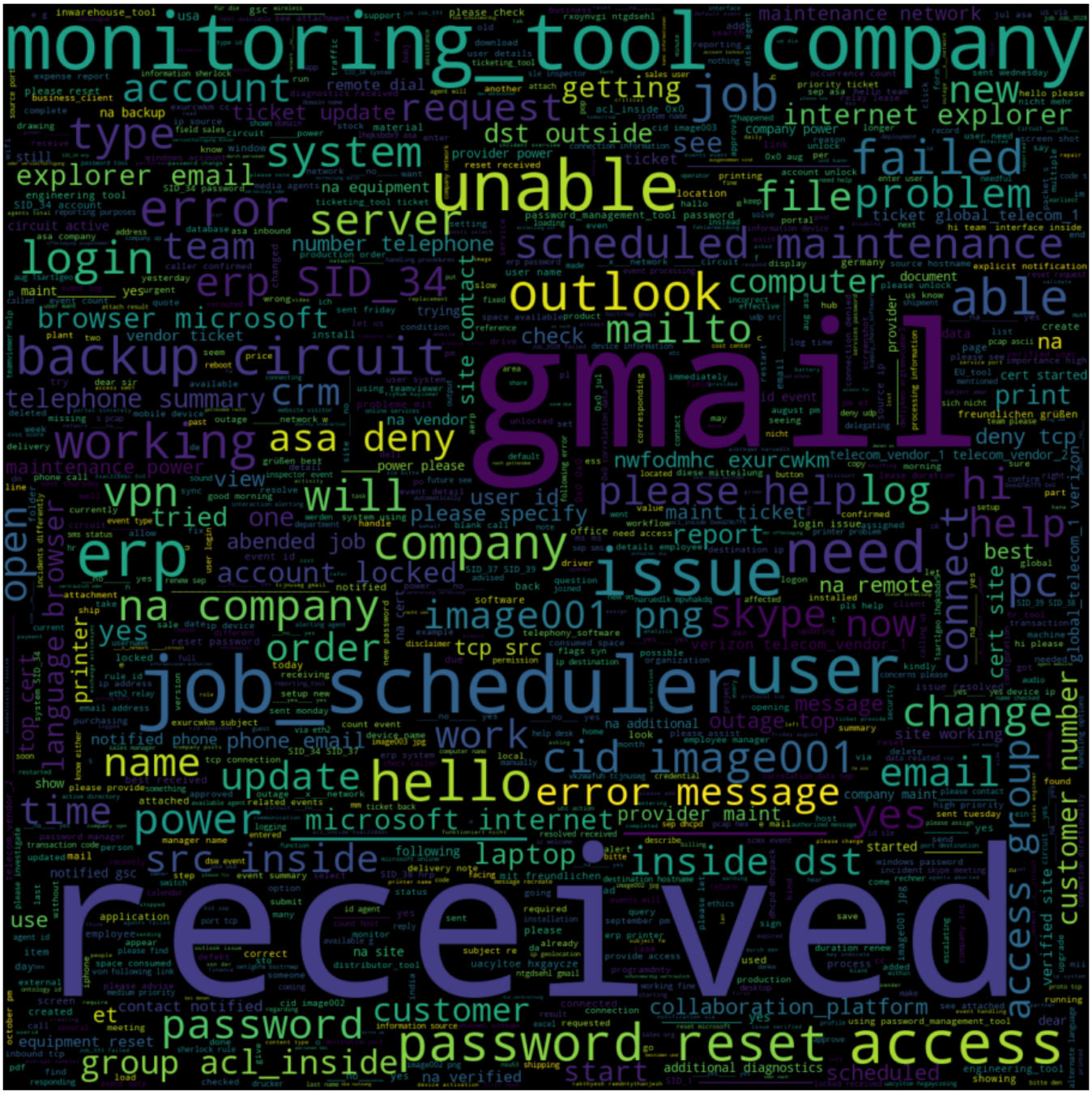
- Many stopwords are occuring most frequently in the dataset. We might need to use stopword removal in our pre-processing if it improves the model performance.

In [64]:
```
descr_string = ""
for description in dataset['description']:
    descr_string += description
    descr_string += " "

short_descr_string = ""
for description in dataset['short_description']:
    short_descr_string += description
    short_descr_string += " "

grp0_string = ""
for description in temp.loc[temp.group == 'Group 0', 'description']:
    grp0_string += description
    grp0_string += " "

other_string = ""
for description in temp.loc[temp.group == 'Other', 'description']:
```

```
    other_string += description
    other_string += " "
```

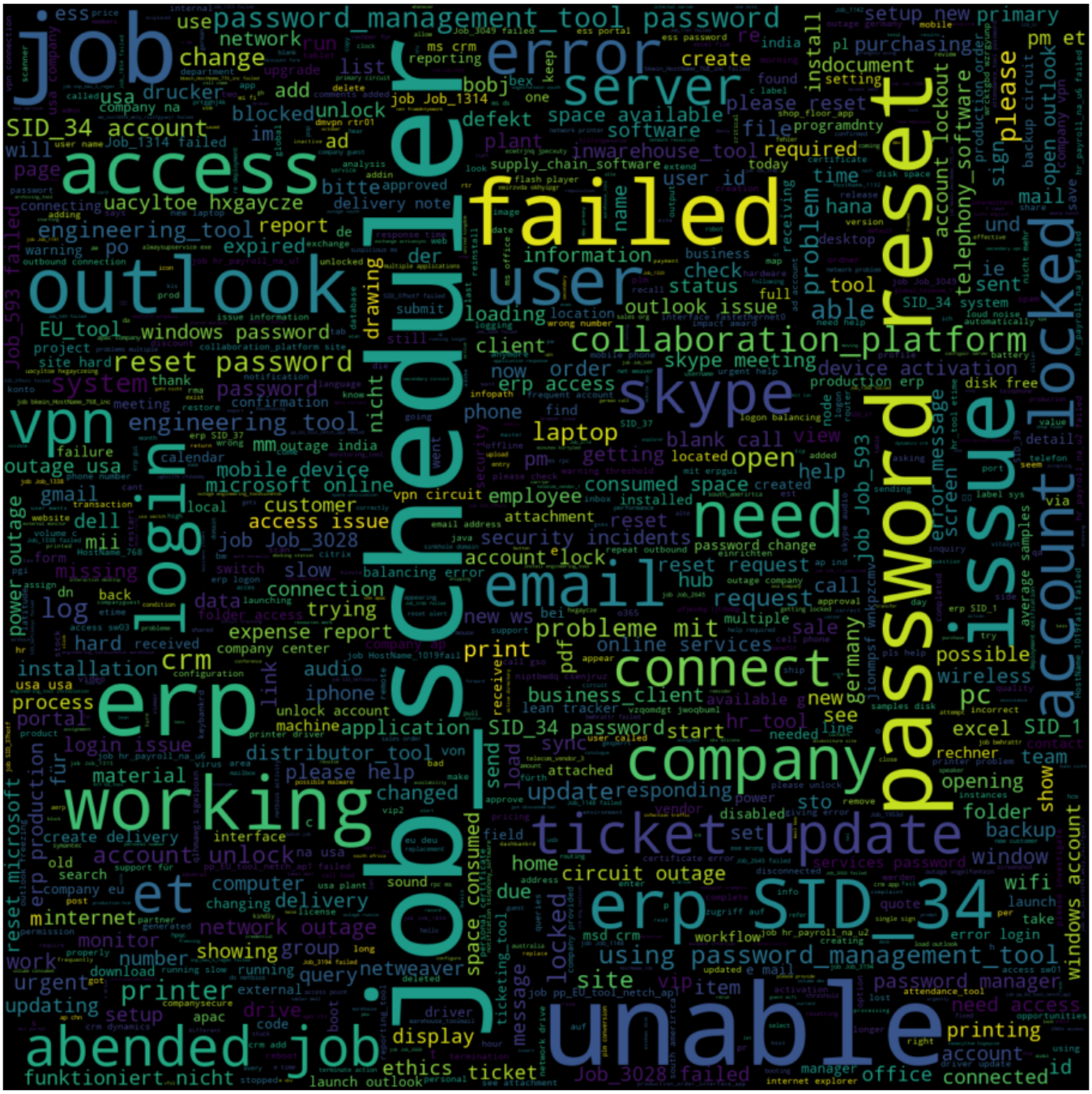
- **Descriptions WordCloud**

In [65]:
```
plt.figure(figsize=(10,10), dpi=120)
WC = WordCloud(width=1200, height=1200, max_words=1000, min_font_size=5)
plt.imshow(WC.generate(descr_string), interpolation='bilinear')
plt.axis("off")
plt.show()
```



- **Short Descriptions WordCloud**

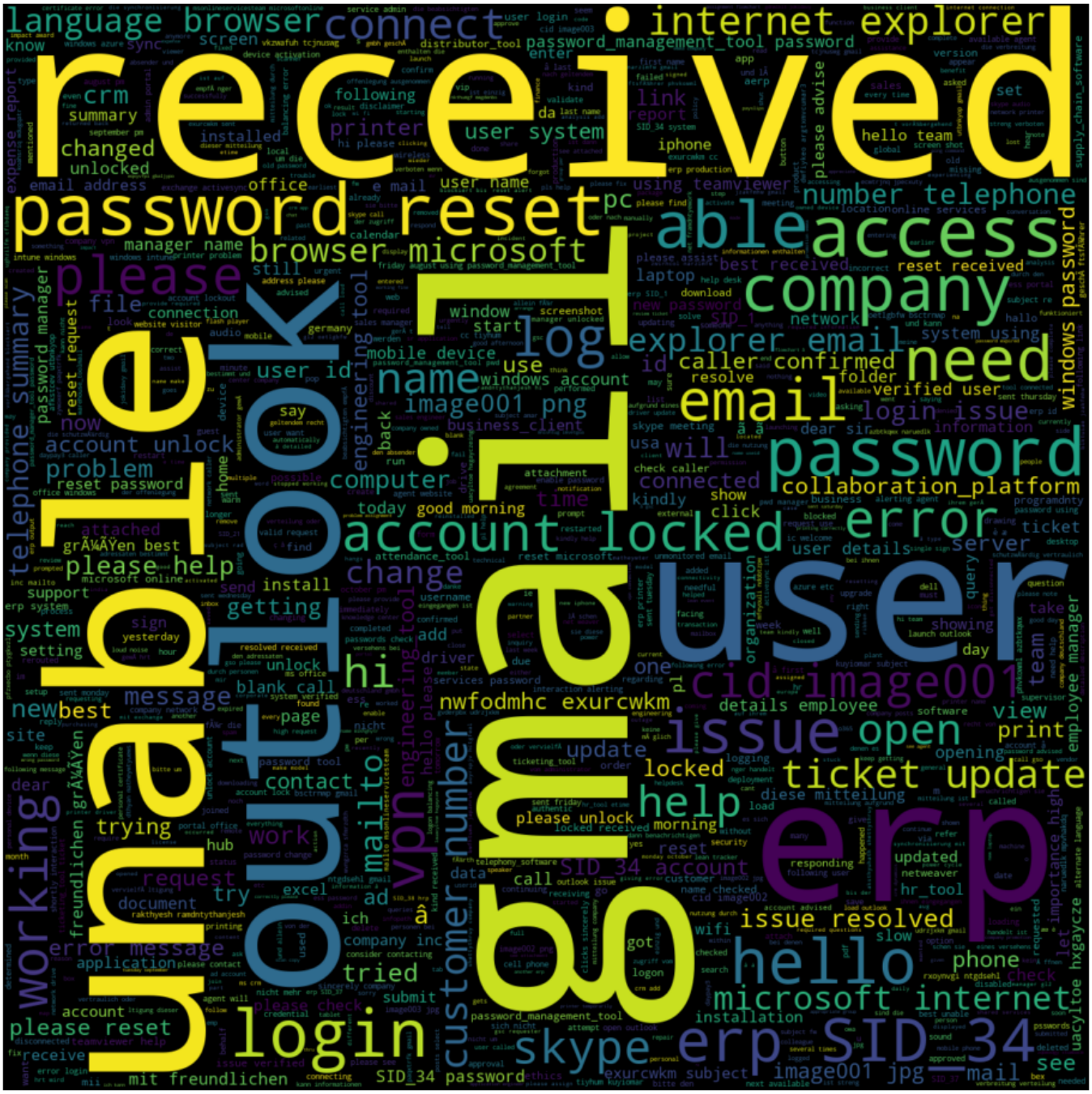
In [66]:
```
# Non-Sarcastic descriptions wordcloud
plt.figure(figsize=(10,10), dpi=120)
WC = WordCloud(width=1200, height=1200, max_words=1000, min_font_size=5)
plt.imshow(WC.generate(short_descr_string), interpolation='bilinear')
plt.axis("off")
plt.show()
```



- **Group 0 Descriptions WordCloud**

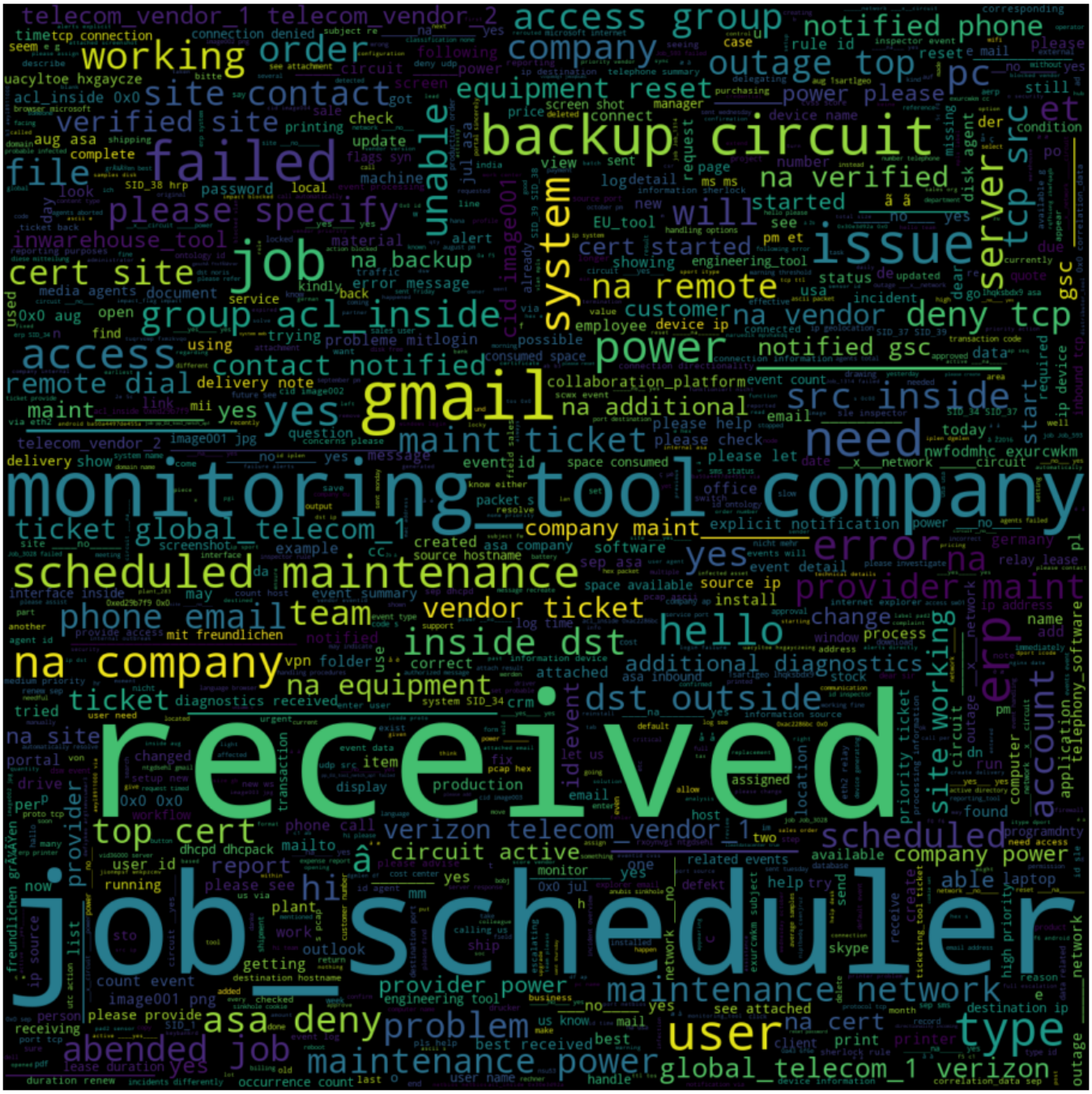
In [67]:

```
# Non-Sarcastic descriptions wordcloud
plt.figure(figsize=(10,10), dpi=120)
WC = WordCloud(width=1200, height=1200, max_words=1000, min_font_size=5)
plt.imshow(WC.generate(grp0_string), interpolation='bilinear')
plt.axis("off")
plt.show()
```



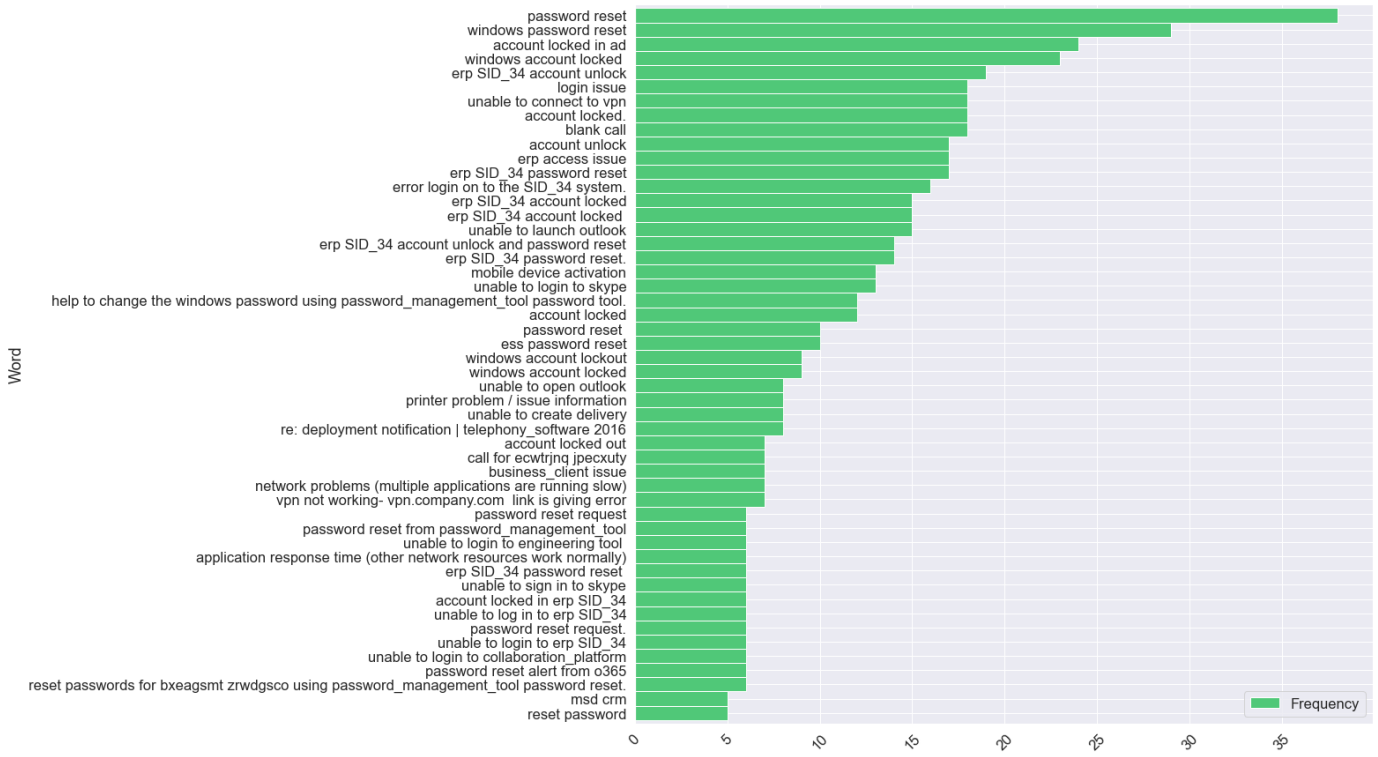
- **Other Groups Descriptions WordCloud**

In [68]:
```
# Non-Sarcastic descriptions wordcloud
plt.figure(figsize=(10,10), dpi=120)
WC = WordCloud(width=1200, height=1200, max_words=1000, min_font_size=5)
plt.imshow(WC.generate(other_string), interpolation='bilinear')
plt.axis("off")
plt.show()
```



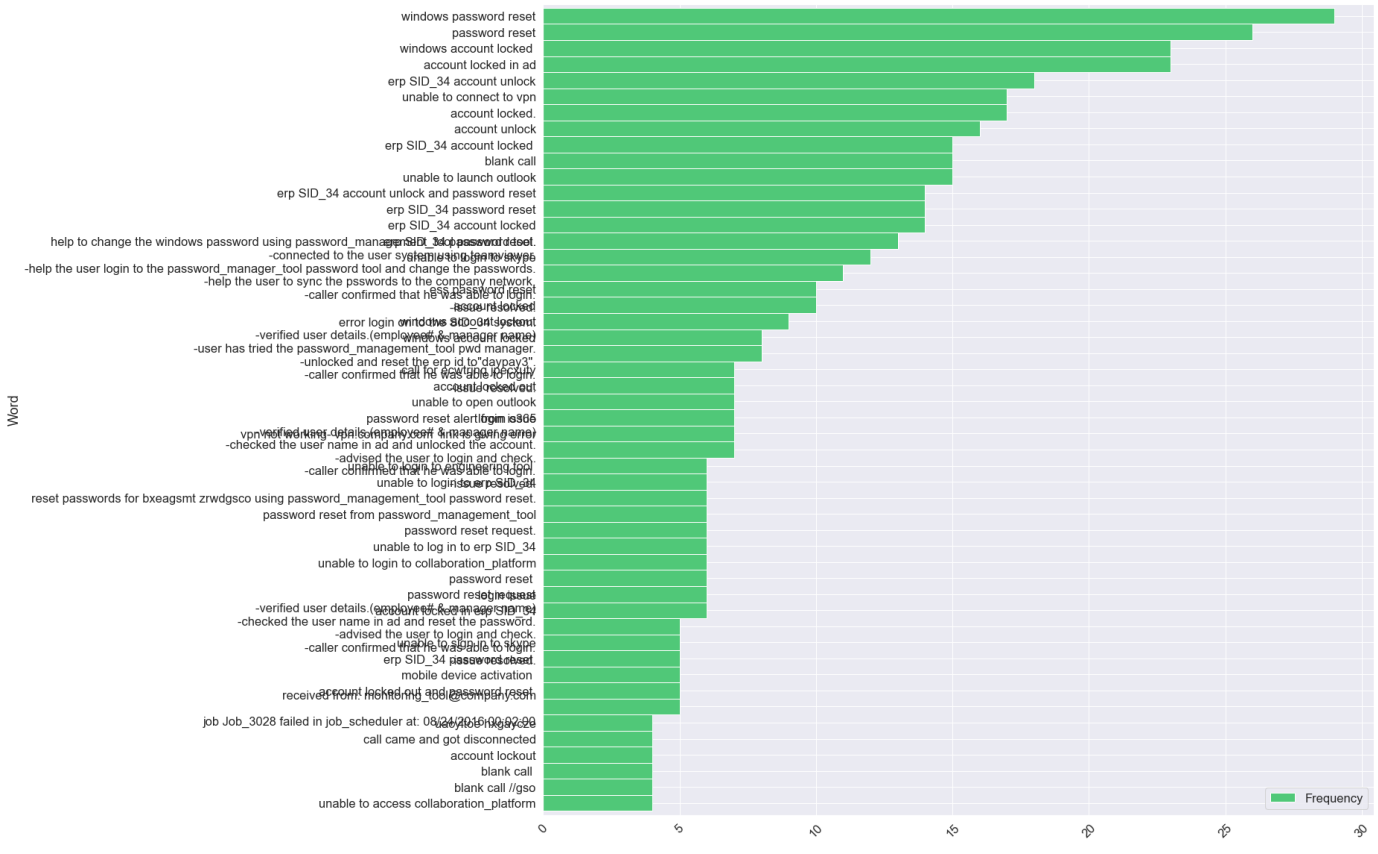
- Short Descriptions

In [69]:
```
top_N = 50
rslt = pd.DataFrame(Counter(dataset.short_description.tolist()).most_common(top_N),
                    columns=['Word', 'Frequency']).set_index('Word')
sns.set(font_scale=1.5)  # scale up font size
rslt.sort_values(by='Frequency', ascending=True).plot(kind='barh', width=1, figsize=(15, 15), c
plt.xticks(rotation=45)
plt.show()
```



- Descriptions

In [70]:
```
top_N = 50
rslt = pd.DataFrame(Counter(dataset.description.tolist()).most_common(top_N),
                    columns=['Word', 'Frequency']).set_index('Word')
sns.set(font_scale=1.5)  # scale up font size
rslt.sort_values(by='Frequency', ascending=True).plot(kind='barh', width=1, figsize=(20, 20), c
plt.xticks(rotation=45)
plt.show()
```



- **Description Lengths vs. Functional Group**

In [71]:
```
le = LabelEncoder()
dataset['group_code'] = le.fit_transform(dataset.group)
```

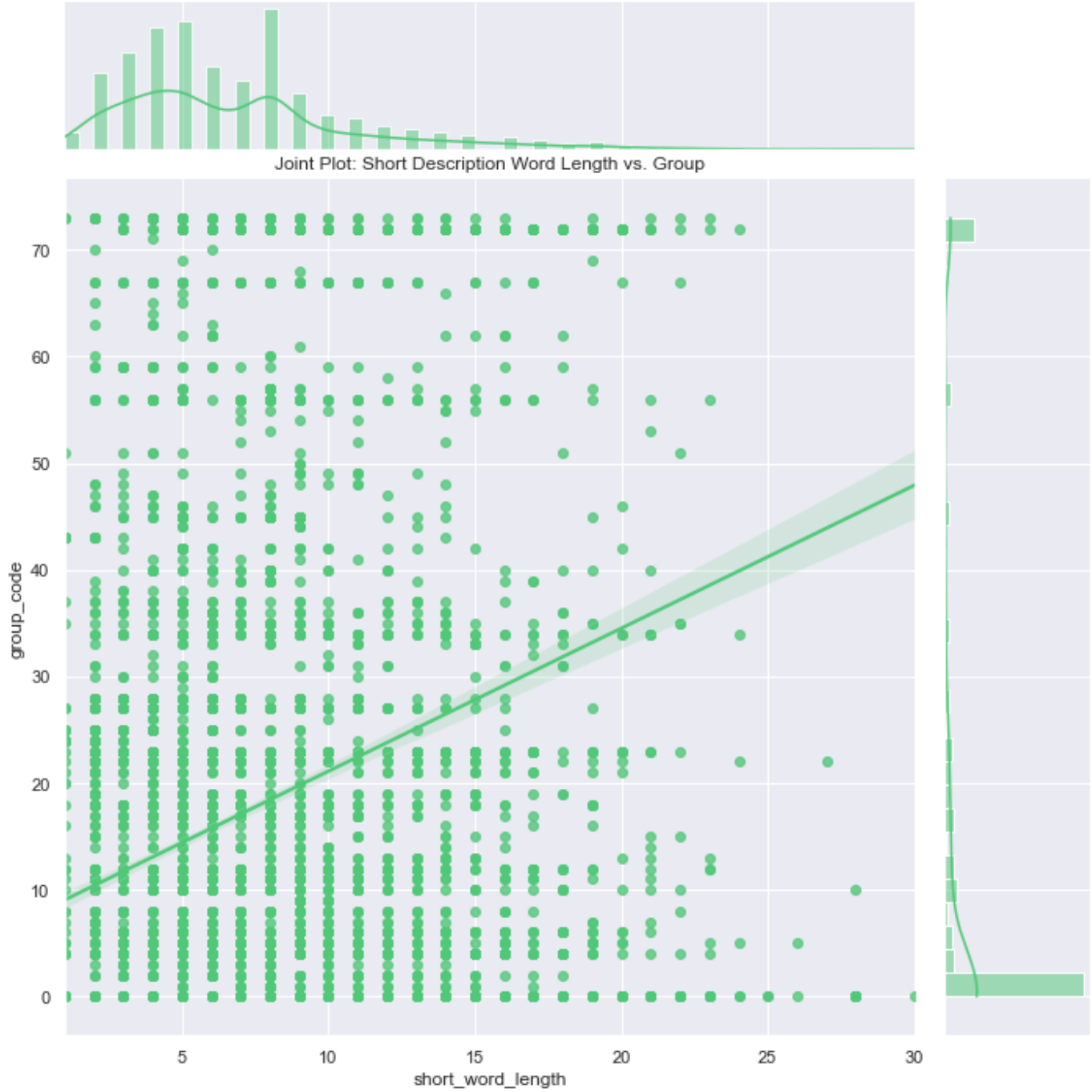
```
dataset.sample(7)
```

Out[71]:

	short_description	description	caller	group	char_length	word_length	short_d
7627	abended job in job_scheduler: Job_1989	received from: monitoring_tool@company.com\n\n...	ZkBogxib QsEJzdZO	GRP_6	106	11	
2163	configair server not available in production (...	configair server not available in production (...	iavozegx jpcudyfi	GRP_14	95	13	
4778	job Job_1148 failed in job_scheduler at: 09/11...	received from: monitoring_tool@company.com\n\n...	bpctwhsn kzqsbmtp	GRP_9	106	11	
5933	msc not communicating with erp	name:tmyeqika hfudpeot\nlanguage:\nbrowser:mic...	tmyeqika hfudpeot	GRP_18	174	15	
5	unable to log in to engineering tool and skype	unable to log in to engineering tool and skype	eflahbxn ltdgrvkz	GRP_0	46	9	
6992	probleme mit erpgui \tmqfjard qzhgdoua	probleme mit erpgui \tmqfjard qzhgdoua	tmqfjard qzhgdoua	GRP_24	38	5	
1490	lean tracker	\n\nreceived from: sdbcpvtx.hzpctsla@gmail.com...	sdbcpvtx hzpctsla	GRP_0	207	26	

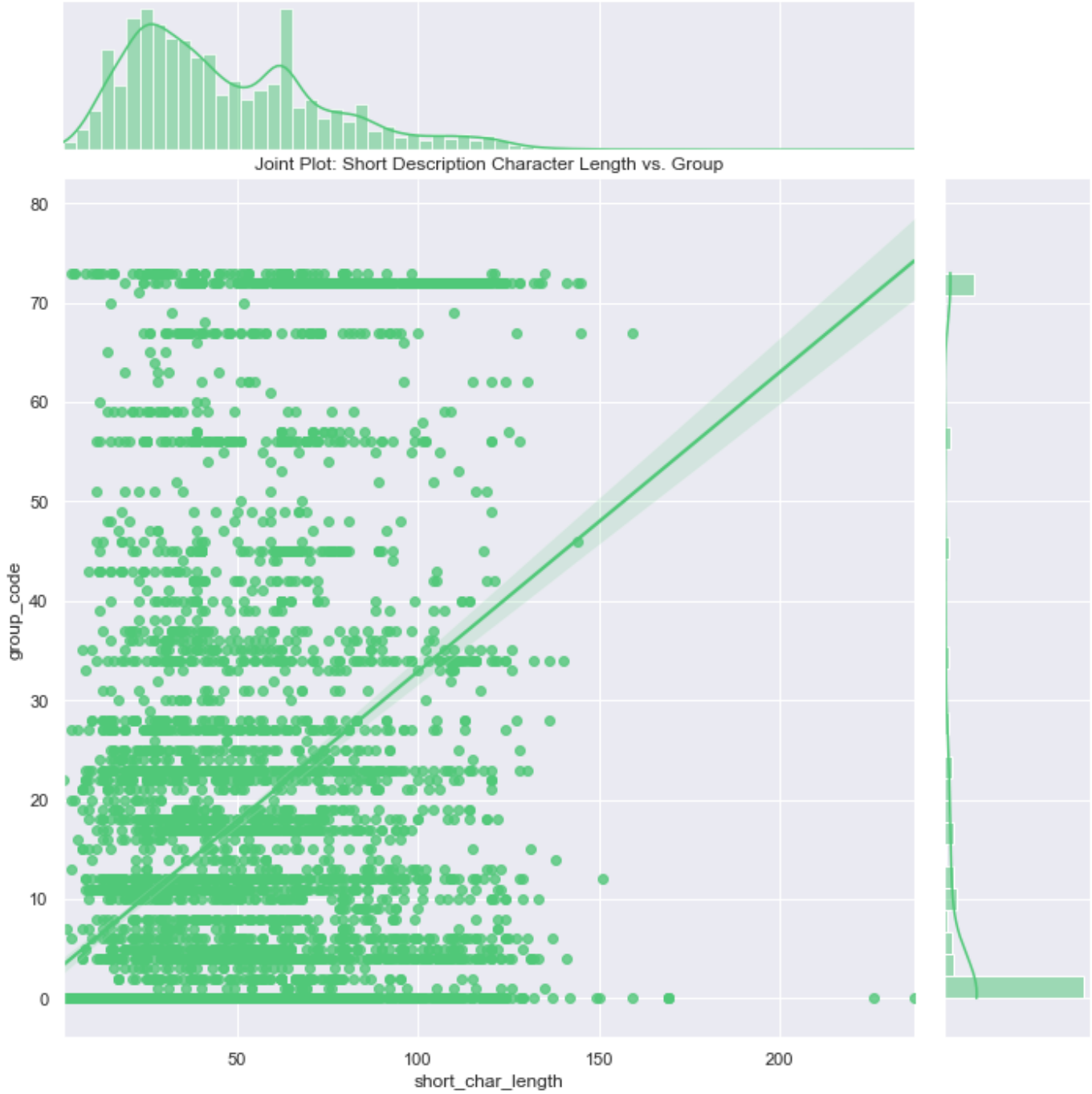
In [72]:

```
sns.set()
sns.jointplot(dataset.short_word_length, dataset.group_code,
              kind='reg', color='#50C878', height=10)
plt.title('Joint Plot: Short Description Word Length vs. Group')
plt.show()
```



In [73]:

```
sns.set()
sns.jointplot(dataset.short_char_length, dataset.group_code,
              kind='reg', color='#50C878', height=10)
plt.title('Joint Plot: Short Description Character Length vs. Group')
plt.show()
```



In [74]:

```
sns.set()
sns.jointplot(dataset.word_length, dataset.group_code,
              kind='reg', color='#50C878', height=10)
plt.title('Joint Plot: Description Word Length vs. Group')
plt.show()
```

In [75]:

```
sns.set()
sns.jointplot(dataset.char_length, dataset.group_code,
              kind='reg', color='#50C878', height=10)
plt.title('Joint Plot: Description Character Length vs. Group')
plt.show()
```

In [76]:
```
# binning the lengths

import jenkspy

NUM_BINS = 100
# calculates the natural breaks in a series, exploiting the Fisher-Jenks algorithm
# https://en.wikipedia.org/wiki/Jenks_natural_breaks_optimization
breaks = jenkspy.jenks_breaks(dataset['char_length'], nb_class=NUM_BINS)
labels = list(range(len(breaks) - 1))
pprint(breaks, compact=True)
```

```
[3.0, 17.0, 23.0, 28.0, 34.0, 40.0, 47.0, 55.0, 63.0, 71.0, 80.0, 90.0, 101.0,
 109.0, 117.0, 126.0, 137.0, 149.0, 160.0, 170.0, 180.0, 190.0, 201.0, 213.0,
 224.0, 235.0, 247.0, 259.0, 271.0, 283.0, 295.0, 308.0, 323.0, 338.0, 354.0,
 372.0, 392.0, 412.0, 431.0, 449.0, 468.0, 486.0, 508.0, 537.0, 568.0, 596.0,
 620.0, 653.0, 684.0, 720.0, 765.0, 801.0, 837.0, 866.0, 908.0, 934.0, 990.0,
 1026.0, 1063.0, 1116.0, 1157.0, 1176.0, 1197.0, 1234.0, 1265.0, 1301.0, 1347.0,
 1398.0, 1478.0, 1526.0, 1692.0, 1877.0, 1952.0, 2013.0, 2172.0, 2293.0, 2548.0,
 2744.0, 2833.0, 3062.0, 3141.0, 3249.0, 3403.0, 3628.0, 3734.0, 4169.0, 4286.0,
 4766.0, 5087.0, 6887.0, 7403.0, 7524.0, 8575.0, 8991.0, 9063.0, 9440.0, 9678.0,
 9912.0, 10077.0, 11968.0, 13001.0]
```

In [77]:
```
dataset['char_length_bins'] = pd.cut(dataset['char_length'] ,bins=breaks, labels=labels, includ
```

In [78]:
```
dataset
```

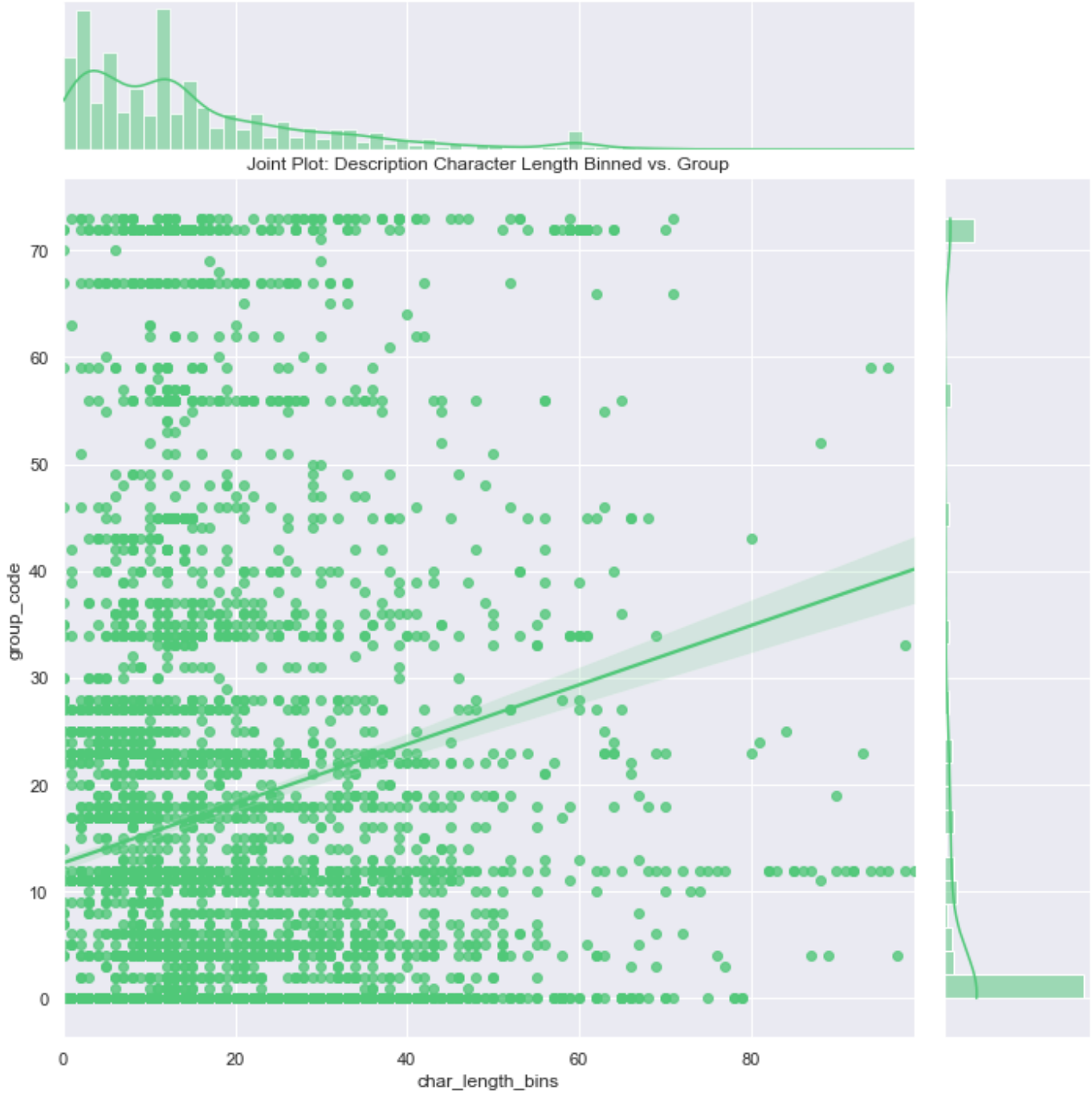
Out[78]:

	short_description	description	caller	group	char_length	word_length	short_cl
0	login issue	-verified user details.(employee# & manager na...	spxjnwir pjlcoqds	GRP_0	206	33	
1	outlook	\n\nreceived from: hmjdrvpb.komuaywn@gmail.com...	hmjdrvpb komuaywn	GRP_0	194	25	
2	cant log in to vpn	\n\nreceived from: eylqgodm.ybqkwiam@gmail.com...	eylqgodm ybqkwiam	GRP_0	87	11	
3	unable to access hr_tool page	unable to access hr_tool page	xbkucsvz gcpydteq	GRP_0	29	5	
4	skype error	skype error	owlgqjme qhcozdfx	GRP_0	12	2	
...	...	...	...	...	...	...	...
8495	emails not coming in from zz mail	\n\nreceived from: avglmrts.vhqmtiua@gmail.com...	avglmrts vhqmtiua	GRP_29	141	19	
8496	telephony_software issue	telephony_software issue	rbozivdq gmlhrtvp	GRP_0	24	2	
8497	vip2: windows password reset for tifpdchb pedx...	vip2: windows password reset for tifpdchb pedx...	oybwdsgx oxyhwrfz	GRP_0	50	7	
8498	machine não está funcionando	i am unable to access the machine utilities to...	ufawcgob aowhxjky	GRP_62	103	17	
8499	an mehreren pc`s lassen sich verschiedene prgr...	an mehreren pc`s lassen sich verschiedene prgr...	kqvbrspl jyzoklfx	GRP_49	82	11	

8499 rows × 12 columns

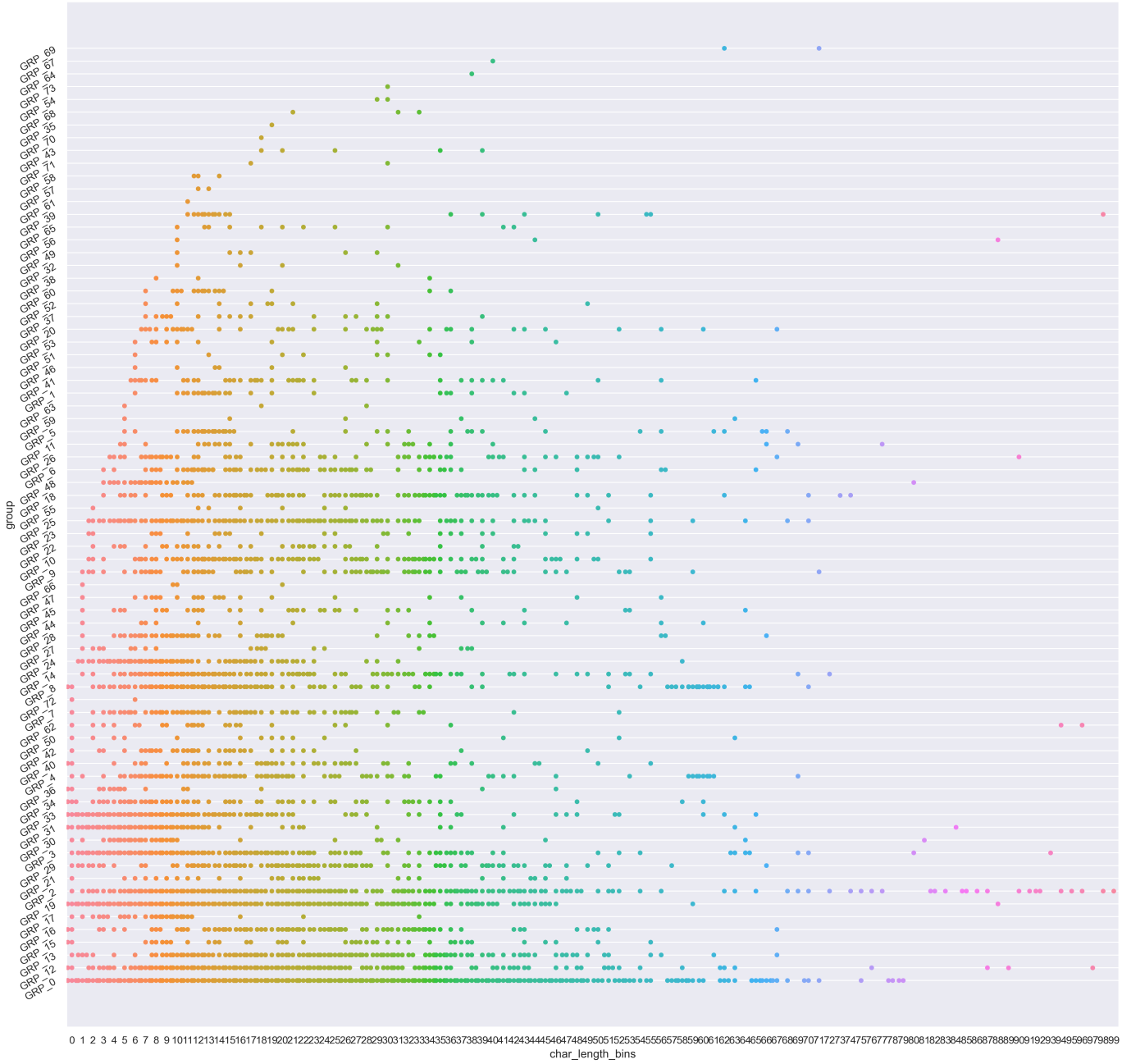
In [79]:

```
sns.set()
sns.jointplot(dataset.char_length_bins.astype(int), dataset.group_code,
              kind='reg', color='#50C878', height=10)
plt.title('Joint Plot: Description Character Length Binned vs. Group')
plt.show()
```



In [80]:

```
sns.set()
plt.figure(figsize=(20,20), dpi=300)
sns.swarmplot(x="char_length_bins", y="group", data=dataset)
plt.yticks(rotation=30)
plt.show()
```



In [81]:
```
# TODO: perform Chi2 Test with group to decide whether to include length features
import scipy.stats as stats
data_crosstab = pd.crosstab(dataset['caller'],
                            dataset['group'],
                            margins=True,
                            margins_name="Total")
data_crosstab
```

Out[81]:

group	GRP_0	GRP_1	GRP_10	GRP_11	GRP_12	GRP_13	GRP_14	GRP_15	GRP_16	GRP_17	...	GRP_68
caller												
DEsowkrHsciejzNE	0	0	0	0	0	1	0	0	0	0	...	0
HdOuWxAtqfKdlJMx	1	0	0	0	0	0	0	0	0	0	...	0
ZkBogxibQsEJzdZO	0	1	2	0	0	0	2	0	0	0	...	0
abiuclvjbrziktno	0	0	0	0	0	0	0	0	0	0	...	0
abtknpewifcekjdb	1	0	0	0	0	0	0	0	0	0	...	0
...	...	...	...	...	...	...	...	...	...	...	...	...

group	GRP_0	GRP_1	GRP_10	GRP_11	GRP_12	GRP_13	GRP_14	GRP_15	GRP_16	GRP_17	...	GRP_68
caller												
zylwdbigwdkbztjp	1	0	0	0	0	0	0	0	0	0	...	0
zymdwqsijzvbthil	1	0	0	0	0	0	0	0	0	0	...	0
zywoxerfpaqxtrfk	9	0	0	0	0	0	0	0	0	0	...	0
zyxjagrovjgozhpn	2	0	0	0	0	0	0	0	0	0	...	0
Total	3975	31	140	30	257	145	118	39	85	81	...	3

2951 rows × 75 columns

In [82]:
```
# significance level
alpha = 0.05

# Calcualtion of Chisquare test statistics
chi_square = 0
rows = dataset['caller'].unique()
columns = dataset ['group'].unique()
for i in columns:
    for j in rows:
        O = data_crosstab[i][j]
        E = data_crosstab[i]['Total'] * data_crosstab['Total'][j] / data_crosstab['Total']['Tot
        chi_square += (O-E)**2/E

# The p-value approach
print("Approach 1: The p-value approach to hypothesis testing in the decision rule")
p_value = 1 - stats.norm.cdf(chi_square, (len(rows)-1)*(len(columns)-1))
conclusion = "Failed to reject the null hypothesis."
if p_value <= alpha:
    conclusion = "Null Hypothesis is rejected."

print("chisquare-score is:", chi_square, " and p value is:", p_value)
print(conclusion)
```

```
Approach 1: The p-value approach to hypothesis testing in the decision rule
chisquare-score is: 268747.2241233948  and p value is: 0.0
Null Hypothesis is rejected.
```

- ## Language Detection

https://fasttext.cc/docs/en/language-identification.html

In [83]:
```
# TODO: try out fastText pre-trained language identification model (less latency) &
# Google CLD3 (Google Compact Language Detector v3)

# ! pip -q install fasttext
# import fasttext

# path_to_pretrained_model = './artifact/models/lid.176.bin'  # lid (Language identification mo
# fmodel = fasttext.load_model(path_to_pretrained_model)
# text = "+86 Hi there"
# fmodel.predict([text])
```

In [84]:
```
from langdetect import DetectorFactory, detect_langs
from langdetect.lang_detect_exception import LangDetectException

DetectorFactory.seed = seed
```

```
# detect the languages in the dataset
languages = []
errs = []
lang_samples = defaultdict(list)
for text in tqdm(dataset.description):
    try:
        lang = detect_langs(text)
        clean_lang = str(lang).split(':')[0][1:]
        lang_samples[clean_lang].append(text)
        languages.append(clean_lang)
    except LangDetectException as e:
        errs.append(text)
        print('text: ', text)
        print(e)
```

```
 74%|███████████████████████████████████████████████████████████████▉                | 6258/8499
[01:37<00:35, 63.05it/s]
text:  +86
No features in text.
100%|████████████████████████████████████████████████████████████████████████████████| 8499/8499
[02:11<00:00, 64.79it/s]
```

In [85]:
```
print("Unique languages in the descriptions: "
      f"{np.unique(languages)}")
```

```
Unique languages in the descriptions: ['af' 'ca' 'cs' 'cy' 'da' 'de' 'en' 'es' 'et' 'fi' 'fr'
'hr' 'hu' 'id'
 'it' 'ja' 'ko' 'lt' 'lv' 'nl' 'no' 'pl' 'pt' 'ro' 'sk' 'sl' 'so' 'sq'
 'sv' 'sw' 'tl' 'tr' 'vi' 'zh-cn']
```

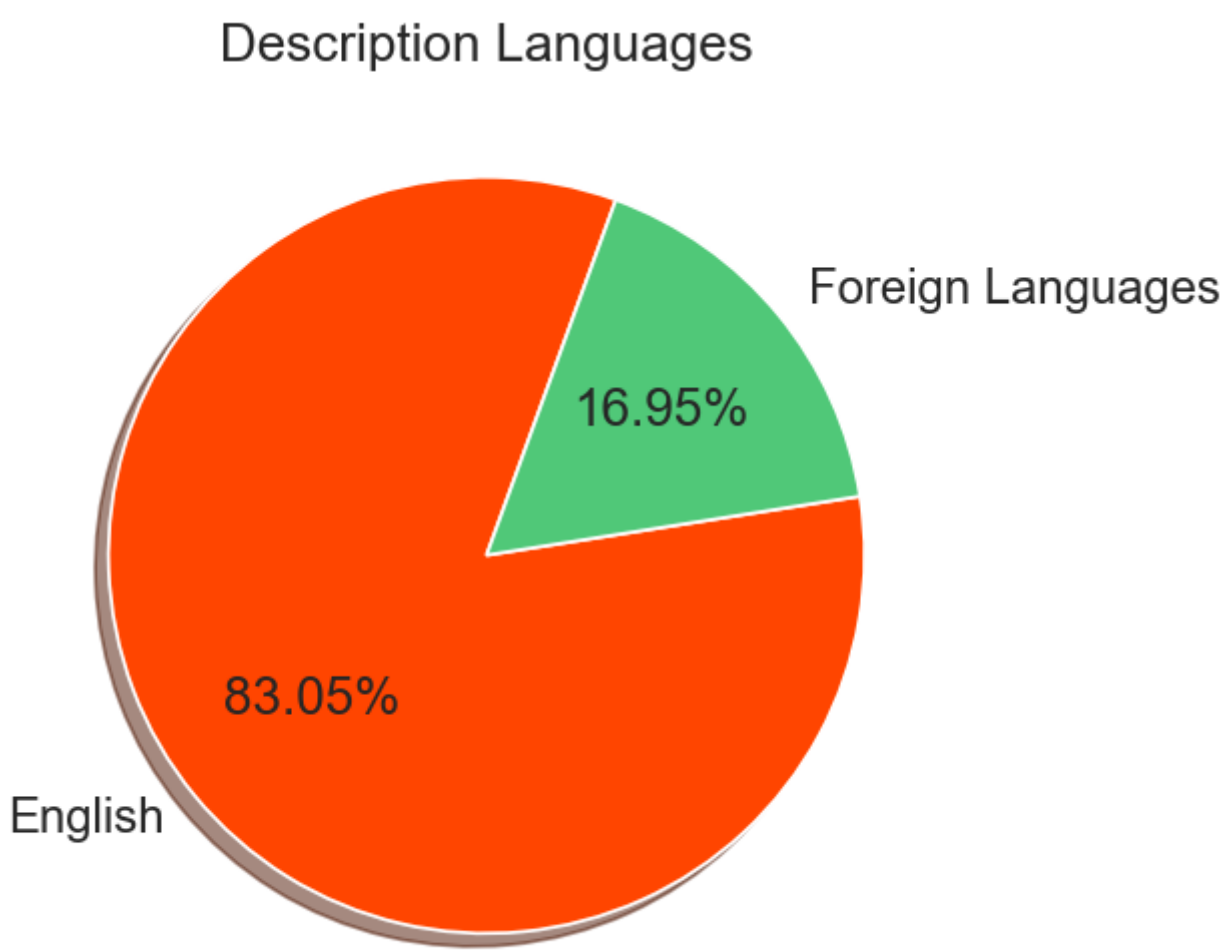
In [86]:
```
lang_freqs = {i: len(lang_samples[i]) for i in lang_samples}
freq_df = pd.DataFrame({'Language': lang_freqs.keys(), 'Frequency': lang_freqs.values()},
                       columns=['Language', 'Frequency']).set_index('Language')
freq_df.T
```

Out[86]:

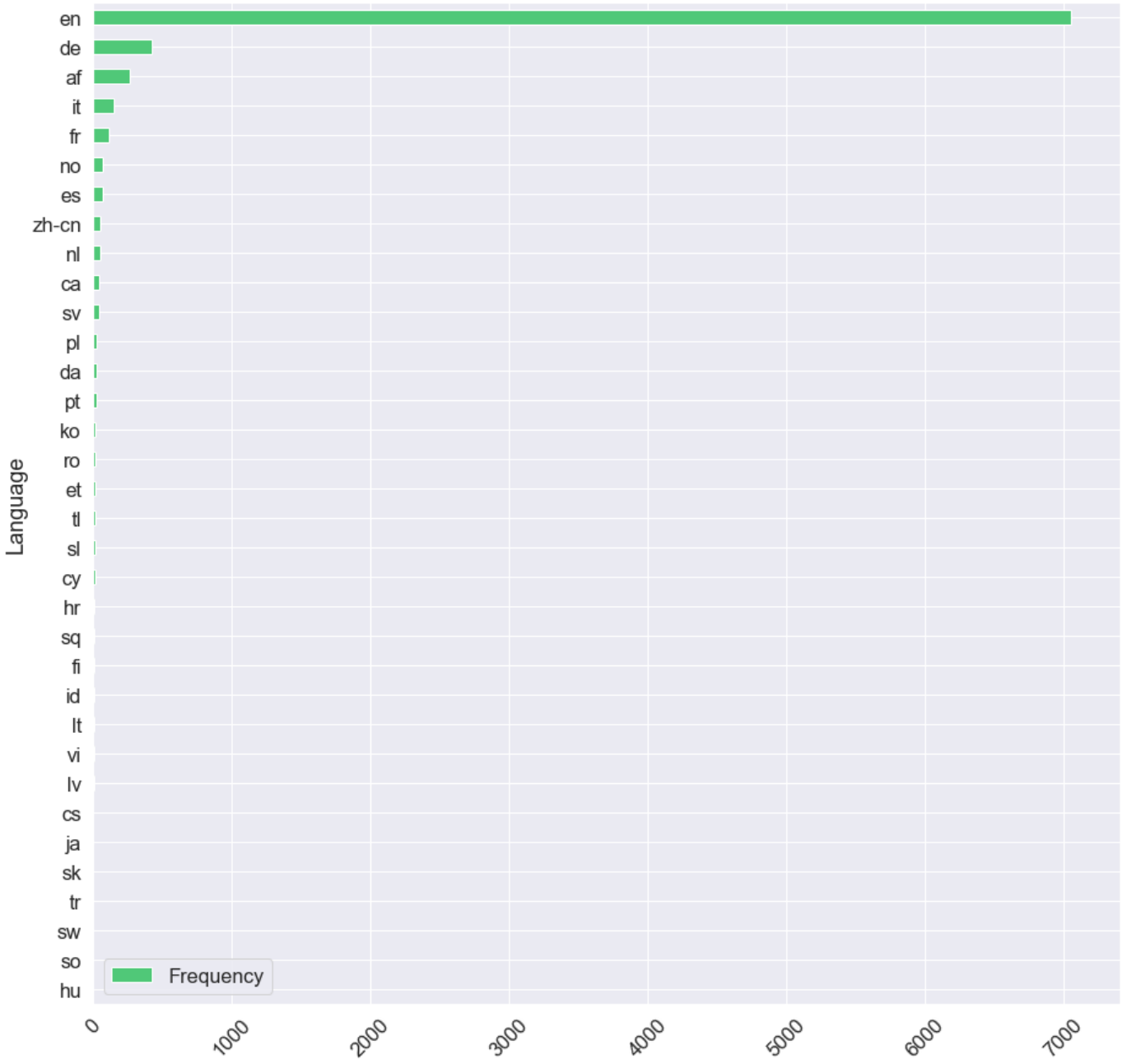
Language	en	no	es	it	af	sv	ca	nl	de	fr	...	lt	ja	sk	tr	sw	vi	so	lv	cs	hu
Frequency	7058	67	65	144	262	37	40	48	424	111	...	4	1	1	1	1	3	1	2	1	1

1 rows × 34 columns

In [87]:
```
sns.set(font_scale=1.25)  # scale up font size

plt.figure(figsize=(5, 5), dpi=125)
eng = freq_df.loc['en'].tolist()[0]
foreign_lang = len(dataset) - freq_df.loc['en'].tolist()[0]

plt.pie(x=[eng, foreign_lang],
        explode=(0, 0),
        labels=['English', 'Foreign Languages'],
        autopct='%1.2f%%',
        shadow=True,
        startangle=70,
        colors=['#FF4500', '#50C878'])

fig = plt.gcf()
fig.set_size_inches(5, 5)
plt.title('Description Languages')
plt.show()
```



In [88]:
```
sns.set(font_scale=1.5)  # scale up font size
freq_df.sort_values(by='Frequency', ascending=True).plot(kind='barh', width=0.5, figsize=(15, 1
plt.xticks(rotation=45)
plt.show()
```



In [89]:
```
# for i in lang_samples:
#     print(i)
#     try:
#         print(random.sample(lang_samples[i], 3))
#     except Exception:
#         print(random.sample(lang_samples[i], 1))
#     print('')
```

In [90]:
```
# detect the languages in the dataset
languages = []
lang_samples = defaultdict(list)
for text in tqdm(dataset.short_description):
    try:
        lang = detect_langs(text)
        clean_lang = str(lang).split(':')[0][1:]
        lang_samples[clean_lang].append(text)
        languages.append(clean_lang)
    except LangDetectException as e:
        errs.append(text)
        print('text: ', text)
        print(e)
```

```
 22%|████████████████▌                                                | 1838/8499
[00:34<01:53, 58.78it/s]
text:  bgflmyar.xgufkidq@gmail.com
No features in text.
 35%|████████████████████████▊                                        | 2976/8499
```

```
[00:55<02:08, 42.83it/s]
text:  ??????????????????
No features in text.
100%|██████████████████████████████████████████████████████████████████████| 8499/8499
[02:39<00:00, 53.45it/s]
```

In [91]:
```
print("Unique languages in the short descriptions: "
      f"{np.unique(languages)}")
```

```
Unique languages in the short descriptions: ['af' 'ca' 'cs' 'cy' 'da' 'de' 'en' 'es' 'et' 'fi'
'fr' 'hr' 'hu' 'id'
 'it' 'ko' 'lt' 'lv' 'nl' 'no' 'pl' 'pt' 'ro' 'sk' 'sl' 'so' 'sq' 'sv'
 'sw' 'tl' 'tr' 'vi' 'zh-cn']
```

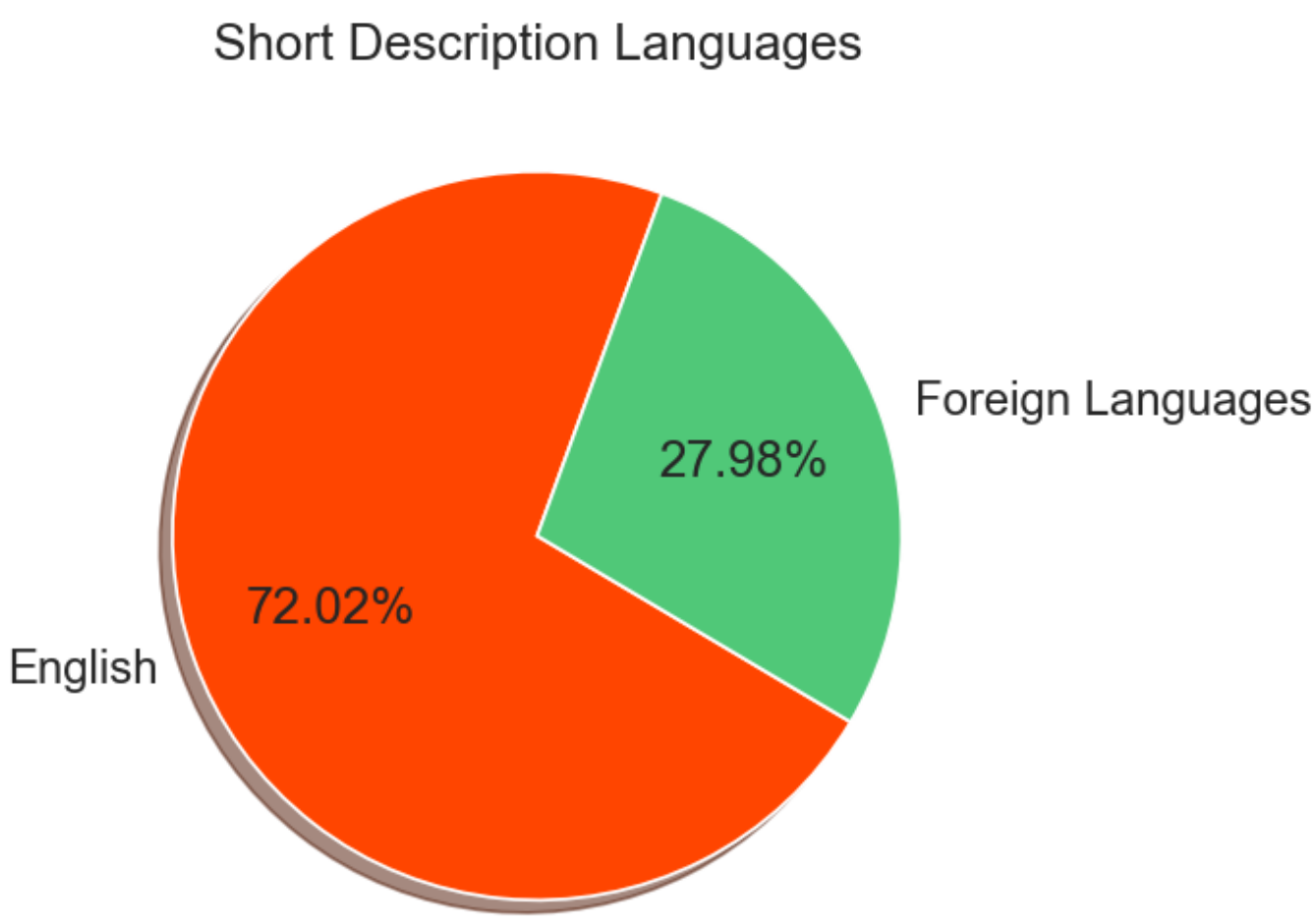
In [92]:
```
lang_freqs = {i: len(lang_samples[i]) for i in lang_samples}
freq_df = pd.DataFrame({'Language': lang_freqs.keys(), 'Frequency': lang_freqs.values()},
                       columns=['Language', 'Frequency']).set_index('Language')
freq_df.T
```

Out[92]:

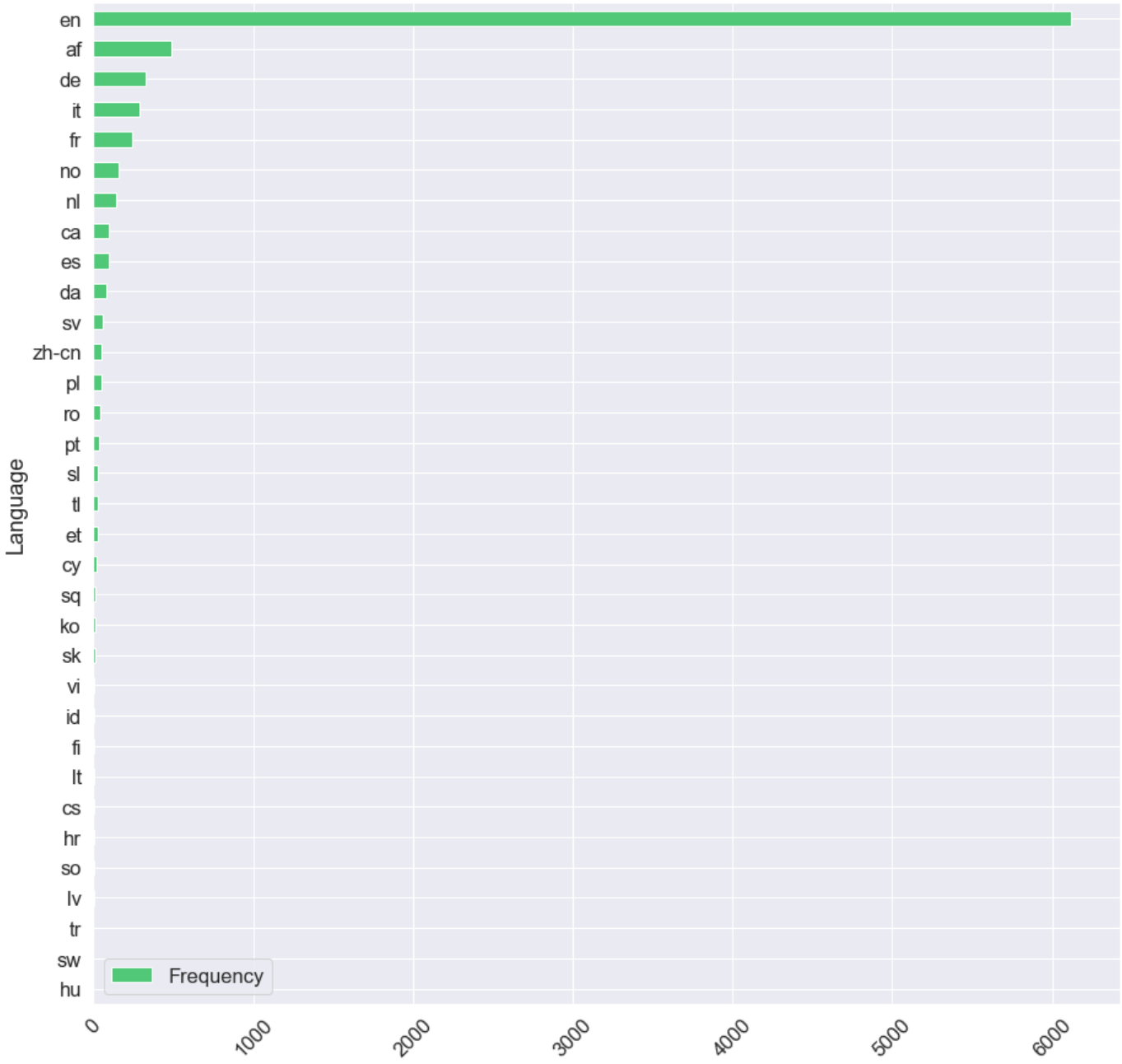
Language	en	et	no	es	it	nl	af	sv	ro	fr	...	vi	hr	pt	ko	lt	so	sw	lv	tr	hu
Frequency	6121	25	157	95	287	140	489	60	45	245	...	9	6	33	15	9	5	1	4	1	1

1 rows × 33 columns

In [93]:
```
sns.set(font_scale=1.25)  # scale up font size

plt.figure(figsize=(5, 5), dpi=125)
eng = freq_df.loc['en'].tolist()[0]
foreign_lang = len(dataset) - freq_df.loc['en'].tolist()[0]

plt.pie(x=[eng, foreign_lang],
        explode=(0, 0),
        labels=['English', 'Foreign Languages'],
        autopct='%1.2f%%',
        shadow=True,
        startangle=70,
        colors=['#FF4500', '#50C878'])

fig = plt.gcf()
fig.set_size_inches(5, 5)
plt.title('Short Description Languages')
plt.show()
```



In [94]:
```
sns.set(font_scale=1.5)  # scale up font size
freq_df.sort_values(by='Frequency', ascending=True).plot(kind='barh', width=0.5, figsize=(15, 1
plt.xticks(rotation=45)
plt.show()
```



In [95]:
```
# for i in lang_samples:
#     print(i)
#     try:
#         print(random.sample(lang_samples[i], 3))
#     except Exception:
#         print(random.sample(lang_samples[i], 1))
#     print('')
```

In [96]:
```
errs  # few errors where lang_detect failed, need to impute these irrelevant values
```

Out[96]:
```
['+86  ', 'bgflmyar.xgufkidq@gmail.com', '??????????????????']
```

In [97]:
```
dataset[dataset.description == errs[0]]
```

Out[97]:

	short_description	description	caller	group	char_length	word_length	short_char_length	short_word
6253	in the inbox always show there are several ema...	+86	mqbxwpfn uclrqfxa	GRP_0	5	1	94	

In [98]:
```
dataset.loc[dataset.description == errs[0], 'description'] = dataset[dataset.description == err
```

In [99]:
```
dataset[dataset.short_description == errs[1]]
```

Out[99]:

	short_description	description	caller	group	char_length	word_length	sho
1836	bgflmyar.xgufkidq@gmail.com	bgflmyar.xgufkidq@gmail.com wanted to check if...	olckhmvx pcqobjnd	GRP_0	83	13	

In [100…
```
dataset.loc[dataset.short_description == errs[1], 'short_description'] = dataset[dataset.short_
```

In [101…
```
dataset[dataset.short_description == errs[2]]
```

Out[101…

	short_description	description	caller	group	char_length	word_length	short_char_
2975	??????????????????	\n\nreceived from: yzbjhmpw.vzrulkog@gmail.com...	yzbjhmpw vzrulkog	GRP_0	1207	131	

In [102…
```
dataset.loc[dataset.short_description == errs[2], 'short_description'] = dataset[dataset.short_
```

In [103…
```
def clean_incosistencies():
    dataset.loc[dataset.description == errs[0], 'description'] = dataset[dataset.description ==
    dataset.loc[dataset.short_description == errs[1], 'short_description'] = dataset[dataset.sh
    dataset.loc[dataset.short_description == errs[2], 'short_description'] = dataset[dataset.sh

# clean_incosistencies()
```

- # Pre-Processing

In [104…
```
dataset
```

Out[104…

	short_description	description	caller	group	char_length	word_length	short_cl
0	login issue	-verified user details.(employee# & manager na...	spxjnwir pjlcoqds	GRP_0	206	33	
1	outlook	\n\nreceived from: hmjdrvpb.komuaywn@gmail.com...	hmjdrvpb komuaywn	GRP_0	194	25	
2	cant log in to vpn	\n\nreceived from: eylqgodm.ybqkwiam@gmail.com...	eylqgodm ybqkwiam	GRP_0	87	11	
3	unable to access hr_tool page	unable to access hr_tool page	xbkucsvz gcpydteq	GRP_0	29	5	
4	skype error	skype error	owlgqjme qhcozdfx	GRP_0	12	2	
...	...	...	...	...	...	...	...
8495	emails not coming in from zz mail	\n\nreceived from: avglmrts.vhqmtiua@gmail.com...	avglmrts vhqmtiua	GRP_29	141	19	
8496	telephony_software issue	telephony_software issue	rbozivdq gmlhrtvp	GRP_0	24	2	
8497	vip2: windows password reset for tifpdchb pedx...	vip2: windows password reset for tifpdchb pedx...	oybwdsgx oxyhwrfz	GRP_0	50	7	
8498	machine não está funcionando	i am unable to access the machine utilities to...	ufawcgob aowhxjky	GRP_62	103	17	
8499	an mehreren pc`s lassen sich verschiedene prgr...	an mehreren pc`s lassen sich verschiedene prgr...	kqvbrspl jyzoklfx	GRP_49	82	11	

8499 rows × 12 columns

Pipeline

- fix text encoding usng ftfy.fix_text
- handle other languages (drop or translate) (TODO)
- parse email messages to retain only subject and body
- clean up emails, links, website links, telephone numbers
- clean up anchor words like: 'Received from:', 'name:', 'hello', 'hello team' , 'cid' ...
- clean up security logs (TODO)
- clean up outage questionnaires (TODO)
- clean html tags if they exist
- clean /r /n characters
- strip caller names in descriptions
- translate accented characters (á -> a)
- convert unicode characters to ascii
- expand contractions (they're -> they are)
- clean stopwords & any custom stopwords
- clean up extra whitespaces & tokenize
- remove gibberish (after language translation)
- strip extra punctuation
- lemmatization/stemming (if needed for a model)

TODO: Language Translation

In [ ]:

TODO: Outage Questionnaires

In [105…
```
outage_df = dataset.reset_index().loc[pd.Series(['outage' in i for i in  dataset.description.to
```

In [106…
```
outage_df.group.value_counts()
```

Out[106…
```
GRP_8     139
GRP_4      16
GRP_0       8
GRP_16      1
Name: group, dtype: int64
```

In [107…
```
outage_df
```

Out[107…

	index	short_description	description	caller	group	char_length	word_length	short_char_length	s
78	78	power outage:UK al, 1st and 5th ave sites hard...	what type of outage: _____network _____ci...	mnlazfsr mtqrkhnx	GRP_8	1197	132	82	
79	79	power outage : germany, mx: site is hard down ...	what type of outage: ___x__network _____c...	jyoqwxhz clhxsoqy	GRP_8	1156	129	74	

	index	short_description	description	caller	group	char_length	word_length	short_char_length	s
189	189	network outage: india: site hard down since at...	what type of outage: __x___network ____c...	mnlazfsr mtqrkhnx	GRP_8	1163	130	86	
215	215	network outage - warehouse: node company-ups-...	what type of outage: __x___network ____c...	jyoqwxhz clhxsoqy	GRP_8	1152	128	80	
240	240	power outage :australia australia: site hard d...	what type of outage: __x___network ____c...	utyeofsk rdyzpwhi	GRP_8	1167	132	78	
...	...	...	...	...	...	...	...	...	
8097	8098	network outage, russia ru, company-russia-vpn-...	what type of outage: _x____network ____c...	uxgrdjfc kqxdjeov	GRP_8	1172	130	123	
8184	8185	circuit outage: vogelfontein, south africa mpl...	what type of outage: ____network ___x__c...	jyoqwxhz clhxsoqy	GRP_8	1158	130	86	
8272	8273	network outage : south amerirtca(argentina) si...	what type of outage: ___x_network ____c...	vbwszcqn nlbqsuyv	GRP_8	1148	129	88	
8274	8275	network outage: usa site is hard down since 05...	what type of outage: _x____network ____c...	vbwszcqn nlbqsuyv	GRP_8	1152	130	65	
8315	8316	network outage: sao pollaurido-mercedes benz p...	what type of outage: __x___network ____c...	dkmcfreg anwmfvlg	GRP_8	1188	130	113	

164 rows × 13 columns

In [108…
```
outage_df.group.value_counts()
```

Out[108…
```
GRP_8     139
GRP_4      16
GRP_0       8
GRP_16      1
Name: group, dtype: int64
```

In [109…
```
print(outage_df.description.tolist()[0])
```

```
what type of outage:  _____network     _____circuit     ___x__power (please specify what type o
f outage)

1. top 23 cert site   ?    ___yes_____     (yes/no/na)

2. when did it start  ?   ________4:31 pm et on 10/30. _____________

3.  scheduled maintenance ( power) ? __yno_____    (yes/no/na)     company power  _________
provider power _________

4.  scheduled maintenance ( network) ? ___no____    (yes/no/na)     company maint________ (yes/
no)       provider maint/ticket #___________________

5. does site have a backup circuit ?  __yes____    (yes/no/na)
```

```
6. backup circuit active ?   ___na_____     (yes/no/na)

7. site contact notified (phone/email) ?  _________     (yes/no/na)

8. remote dial-in ?   ____na_______   (yes/no/na)

9. equipment reset ?    ____na_______    (yes/no/na)

10. verified site working on backup circuit ?  ___na____     (yes/no/na)

11. vendor ticket  # ( global_telecom_1, verizon, telecom_vendor_1, telecom_vendor_2 )    ____g
lobal_telecom_1#000000223670658 _____________________

12. notified  gsc  _________     (yes/no/na)    cert started  ?_________    (yes/no/na)

13.  additional diagnostics
```

In [110…

```
outage_df
```

Out[110…

	index	short_description	description	caller	group	char_length	word_length	short_char_length	s
78	78	power outage:UK al, 1st and 5th ave sites hard...	what type of outage: ____network ____ci...	mnlazfsr mtqrkhnx	GRP_8	1197	132	82	
79	79	power outage : germany, mx: site is hard down ...	what type of outage: __x__network ____c...	jyoqwxhz clhxsoqy	GRP_8	1156	129	74	
189	189	network outage: india: site hard down since at...	what type of outage: __x___network ____c...	mnlazfsr mtqrkhnx	GRP_8	1163	130	86	
215	215	network outage - warehouse: node company-ups-...	what type of outage: __x___network ____c...	jyoqwxhz clhxsoqy	GRP_8	1152	128	80	
240	240	power outage :australia australia: site hard d...	what type of outage: __x___network ____c...	utyeofsk rdyzpwhi	GRP_8	1167	132	78	
...	...	...	...	...	...	...	...	...	...
8097	8098	network outage, russia ru, company-russia-vpn-...	what type of outage: _x___network ____c...	uxgrdjfc kqxdjeov	GRP_8	1172	130	123	
8184	8185	circuit outage: vogelfontein, south africa mpl...	what type of outage: ____network __x__c...	jyoqwxhz clhxsoqy	GRP_8	1158	130	86	
8272	8273	network outage : south amerirtca(argentina) si...	what type of outage: ___x_network ____c...	vbwszcqn nlbqsuyv	GRP_8	1148	129	88	
8274	8275	network outage: usa site is hard down since 05...	what type of outage: _x___network ____c...	vbwszcqn nlbqsuyv	GRP_8	1152	130	65	
8315	8316	network outage: sao pollauridо-mercedes benz p...	what type of outage: __x___network ____c...	dkmcfreg anwmfvlg	GRP_8	1188	130	113	

164 rows × 13 columns

- **TODO: Security/Event Logs**

In [111…
```
dataset.reset_index().loc[pd.Series(['source ip' in i for i in  dataset.description.tolist()])]
```

Out[111…

	index	short_description	description	caller	group	char_length	w
341	341	security incidents - (#in33071122) : [ipbl]: ...	source ip :\nsystem name :lmsl9516338\nuser n...	gzhapcld fdigznbk	GRP_3	166	
2977	2978	security incidents - (#in33987594) : 29866 vi...	source ip :\nsystem name :\nuser name:\nlocat...	gzhapcld fdigznbk	GRP_3	3249	
3096	3097	security incidents - (#in33976733) : suspicio...	source ip: 10.16.90.249\nsource hostname: andr...	gzhapcld fdigznbk	GRP_56	6887	
3097	3098	security incidents - (#in33984033) : internal...	source ip :\nsystem name :\nuser name:\nlocat...	gzhapcld fdigznbk	GRP_19	6868	
3529	3530	security incidents - (#in33944691) : possibl...	source ip: 195.272.28.222\nsource port: 80\nso...	gzhapcld fdigznbk	GRP_2	7524	
3531	3532	security incidents - (#in33944327) :possible ...	source ip :\nsystem name :\nuser name:\nlocat...	gzhapcld fdigznbk	GRP_2	3628	
3704	3705	security incidents - (#in33932723) : possibl...	source ip: 10.44.63.52\nsource hostname: leeng...	gzhapcld fdigznbk	GRP_48	3235	
3705	3706	security incidents - (#in33924718) : possibl...	source ip :195.22.28.222\ndestination ip: 12.9...	gzhapcld fdigznbk	GRP_2	4286	
3960	3961	security incidents - (#in33805815) : possible...	========================\nevent data\n=======...	gzhapcld fdigznbk	GRP_2	3734	
3964	3965	security incidents - (#in33809307) : possibl...	source ip :195.22.28.222 \nsystem name :androi...	gzhapcld fdigznbk	GRP_2	8988	
4086	4087	security incidents - (sw #in33895560) : mage...	source ip : 172.20.10.37 , 208.211.136.158\nsy...	ugyothfz ugrmkdhx	GRP_39	11968	
4088	4089	security incidents - (sw #in33895560) : mage...	source ip : 172.20.10.37 , 208.211.136.158\nsy...	ugyothfz ugrmkdhx	GRP_2	11968	
4729	4730	security incidents - (#in33847938) : possibl...	source ip :195.22.28.222\nsource port: 80\nsou...	gzhapcld fdigznbk	GRP_31	4169	
4824	4825	incident #in33541962 - phishing form submit -...	source ip: 10.38.93.30\nsource hostname: dane-...	ugyothfz ugrmkdhx	GRP_2	2494	
4885	4886	security incidents - (#in33826812) : possibl...	source ip :83.54.03.93209 \nsystem name :rgtw8...	gzhapcld fdigznbk	GRP_3	1838	

	index	short_description	description	caller	group	char_length	w
4892	4893	security incidents - (#in33826812) : possibl...	source ip :83.54.03.93209 \nsystem name :rgtw8...	gzhapcld fdigznbk	GRP_2	1837	
5091	5092	security incidents - (#in33578632) : suspicio...	source ip: 29.26.13.3095\nsource hostname: Hos...	gzhapcld fdigznbk	GRP_3	9063	
5432	5433	security incidents - (#in33765965) : possibl...	source ip :10.40.6.221\nsystem name :rqxl85172...	gzhapcld fdigznbk	GRP_2	8575	
5503	5504	incident #in33541962 - phishing form submit -...	we are seeing your 18.79.63.203/company-intern...	afkstcev utbnkyop	GRP_2	2293	
5505	5506	dsw in22457494	dsw in33568505\n\nwe are seeing your 172.20.10...	afkstcev utbnkyop	GRP_2	1495	
5506	5507	possible vulnerability scan from host.my-tss.c...	dsw in33568733\n\nwe are seeing your 208.211.1...	afkstcev utbnkyop	GRP_2	2833	
6063	6064	engineering_tool installation issue for distri...	detailed description of the problem including ...	rsgqbuln pevsanuf	GRP_0	491	
6733	6734	security incidents - (dsw incident no) : sus...	==========\nincident overview\n=...	gzhapcld fdigznbk	GRP_12	5084	
6749	6750	security incidents - (#in33669678) : possibl...	source ip: 93.115.241.50\nsource hostname: 93....	gzhapcld fdigznbk	GRP_2	1251	
6887	6888	security incidents - (#in33655554) : errata se...	==========\nincident overview\n...	gzhapcld fdigznbk	GRP_2	2744	
6930	6931	'51551 vid67965 microsoft windows httpsys rce ...	dsw in33568767\n\nincident overview\n=========...	afkstcev utbnkyop	GRP_12	2672	
6936	6937	[hw] filesystem near capacity - h: (HostName_894)	dsw in33644259\n\nrelated events: \nevent id: ...	afkstcev utbnkyop	GRP_39	968	
7080	7081	possible bash command injection attempt	dsw in33637966\n\nwe are seeing '50990 vid6315...	afkstcev utbnkyop	GRP_47	1018	
7150	7151	security incidents - (in33426117) : correlat...	related events: \n_____...	gzhapcld fdigznbk	GRP_2	866	
7153	7154	security incidents - (#in33417637) : repeat ...	source ip :10.16.143.221\ndestination ip: 31.1...	gzhapcld fdigznbk	GRP_69	1909	
7330	7331	security incidents - (#in33505432) : repeat ...	source ip :10.16.140.231\nsystem name :evhl811...	gzhapcld fdigznbk	GRP_2	4245	
7337	7338	security incidents - (#in33505432) : repeat ...	source ip :10.16.140.231\nsystem name :evhl811...	gzhapcld fdigznbk	GRP_2	4766	

	index	short_description	description	caller	group	char_length	w
7344	7345	security incidents - (sw #in33501789) : broa...	we are seeing activity indicating the host at ...	ugyothfz ugrmkdhx	GRP_2	13001	
7347	7348	HostName_480 - verify filesystem h:	dsw in31864001\n\nevent id: 67771149\nevent su...	afkstcev utbnkyop	GRP_39	778	
7351	7352	event summary: [hw] service icmp/icmp is down	dsw ticket in33426117\n\nevent id: 80657337\ne...	afkstcev utbnkyop	GRP_2	796	
7353	7354	event summary: [hw] service icmp/icmp is down	dsw ticket in33575214\n\nrelated events: \neve...	afkstcev utbnkyop	GRP_2	677	
7354	7355	event summary: [hw] service icmp/icmp is down	dsw ticket number in33575471\n\nrelated events...	afkstcev utbnkyop	GRP_2	684	
7355	7356	HostName_68 near capacity - 90%	dsw ticket in33575516\n\n related events: \nev...	afkstcev utbnkyop	GRP_39	968	
7646	7647	security incidents - (#in33578632) : suspicio...	source ip :\nsystem name :\nuser name:\nlocat...	gzhapcld fdigznbk	GRP_2	8991	
7981	7982	security incidents - (dsw #in33390850) : sus...	source ip : 78.83.16.293\nsystem name : HostNa...	ugyothfz ugrmkdhx	GRP_2	9881	
7983	7984	security incidents - (dsw #in33390850) : sus...	source ip : 78.83.16.293\nsystem name : HostNa...	ugyothfz ugrmkdhx	GRP_12	10077	
7986	7987	security incidents - (in33536629) : possible t...	source ip :10.44.94.214\ndest ip : 183.91.33.9...	gzhapcld fdigznbk	GRP_30	3403	
7988	7989	security incidents - (dsw #in33407676) : tra...	source ip : 61.01.52.02617\nsystem name : lpaw...	ugyothfz ugrmkdhx	GRP_2	9440	
7990	7991	as per inc1530161::security incidents - (in33...	\nfrom: gzhapcld fdigznbk \nsent: wednesday, a...	gzhapcld fdigznbk	GRP_2	5087	
7994	7995	security incidents - (dsw #in33407676) : tra...	source ip : 61.01.52.02617\nsystem name : lpaw...	ugyothfz ugrmkdhx	GRP_62	9440	
7995	7996	security incidents - (in33490582) : suspicio...	source ip : 29.26.13.3095\nsystem name :HostNa...	gzhapcld fdigznbk	GRP_12	7403	
7996	7997	security incidents - (sw #in33544563) : poss...	source ip : 45.25.35.0499\nsystem name : lpal9...	ugyothfz ugrmkdhx	GRP_2	9678	
8001	8002	security incidents - (sw #in33544563) : poss...	source ip : 45.25.35.0499\nsystem name : lpal9...	ugyothfz ugrmkdhx	GRP_62	9912	

In [112…
```
dataset.reset_index().loc[pd.Series(['source ip' in i for i in  dataset.description.tolist()])]
```

Out[112…
```
GRP_2     26
GRP_12     4
GRP_39     4
GRP_3      4
```

```
GRP_62     2
GRP_56     1
GRP_19     1
GRP_48     1
GRP_47     1
GRP_0      1
GRP_30     1
GRP_69     1
GRP_31     1
Name: group, dtype: int64
```

In [113…
```
dataset.reset_index().loc[pd.Series(['cyber' in i for i in  dataset.short_description.tolist()]
```

Out[113…

	index	short_description	description	caller	group	char_length	word_length	short_char_length	shor
491	491	october cyber security month - ransomware	october cyber security month - ransomware	pyrtfdxu nxfkqmoy	GRP_0	42	6	42	
1729	1729	cyber security - phish uacyltoe hxgaycze repor...	cyber security - phish uacyltoe hxgaycze repor...	ugyothfz ugrmkdhx	GRP_2	60	9	60	
5411	5412	cyber security - phish uacyltoe hxgaycze repor...	cyber security - phish uacyltoe hxgaycze repor...	ugyothfz ugrmkdhx	GRP_2	62	9	62	

TODO: dig deeper into security logs and cyber security issues handled by 2, 39, 12, ...

In [114…
```
import re
from collections import Counter

def remove_duplicates(text: str) -> str:
    '''Remove duplicates'''
    text = text.split(" ")
    for i in range(0, len(text)):
        text[i] = "".join(text[i])
    UniqW = Counter(text)
    text = " ".join(UniqW.keys())
    return str(text)

# remove_duplicates(text)
```

In [115…
```
log = dataset.reset_index().loc[pd.Series(['source ip' in i for i in  dataset.description.tolis
print(log)
```

```
source ip :
system name :
user  name:
location :
sep , sms status :
field sales user ( yes / no) :
dsw event log:
-----------------------------------------------------------------------------------------------
--------------------
event detail(s):

event_id 417013204:
[**] [1:21130977:10] 29866 vid22518 bare http get executable from ip address (possible download
er trojan) [**]
```

[classification: none] [priority: 2] [action: accept_passive] [impact_flag: 0] [impact: 0] [blo
cked: 2] [vlan: 0] [mpls label: 0] [pad2: 1]
[sensor id: 602982][event id: 281895][time: 2585087487.272206]
[xref => vid, 22518]
[src ip: 10.1.43.79][dst ip: 94.102.53.238][sport/itype: 61007][dport/icode: 80][proto: 6]
09/27/2016-11:39:36.272206 10.1.43.79:61007 -> 94.102.53.238:80
tcp ttl:127 tos:0x0 id:16297 iplen:20 dgmlen:119 df
ap seq: 0xfbc9dc8c ack: 0xe57c9433 win: 0x102 tcplen: 20
==pcap s==
=0c=00=00=00xz=eawn'=04=00w=00=00=00w=00=00=00e=00=00w?=a9@=00=7f=06=f23=0a=01+o^f5=ee=eeo=00p=
fb=c9=dc=8c=e5|=943p=18=01=02=d4=3d=00=00get /~yahoo/csrsv.exe http/1.1=0d=0ahost: 94.102.53.23
8=0d=0aconnection: keep-alive=0d=0a=0d=0a
==pcap e==

[ex http_uri 9: /~yahoo/csrsv.exe]

[ex http_hostname 10: 94.102.53.238]

[o:security]

[correlation_data]
sep 27 06:01:47 71.80.15.0714 dhcpd[12774]: dhcpack on 10.1.43.79 to c4:8e:8f:f6:4a:e5 (lhql851
6405) via eth1 relay 10.1.40.8 lease-duration 14400 (renew)

lowercaseurlcorrelation : /~yahoo/csrsv.exe
srcip : 10.1.43.79
urlcorrelation : /~yahoo/csrsv.exe
vendorreference : vid, 22518
foreseeconndirection : outgoing
refererproxycorrelationurl : null
foreseeexternalip : 94.102.53.238
eventtypeid : 200020003203113798
unique_event_hash : 946134710
ontologyid : 200020003203728796
foreseeinternalip : 10.1.43.79
urlpath : /~yahoo/csrsv.exe
srchostname : lhql8516405
inspectorruleid : 277082
inspectoreventid : 077564517
httpmethod : get
netacuity_destination_organization : ecatel ltd
vendoreventid : 281895
device_id : 2550522
foreseemaliciousprobability : 0.0846984
event_summary : 29866 vid22518 bare http get executable from ip address (possible downloader tr
ojan)
tcpflags : ***ap***
agentid : 102805
srchostname : lhql8516405
cvss : -1
foreseedstipgeo : den dolder,nld
devip : 10.32.100.17
inlineaction : 2
proto : tcp
dstport : 80
vendorpriority : 2
ileatdatacenter : true
vendorsigid : 29866
srcport : 61007
globalproxycorrelationurl : csrsv7
host : 94.102.53.238
dstip : 94.102.53.238
source_network_type : internal
url : 94.102.53.238/~yahoo/csrsv.exe
urlfullpath : /~yahoo/csrsv.exe
urlhost : 94.102.53.238
irreceivedtime : 1474976715927
action : not blocked
ctainstanceid : 0
vendorversion : 7
httpversion : http/1.1
logtimestamp : 2585087487
foreseemaliciouscomment : negativeevaluationthreshold:0.0181;positiveevaluationthreshold:1;mode

```
lversion:854922;classifiertype:naivebayes;annotatorlist:action-not blocked->0.7719~0.8136|event
typeid-200020003203113798->0.0005~0.0001|ontologyid-200020003203728796->0.0005~0.0001;evaluatio
nmodels->nb-global-model:0.9736:0.0181;
netacuity_destination_isp : ecatel ltd
device_network_type : internal
srcmacaddress : c4:8e:8f:f6:4a:e5
sherlockruleid : 690393
eventtypepriority : 3
```

In [116…

```
def clean_sec_logs(text: str) -> str:
    '''Clean up security logs'''
    if text.startswith('source ip'):
        words = set(['source','ip','hostname', 'mac', 'events', 'yes / no'])
        word_cleanup = r'\b(?:{})\b'.format('|'.join(words))
        text = text.replace('\n', ' ').replace('\r', '')
        text = re.sub(r'((:)?\s?\d+(.|:)?)+', '', text)
        text = re.sub('(_x000D_|_x_|_x|x_)', '', text)
        text = re.sub(r'(\[|\]|(\-)+|(\=)+|\%|\,|\"|\:|\(|\))?','', text)
        text = re.sub(word_cleanup,'', text)
        text = remove_duplicates(text)
    return str(text)

clean_sec_logs(log)
```

Out[116…

```
" system name user location sep sms status field sales dsw event log details event_id ** vidbar
e http get executable from address possible downloader trojan classification none priority acti
on accept_passive impact_flag impact blocked vlan mpls label pad sensor idevent idtime xref > v
id src ipdst ipsport/itypedport/icodeproto>tcp ttltosidiplendgmlendf ***ap*** seqfbcc ackewintc
plenpcap s zeawn'?afo^feeeeofbcdcedet /~yahoo/csrsv.exe http/hostconnection keepalive pcap e ex
http_uri http_hostname osecurity correlation_data sepdhcpd dhcpack onto cfelhql via ethrelaylea
sedurationrenew lowercaseurlcorrelation srcip urlcorrelation vendorreference vidforeseeconndire
ction outgoing refererproxycorrelationurl null foreseeexternalip eventtypeid unique_event_hash
ontologyid foreseeinternalip urlpath srchostname lhqlinspectorruleid inspectoreventid httpmetho
d netacuity_destination_organization ecatel ltd vendoreventid device_id foreseemaliciousprobabi
lity event_summary tcpflags agentid lhqlcvss foreseedstipgeo den doldernld devip inlineaction p
roto tcp dstport vendorpriority ileatdatacenter true vendorsigid srcport globalproxycorrelation
url csrsvhost dstip source_network_type internal url ~yahoo/csrsv.exe urlfullpath urlhost irrec
eivedtime not ctainstanceid vendorversion httpversion http/logtimestamp foreseemaliciouscomment
negativeevaluationthresholdpositiveevaluationthresholdmodelversionclassifiertypenaivebayes;anno
tatorlistactionnot blocked>eventtypeid>ontologyid>evaluationmodels>nbglobalmodel netacuity_dest
ination_isp device_network_type srcmacaddress cfesherlockruleid eventtypepriority"
```

• Parse Emails

In [117…

```
# !pip -q installmail-parser
import re
import email
import spacy
import mailparser
from utils.utils import clean_text, is_blank, is_not_blank
from utils.email_handler import email_regex
from charset_normalizer import from_bytes
from utils.email_handler import EmailHandler

nlp = spacy.load('en_core_web_sm', disable=['parser'])
```

- detect emails in description

In [118…

```
email_hdlr = EmailHandler()
emails_df = defaultdict(list)

for idx, row in tqdm(dataset.iterrows()):
    doc = nlp(row.description)
    toks = [t.text for t in doc]
    toks = [t.strip() for t in toks]
    text = " ".join(toks)
    tags = ['O']*len(doc)
    tag = "MAIL"
```

```
    email_indices = email_hdlr.match_ref(text, toks, tags,
                                         entity=tag,
                                         verbose=False)
    # if email is present add to accum
    if email_indices:
        row = dict(row)
        for i in row:
            emails_df[i].append(row[i])

emails_df = pd.DataFrame(dict(emails_df))
```

```
8499it [01:47, 78.97it/s]
```

In [119…

```
emails_df  # 2616 rows with an email id in description
```

Out[119…

	short_description	description	caller	group	char_length	word_length	short_
0	outlook	\n\nreceived from: hmjdrvpb.komuaywn@gmail.com...	hmjdrvpb komuaywn	GRP_0	194	25	
1	cant log in to vpn	\n\nreceived from: eylqgodm.ybqkwiam@gmail.com...	eylqgodm ybqkwiam	GRP_0	87	11	
2	unable to login to company vpn	\n\nreceived from: xyz@company.com\n\nhi,\n\ni...	chobktqj qdamxfuc	GRP_0	244	44	
3	vpn issue	\n\nreceived from: ugephfta.hrbqkvij@gmail.com...	ugephfta hrbqkvij	GRP_0	473	68	
4	vpn not working	\n\nreceived from: dceoufyz.saufqkmd@gmail.com...	dceoufyz saufqkmd	GRP_0	206	33	
...	...	...	...	...	...	...	...
2611	customer group enhanced field	\n\nreceived from: nlearzwi.ukdzstwi@gmail.com...	nlearzwi ukdzstwi	GRP_9	598	81	
2612	ess portal	\n\nreceived from: eagvusbr.nguqityl@gmail.com...	eagvusbr nguqityl	GRP_9	331	51	
2613	erp account unlock	name:mfeyouli ndobtzpw\nlanguage:\nbrowser:mic...	rbozivdq gmlhrtvp	GRP_0	197	19	
2614	vpn for laptop	\n\nreceived from: jxgobwrm.qkugdipo@gmail.com...	jxgobwrm qkugdipo	GRP_34	216	20	
2615	emails not coming in from zz mail	\n\nreceived from: avglmrts.vhqmtiua@gmail.com...	avglmrts vhqmtiua	GRP_29	141	19	

2616 rows × 12 columns

In [120…

```
def parse_body(msg: str, encoding='utf-8'):
    b = email.message_from_string(msg)
    body = ""

    if b.is_multipart():
        for part in b.walk():
            ctype = part.get_content_type()
            cdispo = str(part.get('Content-Disposition'))

            # skip any text/plain (txt) attachments
            if ctype == 'text/plain' and 'attachment' not in cdispo:
                body = part.get_payload(decode=True)  # decode
                break

    # not multipart - i.e. plain text & no attachments
    else:
        # print('Plain Text')
        body = b.get_payload(decode=True)
```

```
    subject_matches =  re.findall('subject:.*\n', test)
    if subject_matches:
        subject = subject_matches[0]
    else:
        subject = b.get_all('subject')

    if not isinstance(subject, str):
        subject = ''

    # print("Subject: ", subject)
    # print("Body: ", body)

    body = str(from_bytes(body).best())
    # body = str(body, encoding)

    parsed_msg = str(subject + body)
    if is_blank(parsed_msg):
        # return original message if no payload is found after parsing
        return msg
    return parsed_msg
```

In [121…

```
msgs = list(set(emails_df.description.tolist()))
msgs = [i for i in msgs if 'to:' in i]
len(msgs)
```

Out[121…

```
303
```

In [122…

```
test = msgs[20]

print(test)
print('\nParsed:')
print(parse_body(test))
```

```
received from: oetlgbfw.bsctrnwp@gmail.com

i am not able to login my account
i must have the wrong password

matghyuthdw is my username

dan,
welcome to s&op .. i've created your user account; please use following logn credentials:



dthyan matheywtyuews
sales manager gl2
oetlgbfw.bsctrnwp@gmail.com<mailto:oetlgbfw.bsctrnwp@gmail.com>
t


Parsed:

received from: oetlgbfw.bsctrnwp@gmail.com

i am not able to login my account
i must have the wrong password

matghyuthdw is my username

dan,
welcome to s&op .. i've created your user account; please use following logn credentials:



dthyan matheywtyuews
sales manager gl2
oetlgbfw.bsctrnwp@gmail.com<mailto:oetlgbfw.bsctrnwp@gmail.com>
```

t

In [123…
```
def strip_headers(text: str) -> str:
    '''strip headers in email messages like: "received from: xyz@gmail.com"'''
    patterns = list()
    patterns.append(r'received from:?\s?' + email_regex)
    patterns.append(r'from:?\s?' + email_regex)
    patterns.append(r'email:?\s?' + email_regex)
    patterns.append(r'to:?\s?' + email_regex)
    patterns.append(r'to:\s')
    patterns.append(r'sent:\s.*[ap]m')
    patterns.append(r'customer number:')
    patterns.append(r'summary:')
    patterns.append(r'subject:')
    patterns.append(r'telephone:')
    patterns.append(r'regional controller')
    patterns.append(r'<mail>')
    patterns.append(r'help desk,')
    patterns.append(r'it team,')
    patterns.append(r'global it team,')
    patterns.append(r'\nbest\n')
    patterns.append(r'\nregards\n')
    patterns.append(r'\[cid:?.*\]')  # attachments link

    # patterns.append(r'subject:?.*\b')
    patterns = [re.compile(p) for p in patterns]
    for pattern in patterns:
        text = re.sub(pattern, '', text.strip()).strip()
    return str(text.strip())

print(strip_headers(test))
```

```
i am not able to login my account
i must have the wrong password

matghyuthdw is my username

dan,
welcome to s&op .. i've created your user account; please use following logn credentials:



dthyan matheywtyuews
sales manager gl2
oetlgbfw.bsctrnwp@gmail.com
t
```

In [124…
```
def parse_email(msg: str) -> str:
    '''parses and cleans email messages'''
    doc = nlp(msg)
    toks = [t.text for t in doc]
    toks = [t.strip() for t in toks]
    text = " ".join(toks)
    tags = ['O']*len(doc)
    tag = "MAIL"
    email_indices = email_hdlr.match_ref(text, toks, tags,
                                         entity=tag,
                                         verbose=False)
    if email_indices:
        msg = parse_body(msg)
        msg = strip_headers(msg)
    return msg

def parse_email_row(row):
    descr = row.description
    short_descr = row.short_description
    # text_normalized = str(from_bytes(text.encode('utf-8')).best())
    row['cleaned_short_description'] = parse_email(short_descr)
    row['cleaned_description'] = parse_email(descr)
```

```
    return row

print(parse_email(test))
```

```
i am not able to login my account
i must have the wrong password

matghyuthdw is my username

dan,
welcome to s&op .. i've created your user account; please use following logn credentials:



dthyan matheywtyuews
sales manager gl2
oetlgbfw.bsctrnwp@gmail.com
t
```

In [125…

```
emails_df = emails_df.progress_apply(parse_email_row, axis=1)
emails_df.to_csv('./data/cleaned_emails_test.csv', index=None)
```

```
100%|██████████████████████████████████████████████████████████████████████| 2616/2616
[01:31<00:00, 28.65it/s]
```

In [126…

```
emails_df
```

Out[126…

	short_description	description	caller	group	char_length	word_length	short_
0	outlook	\n\nreceived from: hmjdrvpb.komuaywn@gmail.com...	hmjdrvpb komuaywn	GRP_0	194	25	
1	cant log in to vpn	\n\nreceived from: eylqgodm.ybqkwiam@gmail.com...	eylqgodm ybqkwiam	GRP_0	87	11	
2	unable to login to company vpn	\n\nreceived from: xyz@company.com\n\nhi,\n\ni...	chobktqj qdamxfuc	GRP_0	244	44	
3	vpn issue	\n\nreceived from: ugephfta.hrbqkvij@gmail.com...	ugephfta hrbqkvij	GRP_0	473	68	
4	vpn not working	\n\nreceived from: dceoufyz.saufqkmd@gmail.com...	dceoufyz saufqkmd	GRP_0	206	33	
...	...	...	...	...	...	...	...
2611	customer group enhanced field	\n\nreceived from: nlearzwi.ukdzstwi@gmail.com...	nlearzwi ukdzstwi	GRP_9	598	81	
2612	ess portal	\n\nreceived from: eagvusbr.nguqityl@gmail.com...	eagvusbr nguqityl	GRP_9	331	51	
2613	erp account unlock	name:mfeyouli ndobtzpw\nlanguage:\nbrowser:mic...	rbozivdq gmlhrtvp	GRP_0	197	19	
2614	vpn for laptop	\n\nreceived from: jxgobwrm.qkugdipo@gmail.com...	jxgobwrm qkugdipo	GRP_34	216	20	
2615	emails not coming in from zz mail	\n\nreceived from: avglmrts.vhqmtiua@gmail.com...	avglmrts vhqmtiua	GRP_29	141	19	

2616 rows × 14 columns

In [127…

```
for msg in random.sample(msgs, 10):
    print(msg)
    print("Cleaned:")
    print(parse_email(msg))
    print('~'*200)
```

```
received from: afwzehqs.jfbxegac@gmail.com

hello team

we are unable to send any quotations to customer or sales through zz_mails.

please check and resolve immediately.

example: 3116105044 was sent to afwzehqs.jfbxegac@gmail.com<mailto:afwzehqs.jfbxegac@gmail.com>
but not received.

warm
Cleaned:
hello team

we are unable to send any quotations to customer or sales through zz_mails.

please check and resolve immediately.

example: 3116105044 was sent  but not received.

warm
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
from: tsbnfixp numwqahj
sent: wednesday, september 28, 2016 9:01 pm
to: nwfodmhc exurcwkm
subject: amar wg: die synchronisierung mit exchange activesync ist auf ihrem gerät vorübergehen
d blockiert, bis der zugriff vom administrator gewährt wird.
importance: high

please usa access for that company owned device.
user-id: grbhybrdg
tsbnfixp numwqahj

thanking you in anticipation!

oqlcdvwi pulcqkzo


von: microsoft outlook
gesendet: mittwoch, 28. september 2016 17:29
an: tsbnfixp numwqahj <tsbnfixp.numwqahj@gmail.com>
betreff: die synchronisierung mit exchange activesync ist auf ihrem gerät vorübergehend blockie
rt, bis der zugriff vom administrator gewährt wird.

der zugriff von ihrem mobilen gerät auf inhalte über exchange activesync ist vorrübergehend blo
ckiert, da es in quarantäne gestellt ist. sie müssen keine aktion durchführen. der inhalt wird
automatisch heruntergeladen, sobald der zugriff vom administrator gewährt wird.

please ignore the above paragraph. we cannot change it or delete it.

special note 30-jan-2015: the microsoft outlook app for ios and android released yesterday is n
ot currently approved software for accessing company e-mail. until it is uacyltoe hxgayczeed an
d approved, please consider using one of the other (1) the embedded e-mail software in your mob
ile device (2) the browser on your mobile device or (3) the microsoft owa app published for you
r mobile device platform.

beginning 01-mar-2012 employees, with supervisor approval, may use personally owned mobile devi
ces to access outlook email. company is moving forward and providing the opportstorage_product
for our employees to use specified personally owned devices to allow for productivity improveme
nt and enable work-life balance. this is an addition to the policy for company owned devices.

currently approved handheld devices can be found in this policy:
```

wireless mobility technical document

the above policy will be updated as other devices are approved for use.

if you own an approved device and would like to take advantage of this opportstorage_product yo
u can submit a ticketing_tool ticket to the it global support center (gsc). if it is a personal
ly owned device, you need to attach the agreement form found in the wireless mobility standard
procedure. this agreement must be signed by you and your next level supervisor and provided to
the gsc prior to a ticket being entered. you can attach the signed form to the ticket or send t
he signed form to the gsc and they will attach it.

any ticket without the signed form will be cancelled. you have 2 weeks to process and submit th
e form before your device will be denied (deleted from quarantine).

wireless mobility standard procedure
informationen zu ihrem mobilen gerät:
gerätemodell: iphone6c2
gerätetyp: iphone
geräte-id: fvqfj874r56gj3r4jb39kdgu3s
gerätebetriebssystem: ios 10.0.sartlgeo lhqksbdx4a403
gerätebenutzer-agent: apple-iphone6c2/1401.403
geräte-imei:
exchange activesync-version: 16.0
gerätezugriffsstatus: quarantined
grund für gerätezugriffsstatus: global
um 28.09.2016 15:28:41 an tsbnfixp.numwqahj@gmail.com gesendet.

Cleaned:
please usa access for that company owned device.
user-id: grbhybrdg
tsbnfixp numwqahj

thanking you in anticipation!

oqlcdvwi pulcqkzo

von: microsoft outlook
gesendet: mittwoch, 28. september 2016 17:29
an: tsbnfixp numwqahj <tsbnfixp.numwqahj@gmail.com>
betreff: die synchronisierung mit exchange activesync ist auf ihrem gerõt vor³bergehend blockie
rt, bis der zugriff vom administrator gewõhrt wird.

der zugriff von ihrem mobilen gerõt auf inhalte ³ber exchange activesync ist vorr³bergehend blo
ckiert, da es in quarantõne gestellt ist. sie m³ssen keine aktion durchf³hren. der inhalt wird
automatisch heruntergeladen, sobald der zugriff vom administrator gewõhrt wird.

please ignore the above paragraph. we cannot change it or delete it.

special note 30-jan-2015: the microsoft outlook app for ios and android released yesterday is n
ot currently approved software for accessing company e-mail. until it is uacyltoe hxgayczeed an
d approved, please consider using one of the other (1) the embedded e-mail software in your mob
ile device (2) the browser on your mobile device or (3) the microsoft owa app published for you
r mobile device platform.

beginning 01-mar-2012 employees, with supervisor approval, may use personally owned mobile devi
ces to access outlook email. company is moving forward and providing the opportstorage_product
for our employees to use specified personally owned devices to allow for productivity improveme
nt and enable work-life balance. this is an addition to the policy for company owned devices.

currently approved handheld devices can be found in this policy:

wireless mobility technical document

the above policy will be updated as other devices are approved for use.

if you own an approved device and would like to take advantage of this opportstorage_product yo
u can submit a ticketing_tool ticket to the it global support center (gsc). if it is a personal
ly owned device, you need to attach the agreement form found in the wireless mobility standard
procedure. this agreement must be signed by you and your next level supervisor and provided to
the gsc prior to a ticket being entered. you can attach the signed form to the ticket or send t
he signed form to the gsc and they will attach it.

any ticket without the signed form will be cancelled. you have 2 weeks to process and submit th

e form before your device will be denied (deleted from quarantine).

wireless mobility standard procedure
informationen zu ihrem mobilen gerõt:
gerõtemodell: iphone6c2
gerõtetyp: iphone
gerõte-id: fvqfj874r56gj3r4jb39kdgu3s
gerõtebetriebssystem: ios 10.0.sartlgeo lhqksbdx4a403
gerõtebenutzer-agent: apple-iphone6c2/1401.403
gerõte-imei:
exchange activesync-version: 16.0
gerõtezugriffsstatus: quarantined
grund f³r gerõtezugriffsstatus: global
um 28.09.2016 15:28:41 an tsbnfixp.numwqahj@gmail.com gesendet.
~~~~~~

received from: bqapjkcl.ljeakcqf@gmail.com

please reset my password in erp-engineering tool.

dipl.-ing.(ba) bqapjkcl ljeakcqf
bqapjkcl.ljeakcqf@gmail.com<mailto:bqapjkcl.ljeakcqf@gmail.com>

company shared services gmbh
managing directors/geschäftsführer: phvkowml azbtkqwx, naruedlk mpvhakdq

Cleaned:
please reset my password in erp-engineering tool.

dipl.-ing.(ba) bqapjkcl ljeakcqf
bqapjkcl.ljeakcqf@gmail.com

company shared services gmbh
managing directors/geschõftsf³hrer: phvkowml azbtkqwx, naruedlk mpvhakdq
~~~~~~

received from: dpuifqeo.eglwsfkn@gmail.com

getting this error when trying to print labels using zebra printers in shipping

[cid:image001.jpg@01d222cc.865bed90]

from: thoyhts brthyrtiv
sent: monday, october 10, 2016 8:02 am
to: dpuifqeo eglwsfkn <dpuifqeo.eglwsfkn@gmail.com>
subject: re: imp

yes because that is a erp error message.

Cleaned:
getting this error when trying to print labels using zebra printers in shipping

from: thoyhts brthyrtiv

dpuifqeo eglwsfkn <dpuifqeo.eglwsfkn@gmail.com>
re: imp

yes because that is a erp error message.
~~~~~~
from: oinqckds qieswrfu
sent: thursday, september 29, 2016 5:30 pm
~~~~~~

to: nwfodmhc exurcwkm
cc: chrthryui stavenheim
subject: trurthyuft aw: tess account

hello, can someone check whether chrthryui stavenheim can login with his account ccftv15
via the citrix access
with this access.
please inform him and myself when the issue is fixed as we have to generate turnover and satisf
y customers

oinqckds qieswrfu
manager tess holemaking design automation
oinqckds.qieswrfu@gmail.com

von: chrthryui stavenheim [mailto:chrthryui.stavenheim@bank.se]
gesendet: donnerstag, 29. september 2016 08:12
an: oinqckds qieswrfu
betreff: tess account

hello robhyertyj

i've been unable to login to tess a few weeks now. either password och user name is wrong. user
name is ccftv15
could you please help me out?

med vänlig hälsning / best
Cleaned:
hello, can someone check whether chrthryui stavenheim can login with his account ccftv15
via the citrix access
with this access.
please inform him and myself when the issue is fixed as we have to generate turnover and satisf
y customers

oinqckds qieswrfu
manager tess holemaking design automation
oinqckds.qieswrfu@gmail.com

von: chrthryui stavenheim [mail]
gesendet: donnerstag, 29. september 2016 08:12
an: oinqckds qieswrfu
betreff: tess account

hello robhyertyj

i've been unable to login to tess a few weeks now. either password och user name is wrong. user
name is ccftv15
could you please help me out?

med vänlig hälsning / best
~~~~~~
~~~~~~
~~~~~~~~~~

received from: bwfhtumx.japznrvb@gmail.com

[cid:image001.jpg@01SID_35b47.e1412191]

bwfhtumx japznrvb
 regional controller
bwfhtumx.japznrvb@gmail.com<mailto:bwfhtumx.japznrvb@gmail.com>
~~~~~~~~~~

Cleaned:
bwfhtumx japznrvb

bwfhtumx.japznrvb@gmail.com
~~~~~~
~~~~~~
~~~~~~~~~~

received from: idkfgcnq.vjwhmzor@gmail.com

good afternoon,

can you please unlock my username for erp. i have been locked for too many log in attempts.

vyjmlain hvjbmdgi
senior technical service rep
inside-sales@company.com<mailto:inside-sales@company.com>

Cleaned:
good afternoon,

can you please unlock my username for erp. i have been locked for too many log in attempts.

vyjmlain hvjbmdgi
senior technical service rep
inside-sales@company.com
~~~~~~
~~~~~~
~~~~~~~~~~

received from: koahsriq.wdugqatr@gmail.com

hello,
we need a new pc name
service tag of the pc is
fy80nkssc2

с уважением,
евгения.

koahsriq wdugqatr,
administrative assistant.
koahsriq.wdugqatr@gmail.com<mailto:koahsriq.wdugqatr@gmail.com>

company ooo | vavilova 5, corp.3 | russia, 119334 russia | www.company.com<

Cleaned:
hello,
we need a new pc name
service tag of the pc is
fy80nkssc2

\u0441 \u0443\u0432\u0430\u0436\u0435\u043d\u0438\u0435\u043c,
\u0435\u0432\u0433\u0435\u043d\u0438\u044f.

koahsriq wdugqatr,
administrative assistant.
koahsriq.wdugqatr@gmail.com

company ooo | vavilova 5, corp.3 | russia, 119334 russia | www.company.com<
~~~~~~
~~~~~~
~~~~~~~~~~
~~~~~~~~~~

```
received from: eqwaiphc.qxwfeuth@gmail.com

could you please advise what steps i should take to allow the attached spreadsheet to refresh d
ata from crm?   below is the error message that i get when i open and enable content.

[cid:image001.jpg@01d1fc45.6ce205f0]

mictbdhryhle w.  burnhntyham
regional key account manager – msc east coast
eqwaiphc.qxwfeuth@gmail.com<mailto:eqwaiphc.qxwfeuth@gmail.com>

Cleaned:
could you please advise what steps i should take to allow the attached spreadsheet to refresh d
ata from crm?   below is the error message that i get when i open and enable content.



mictbdhryhle w.  burnhntyham
regional key account manager \u2013 msc east coast
eqwaiphc.qxwfeuth@gmail.com
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~


received from: crkdjbot.qiztrxne@gmail.com

hello,


t:\HostName_768\teams\corporate governance and  t:\HostName_768\teams\proxy… these folders were
deleted by an it helpdesk employee over the weekend.  we need immediate restoration back to fri
day so these folders and all of the files contained within are restored.

this is a priority request.  please respond within the hour.


best,
debgrtybie savgrtyuille
sr. corporate paralegal
company inc.
crkdjbot.qiztrxne@gmail.com<mailto:crkdjbot.qiztrxne@gmail.com>



Cleaned:
hello,


t:\HostName_768\teams\corporate governance and  t:\HostName_768\teams\proxy\u2026 these folders
were deleted by an it helpdesk employee over the weekend.  we need immediate restoration back t
o friday so these folders and all of the files contained within are restored.

this is a priority request.  please respond within the hour.


best,
debgrtybie savgrtyuille
sr. corporate paralegal
company inc.
crkdjbot.qiztrxne@gmail.com
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~
```

- **Clean up caller ids in description**

In [128…

```
uniq_callers = set(dataset.caller.tolist())
len(uniq_callers)
```

Out[128…

```
2950
```

In [129…
```
from pandas.core.common import flatten
callers_tokens = set(flatten([i.split() for i in uniq_callers]))
len(callers_tokens)
```

Out[129… 5900

In [130…
```
def clean_callers(text: str, callers_tokens=callers_tokens) -> str:
    '''strips out caller ids from the descriptions'''
    return ' '.join([w for w in text.split()  if w.lower() not in callers_tokens])

test = '''yfqoaepn xnezhsit
managing director
finance manager cee
'''

clean_callers(test)
```

Out[130… 'managing director finance manager cee'

• Clean Irrelevant Information

- Clean up Emails, Links, Dates, Telephone Numbers

In [131…
```
# !pip -q installspacy
# !python -m spacy downlaod en_core_web_sm
# !pip -q installcontractions
import nltk
import spacy

nltk.download('punkt')
nltk.download('stopwords')
# Initialize spacy 'en_core_web_sm' model
nlp = spacy.load('en_core_web_sm', disable=['parser'])
```

```
[nltk_data] Downloading package punkt to
[nltk_data]     C:\Users\surya\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\surya\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

In [132…
```
from utils.link_handler import LinkHandler
from utils.email_handler import EmailHandler
from utils.date_handler import DateHandler
from utils.tel_handler import TelHandler
```

In [133…
```
test = "Date: 12/2/2024"
doc = nlp(test)
toks = [t.text for t in doc]
toks = [t.strip() for t in toks]
text = " ".join(toks)
tags = ['O']*len(doc)
tag = "DATE"
date_hdlr = DateHandler()
date_indices = date_hdlr.match_ref(text, toks, tags,
                                   entity=tag,
                                   verbose=True)
tags = ['DATE' if idx in date_indices else 'O'
        for idx in range(len(toks))]
print(text)
print(toks)
print(tags)
pprint(list(zip(toks, tags)), compact=True)
```

```
Final Matches DATE: ['12/2/2024']
```

```
Date : 12/2/2024
['Date', ':', '12/2/2024']
['O', 'O', 'DATE']
[('Date', 'O'), (':', 'O'), ('12/2/2024', 'DATE')]
```

In [134…

```
test = "mailto: john.doe@gmail.com from: jane.doe@outlook.com"
doc = nlp(test)
toks = [t.text for t in doc]
toks = [t.strip() for t in toks]
text = " ".join(toks)
tags = ['O']*len(doc)
tag = "MAIL"
email_hdlr = EmailHandler()
email_indices = email_hdlr.match_ref(text, toks, tags,
                                     entity=tag,
                                     verbose=True)
tags = [tag if idx in email_indices else 'O'
        for idx in range(len(toks))]
print(text)
print(toks)
print(tags)
pprint(list(zip(toks, tags)), compact=True)
```

```
Final Matches MAIL: ['john.doe@gmail.com', 'jane.doe@outlook.com']

mailto : john.doe@gmail.com from : jane.doe@outlook.com
['mailto', ':', 'john.doe@gmail.com', 'from', ':', 'jane.doe@outlook.com']
['O', 'O', 'MAIL', 'O', 'O', 'MAIL']
[('mailto', 'O'), (':', 'O'), ('john.doe@gmail.com', 'MAIL'), ('from', 'O'),
 (':', 'O'), ('jane.doe@outlook.com', 'MAIL')]
```

In [135…

```
test = "www.google.com/?search Search Results: ..."
doc = nlp(test)
toks = [t.text for t in doc]
toks = [t.strip() for t in toks]
text = " ".join(toks)
tags = ['O']*len(doc)
tag = "LINK"
link_hdlr = LinkHandler()
link_indices = link_hdlr.match_ref(text, toks, tags,
                                   entity=tag,
                                   verbose=True)
tags = [tag if idx in link_indices else 'O'
        for idx in range(len(toks))]
print(text)
print(toks)
print(tags)
pprint(list(zip(toks, tags)), compact=True)
```

```
Final Matches LINK: ['www.google.com/?search']

www.google.com/?search Search Results : ...
['www.google.com/?search', 'Search', 'Results', ':', '...']
['LINK', 'O', 'O', 'O', 'O']
[('www.google.com/?search', 'LINK'), ('Search', 'O'), ('Results', 'O'),
 (':', 'O'), ('...', 'O')]
```

In [136…

```
test = "Tel +1 724 539 5257"
doc = nlp(test)
toks = [t.text for t in doc]
toks = [t.strip() for t in toks]
text = " ".join(toks)
tags = ['O']*len(doc)
tag = "TEL"
tel_hdlr = TelHandler()
tel_indices = tel_hdlr.match_ref(text, toks, tags,
                                 entity=tag,
                                 verbose=True)
tags = [tag if idx in tel_indices else 'O'
        for idx in range(len(toks))]
```

```
print(text)
print(toks)
print(tags)
pprint(list(zip(toks, tags)), compact=True)
```

```
Final Matches TEL: ['Tel', '+1', '724', '539', '5257']

Tel +1 724 539 5257
['Tel', '+1', '724', '539', '5257']
['TEL', 'TEL', 'TEL', 'TEL', 'TEL']
[('Tel', 'TEL'), ('+1', 'TEL'), ('724', 'TEL'), ('539', 'TEL'), ('5257', 'TEL')]
```

In [137…

```
def clean_irrelevant_info(text: str) -> str:
    '''strips out emails, dates, website links and telephone numbers in the text'''
    doc = nlp(text)
    toks = [t.text for t in doc]
    toks = [t.strip() for t in toks]
    text = " ".join(toks)
    tags = ['O']*len(doc)
    tel_indices = tel_hdlr.match_ref(text, toks, tags,
                                     entity='TEL',
                                     verbose=False)
    tags = ['TEL' if idx in tel_indices else 'O'
            for idx in range(len(toks))]
    link_indices = link_hdlr.match_ref(text, toks, tags,
                                       entity='LINK',
                                       verbose=False)
    tags = ['LINK' if idx in link_indices else tag
            for idx, tag in enumerate(tags)]
    date_indices = date_hdlr.match_ref(text, toks, tags,
                                       entity='DATE',
                                       verbose=False)
    tags = ['DATE' if idx in date_indices else tag
            for idx, tag in enumerate(tags)]
    email_indices = email_hdlr.match_ref(text, toks, tags,
                                         entity='MAIL',
                                         verbose=False)
    tags = ['MAIL' if idx in email_indices else tag
            for idx, tag in enumerate(tags)]
    # print(toks)
    # print(tags)
    # print([tok for tok, tag in zip(toks, tags) if tag == "O"])

    text = str(" ".join([tok for tok, tag in zip(toks, tags) if tag == "O"]))
    # Remove HTML special entities (e.g. &amp;)
    text = re.sub(r'\&\w*;', '', text)
    # Remove hyperlinks
    text = re.sub(r'https?:\/\/.*\/\w*', '', text)
    return text
```

In [138…

```
%%time

test = "Tel +1 724 539 5257 www.google.com/?search Search Results: ... Date: 12/2/2024 mailto:
clean_irrelevant_info(test)
```

```
Wall time: 9 ms
```

Out[138…

```
'Search Results : ... Date : mailto : from :'
```

- ## Clean Anchors

In [139…

```
import re

anchors = ['received from:', 'received from :', 'received from:',
           'from:', 'to:', 'from :', 'to :',
           'date:', 'date :', 'cid', 'gentles,',
           '^hi', '^hello', '^hello,', '^hello ,', '^dear team',
           'good morning', 'good morning,' '^hi there', '^hi there,',
           'received from:?', 'hello helpdesk', 'best$',
```

```
            'hello', 'hi', 'employee', 'manager', 'etc', 'meetings', 'welcome',
            'please', 'pls', 'sir','mam', 'regards', 'jpg', 'image', 'fyi',
            'good', 'afternoon', 'morning', 'greetings',
           ]

# to strip out larger anchors first
anchors = sorted(list(set(anchors)), key=len, reverse=True)
anchors = [re.compile(a) for a in anchors]

def clean_anchors(text: str, anchors=anchors) -> str:
    '''strips out anchor words'''
    for anchor in anchors:
        text = re.sub(anchor, '', text.strip()).strip()
    return str(text.strip())
```

In [140…

```
%%time
test = '''
hi there,

would you please help me unlock my supply_chain_software account and reset my supply_chain_soft
clean_anchors(test)
```

```
Wall time: 0 ns
```

Out[140…

```
'would you  help me unlock my supply_chain_software account and reset my supply_chain_software
password?'
```

- **Gibberish Removal**

In [141…

```
import re
import nltk
from nltk.corpus import words
nltk.download('words')
vocab = set(nltk.corpus.words.words())

gib = set(['æ','¹','å','†ç','ï¼Œ','Ž','¥','Š','å','©','ç™','å½','ä','º','†','ã','€','æ','–','¶'
gibb = r'\b(?:{})\b'.format('|'.join(gib))
def clean_gibberish(text: str) -> str:
    #gibb = r'\b(?:{})\b'.format('|'.join(gib))
    text=text.lower()
    # Remove hashtag while keeping hashtag text
    text = re.sub(r'#','', text)
    # replace '&' with 'and'
    text = re.sub(r'&;?', 'and', text)

    # Remove characters beyond Readable formart by Unicode:
    # text= ''.join(c for c in text if c <= '\uFFFF')
    # text = text.strip()
    # Remove unreadable characters  (also extra spaces)
    text = ' '.join(re.sub("[^\u0030-\u0039\u0041-\u005a\u0061-\u007a]", " ", text).split())
    text = re.sub(r"\s+[a-zA-Z]\s+", ' ', text)
    text = re.sub(' +', ' ', text)
    text = re.sub('xd','',text)
    text = re.sub(gibb,'',text)
    return str(text.strip())


def clean_oov(text: str, vocab=vocab) -> str:
    '''strips out words that are outside the given vocabulary
       (exclude out-of-vocab tokens)'''
    text = " ".join([i for i in text.split() if i.lower() in vocab])
    return str(text.strip())
```

```
[nltk_data] Downloading package words to
[nltk_data]     C:\Users\surya\AppData\Roaming\nltk_data...
[nltk_data]   Package words is already up-to-date!
```

- General Preprocessing helper functions

In [142…

```
# utility functions for text preprocesing
import string
import unicodedata
import contractions
from bs4 import BeautifulSoup
from nltk.tokenize import word_tokenize, sent_tokenize
from nltk.stem.snowball import SnowballStemmer

CUSTOM = True

stemmer = SnowballStemmer('english')
if CUSTOM:
    stop_words = set(nltk.corpus.stopwords.words('english'))
    # custom stopwords added from the most frequent words which are generic
    # and might not relate to the sentiment of the review
    stop_words.update(['urllink', ])
else:
    stop_words = set(nltk.corpus.stopwords.words('english'))


def replace_accented_chars(text: str) -> str:
    '''normalizes and replaces accented characters'''
    unaccented_text = unicodedata.normalize('NFKD', text).encode('ascii', 'ignore').decode('utf
    return str(unaccented_text)


def strip_html_tags(text: str) -> str:
    '''strips html tags like <h4> ..etc'''
    soup = BeautifulSoup(text, "html.parser")
    [s.extract() for s in soup(['iframe', 'script'])]
    stripped_text = soup.get_text()
    stripped_text = re.sub(r'[\r|\n|\r\n]+', '\n', stripped_text)
    return str(stripped_text)


def expand_contractions(text: str) -> str:
    text = contractions.fix(text)
    return str(text)


def remove_special_characters(text: str) -> str:
    '''
    Remove special characters but preserve digits and excalamation marks
    as they indicate emotionally charged review '''
    text = re.sub(r"[^A-Za-z0-9!?\'\`]", " ", text)
    return str(text)


def strip_stops(text: str, is_lower_case=False, stop_words=stop_words) -> str:
    '''strip stopwrds'''
    tokens = text.split()
    tokens = [token.strip() for token in tokens]
    if is_lower_case:
        filtered_tokens = [token for token in tokens if token not in stop_words]
    else:
        filtered_tokens = [token for token in tokens if token.lower() not in stop_words]
    filtered_text = ' '.join(filtered_tokens)
    return str(filtered_text)


def snowball_stem(text: str, stemmer=stemmer) -> str:
    '''stemming using snowball stemmer'''
    words = text.split()
    stemmed_words = [stemmer.stem(word) for word in words]
    text = " ".join(stemmed_words)
    return str(text)


def tokenize(text: str) -> str:
```

```
    '''tokenize using spaCy'''
    doc = nlp(text)
    return str(" ".join([t.text for t in doc]))


def lemmatize(text: str) -> str:
    '''lemmatize using spaCy'''
    doc = nlp(text)
    return str(" ".join([t.lemma_ for t in doc]))
```

In [173…

```
def preprocess_text(text: str,
                    fix_encoding=True,
                    translate_to_english=True,
                    clean_security_logs=True,
                    clean_emails=True,
                    strip_irrelevant=True,
                    strip_anchors=True,
                    strip_callers=True,
                    clean_html_tags=True,
                    replace_accented=True,
                    remove_special=True,
                    strip_stopwords=True,
                    clean_whitespace=True,
                    tokenize_text=True,
                    remove_gibberish=True,
                    remove_oov=False,
                    lower=True,
                    get_lemmas=True,
                    strip_numbers=True)-> str:
    '''performs all preprocessing techniques in the pipeline on a string to return a cleaned st

    if fix_encoding:
        text = fix_text(text)

    # TODO
    if translate_to_english:
        pass

    if clean_security_logs:
        text = clean_sec_logs(text)

    if clean_emails:
        text = parse_email(text)

    if strip_irrelevant:
        text = clean_irrelevant_info(text)

    if strip_anchors:
        text = clean_anchors(text)

    if strip_callers:
        text = clean_callers(text)

    if clean_html_tags:
        text = strip_html_tags(text)

    if replace_accented:
        text = replace_accented_chars(text)
    text = expand_contractions(text)

    if remove_special:
        text = remove_special_characters(text)

    if strip_stopwords:
        text = strip_stops(text)

    if clean_whitespace:
        # remove extra whitespace between tokens
        text = ' '.join(text.split())
```

```
    if tokenize_text:
        text = tokenize(text)

    if remove_gibberish:
        text = clean_gibberish(text)

    if remove_oov:
        text = clean_oov(text)

    if lower:
        text = text.lower()

    if get_lemmas:
        text = lemmatize(text)

    if strip_numbers:
        text = ' '.join([re.sub(r'\d+', '', tok) for tok in text.split()])

    return str(text.strip())
```

In [ ]:

In [174…
```
test = """
received from: phfduvwl.yqnaucep@gmail.com

hello
i failed to login my hpqc account and got below message.
could you please reset password for my hpqc account, i need it to do uacyltoe hxgaycze next wee
my user id is zhudrs

[cid:image001.png@01SID_358c2.0b26f430]
"""
```

In [175…
```
test = """received from: tbvpkjoh.wnxzhqoa@gmail.com

i need access to the following path.  please see pmgzjikq potmrkxy for approval.



tbvpkjoh wnxzhqoa
company usa plant controller
tbvpkjoh.wnxzhqoa@gmail.com<tbvpkjoh.wnxzhqoa@gmail.com>

ticket update on inplant_872683
unable to login to collaboration_platform // password reset
all my calls to my ip phone are going to warehouse_toolmail, it is not even ringing.
sales area selection on opportunities not filtering to those in which the account """
print(test)
```

```
received from: tbvpkjoh.wnxzhqoa@gmail.com

i need access to the following path.  please see pmgzjikq potmrkxy for approval.



tbvpkjoh wnxzhqoa
company usa plant controller
tbvpkjoh.wnxzhqoa@gmail.com<tbvpkjoh.wnxzhqoa@gmail.com>

ticket update on inplant_872683
unable to login to collaboration_platform // password reset
all my calls to my ip phone are going to warehouse_toolmail, it is not even ringing.
sales area selection on opportunities not filtering to those in which the account
```

In [176…
```
%%time

cleaned = preprocess_text(test)
pprint(cleaned, compact=True)
```

```
('need access follow path see pmgzjikq potmrkxy approval company usa plant '
 'controller ticket update inplant  unable login collaboration platform '
 'password reset call ip phone go warehouse toolmail even ring sale area '
 'selection opportunity filtering wch account')
Wall time: 52 ms
```

In [177…
```
k = custom_kw_extractor.extract_keywords(test)
k[0][0]
```

Out[177…
```
'access to the following path'
```

In [178…
```
def preprocess(row):
    descr = row.description
    short_descr = row.short_description

    if isinstance(descr, str):
        descr = preprocess_text(descr)
    else:
        descr = np.nan
    row['cleaned_description'] = descr

    if isinstance(short_descr, str):
        short_descr = preprocess_text(short_descr)
    else:
        short_descr = np.nan
    row['cleaned_short_description'] = short_descr
    return row
```

In [179…
```
dataset = dataset.progress_apply(preprocess, axis=1)
```

```
100%|██████████████████████████████████████████████████████████████████████████| 8432/8432
[08:59<00:00, 15.62it/s]
```

In [180…
```
dataset.isna().sum()
```

Out[180…
```
short_description             0
description                   0
caller                        0
group                         0
char_length                   0
word_length                   0
short_char_length             0
short_word_length             0
description_keywords          0
short_description_keywords    0
group_code                    0
char_length_bins              0
cleaned_description           0
cleaned_short_description     0
cleaned_char_length           0
cleaned_word_length           0
cleaned_short_char_length     0
cleaned_short_word_length     0
dtype: int64
```

In [181…
```
# claned dataset
dataset.sample(50)
```

Out[181…

	short_description	description	caller	group	char_length	w
693	unable to connect to erp	unable to connect to erp	ljodkepq dsifumxl	GRP_0	25	
2776	windows account lock out issue	windows account lock out issue	htsnaodb adjtmlzn	GRP_0	31	
4130	unable to login to engineering tool	unable to login to engineering tool	gxaudkip pgkuwtlv	GRP_0	37	
3978	trying to get the status of ticket number tick...	name:erirtc\nlanguage:\nbrowser:microsoft inte...	ntsowaem jfgslyde	GRP_0	187	

	short_description	description	caller	group	char_length	w
5186	dob report showing blank data for yzugpdco nsy...	this report was working up until september 3rd...	kyzhcsrq fwyltvpd	GRP_9	263	
3648	erp SID_1 account locked out	erp SID_1 account locked out	iauqlrjk nijdaukz	GRP_0	28	
6056	we 108 scannt nicht mehr	hallo ,\n\nunser drucker we 108 scannt nicht m...	whykbjdq gfqlnysm	GRP_24	283	
7273	document sent out via zz_mails can't be receiv...	i made a uacyltoe hxgaycze as sent document 76...	zupifghd vdqxepun	GRP_13	164	
7301	virus has been found in my laptop	\n\nreceived from: gqwdslpc.clhgpqnb@gmail.com...	omLHxJVE PYudFZBW	GRP_0	163	
6435	HostName_145 (erp SID_28): volume: /dev/hd3 on...	HostName_145 (erp SID_28): volume: /dev/hd3 on...	dkmcfreg anwmfvlg	GRP_47	104	
5645	unable to connect to printer	i am unable to connect to my printer and print...	flyqjavm knlzewqs	GRP_0	162	
4695	bitte die schreib / leseberechtigung für ordn...	guten morgen,\n\nbitte die schreib / leseberec...	htvepyua izgulrcf	GRP_24	132	
571	erp SID_34 password reset.	erp SID_34 password reset.	jvxtfhkg heptuizn	GRP_0	26	
3621	error login on to the SID_34 system.	error login on to the SID_34 system.\n-verifie...	jvshydix rzpmnylt	GRP_0	228	
3605	apac- china and apac plant have reported erp s...	apac- china and apac plant have reported erp s...	obuwfnkm ufpwmybi	GRP_14	53	
7434	windows system doesn't start	windows system doesn't start	pwksivmq dbxajims	GRP_31	28	
3288	locked me out of erp	\ni tried to change my password by password_ma...	adxuzbcv uojxrivy	GRP_0	320	
2335	ms crm dynamics error : outlook addin	ms crm dynamics error : outlook addin	eziswfym cehwzojy	GRP_0	38	
59	job mm_zscr0099_dly_merktc2 failed in job_sche...	received from: monitoring_tool@company.com\n\n...	bpctwhsn kzqsbmtp	GRP_8	119	
7558	net weaver business client does not work.	net weaver business client does not work. \ner...	gusyjcer lvbxfimr	GRP_0	78	
8196	network problems (multiple applications are ru...	my home location is usa. whenever i come to u...	clgfntoe rhtmnzsk	GRP_0	655	
1285	not able to submit lean tracker	not able to submit lean tracker	ilbkhgxd hirsqytd	GRP_0	31	
5380	unable to connect to web	unable to connect to web ,and check download f...	bearzclk xnvqipcz	GRP_0	51	
4371	outlook/crm error	\n\nreceived from: qjiutmel.fgvtxeoy@gmail.com...	qjiutmel fgvtxeoy	GRP_0	215	

	short_description	description	caller	group	char_length	w
5718	job Job_728 failed in job_scheduler at: 08/31/...	received from: monitoring_tool@company.com\n\n...	bpctwhsn kzqsbmtp	GRP_8	103	
3030	outlook not working, was ok this morning, but ...	outlook down\nwlhxrogv yawtxuod	wlhxrogv yawtxuod	GRP_0	31	
6815	kpm not working	hours punched on task are not visible in kpm	mbkxwcav wsfvmpzg	GRP_25	44	
2222	erp incident 336553/2016 status change	this is in SID_21.\n\n	kflqpite gbeoqsnc	GRP_0	20	
499	badges for msc training	\n\nreceived from: gdkiehbr.kdithjsr@gmail.com...	gdkiehbr kdithjsr	GRP_3	239	
4343	kabel vga tauschen \jionmpsf wnkpzcmv	kabel vga tauschen \jionmpsf wnkpzcmv	jionmpsf wnkpzcmv	GRP_24	37	
4623	hana report will not refresh - analysis button...	see attached workbook required to be refreshed...	pfzxecbo ptygkvzl	GRP_9	67	
2670	ooo until 5/oct/ : \\HostName_771\teams\materials	ooo until 5/oct/\n\nreceived from: zxobmreq.ud...	zxobmreq udikorhv	GRP_0	172	
7724	abended job in job_scheduler: Job_1960	received from: monitoring_tool@company.com\n\n...	ZkBogxib QsEJzdZO	GRP_6	104	
8446	configair server in production not responding ...	config air server runs into 500 internal serve...	iavozegx jpcudyfi	GRP_14	364	
734	company.com not working : dns issue	company.com not working : dns issue	iqthfjvx qkpgrfzx	GRP_0	36	
5703	.rar file query	.rar file query	ilypdtno mkdfetuq	GRP_0	15	
2409	msd crm	urgent help required, i cant link an appointme...	qmpobijv wamtbupd	GRP_40	223	
8115	abended job in job_scheduler: SID_50filesys	received from: monitoring_tool@company.com\n\n...	ZkBogxib QsEJzdZO	GRP_8	109	
5388	#mm2247736 can not inwarehouse_tool consignmen...	#mm2247736 can not inwarehouse_tool consignmen...	houcdelq wnypackq	GRP_13	84	
706	outlook/crm plug in issue	\n\nreceived from: tskvmwag.awkrdqzb@gmail.com...	tskvmwag awkrdqzb	GRP_0	350	
7897	urgent: need full py access for all german loc...	hi,\ncould you please usa me full access + per...	couskjgd uzojtkmh	GRP_2	221	
5036	connection issue	\n\nreceived from: riqmdnzs.mtlghwex@gmail.com...	riqmdnzs mtlghwex	GRP_2	234	
2653	rma #6001504109	rma #7112615210\n\nworkflow error rma needs to...	asxmeruj drqufvgj	GRP_13	77	
3420	re: need a little help--please	\n\nreceived from: bcefayom.lzhwcgvb@gmail.com...	bcefayom lzhwcgvb	GRP_18	728	

	short_description	description	caller	group	char_length	w
1974	job Job_484 failed in job_scheduler at: 10/08/...	received from: monitoring_tool@company.com\n\n...	bpctwhsn kzqsbmtp	GRP_8	103	
2560	lbxugpjw cnmfbdui-mobile phone changed	\n\nreceived from: lbxugpjw.cnmfbdui@gmail.com...	lbxugpjw cnmfbdui	GRP_0	200	
7409	unable to update password on password_manageme...	unable to update password on password_manageme...	ewvugfcy nxbdajgh	GRP_0	70	
3902	wrong unit price on inwarehouse_tool	\n\nreceived from: ovhtgsxd.dcqhnrmy@gmail.com...	ovhtgsxd dcqhnrmy	GRP_13	266	
7821	orders can not be printed.	es kann keine auftrag gedruckt werden. ziehe f...	bfjnyqhe wqhuvlfb	GRP_0	53	
160	job Job_137 failed in job_scheduler at: 10/28/...	received from: monitoring_tool@company.com\n\n...	bpctwhsn kzqsbmtp	GRP_5	103	

In [182...]
```
def get_length(row):
    try:
        row['char_length'] = len(row.description)
        row['word_length'] = len(row.description.split())
        row['short_char_length'] = len(row.short_description)
        row['short_word_length'] = len(row.short_description.split())
        row['cleaned_char_length'] = len(row.cleaned_description)
        row['cleaned_word_length'] = len(row.cleaned_description.split())
        row['cleaned_short_char_length'] = len(row.cleaned_short_description)
        row['cleaned_short_word_length'] = len(row.cleaned_short_description.split())
    except Exception:  # assign 0 length to missing rows, if any
        row['char_length'] = 0
        row['word_length'] = 0
        row['short_char_length'] = 0
        row['short_word_length'] = 0
        row['cleaned_char_length'] = 0
        row['cleaned_word_length'] = 0
        row['cleaned_short_char_length'] = 0
        row['cleaned_short_word_length'] = 0
    return row

dataset = dataset.progress_apply(get_length, axis=1)
```

```
100%|██████████| 8432/8432 [0
0:02<00:00, 3705.35it/s]
```

In [183...]
```
dataset.shape
```

Out[183...]
```
(8432, 18)
```

In [184...]
```
dataset[dataset.cleaned_char_length == 0][dataset.cleaned_short_char_length == 0]  # drop these
```

Out[184...]

	short_description	description	caller	group	char_length	word_length	short_char_length	short_word_length

In [185...]
```
dataset[dataset.cleaned_char_length == 0]  # impute with cleaned_short_description
```

Out[185...]

	short_description	description	caller	group	char_length	word_length	short_char_length	short_word

	short_description	description	caller	group	char_length	word_length	short_char_length	short_word
8266	erp无法进行采购(转给贺正平)	进行采购时显示"找不到员工1111154833的数据,请通知系统管理员"	kyagjxdhdmtjpbnz	GRP_30	36	1	16	

In [186…
```
dataset[dataset.cleaned_short_char_length == 0]  # impute with short_description
```

Out[186…

	short_description	description	caller	group	char_length	word_length	short_cha
711	id 04637	id 04637 printer have paper stuck up issue.	ongumpdzpjkrfmbc	GRP_19	43	8	
1118	id : 1064870825	id : 2175981936\n\nperson on other side discon...	efbwiadpdicafxhv	GRP_0	51	8	
5013	转发: 订单号:5212346451可以签字了	\n\nreceived from: apacjun.zhang@company.com\n...	qzbxfncrkysuqema	GRP_29	73	7	
5464	答复: 35969737/2032252	\n\nreceived from: wqzarvhx.hfsojckw@gmail.com...	wqzarvhxhfsojckw	GRP_13	174	27	

In [187…
```
def postprocess_cleaned_data(dataset):
    # TODO impute non-chineese descriptions later
    dataset = dataset[~((dataset.cleaned_char_length == 0) | (dataset.cleaned_short_char_length
    return dataset

dataset = postprocess_cleaned_data(dataset)
```

In [188…
```
# dataset.to_excel('./data/cleaned_data.xlsx', index=None)
```

In [192…
```
dataset
```

Out[192…

	short_description	description	caller	group	char_length	word_length	short_cl
0	login issue	-verified user details.(employee# & manager na...	spxjnwirpjlcoqds	GRP_0	202	33	
1	outlook	\n\nreceived from: hmjdrvpb.komuaywn@gmail.com...	hmjdrvpbkomuaywn	GRP_0	186	25	
2	cant log in to vpn	\n\nreceived from: eylqgodm.ybqkwiam@gmail.com...	eylqgodmybqkwiam	GRP_0	79	11	
3	unable to access hr_tool page	unable to access hr_tool page	xbkucsvzgcpydteq	GRP_0	29	5	
4	skype error	skype error	owlgqjmeqhcozdfx	GRP_0	12	2	
...	...	...	...	...	...	...	...
8495	emails not coming in from zz mail	\n\nreceived from: avglmrts.vhqmtiua@gmail.com...	avglmrtsvhqmtiua	GRP_29	133	19	
8496	telephony_software issue	telephony_software issue	rbozivdqgmlhrtvp	GRP_0	24	2	

	short_description	description	caller	group	char_length	word_length	short_cl
8497	vip2: windows password reset for tifpdchb pedx...	vip2: windows password reset for tifpdchb pedx...	oybwdsgx oxyhwrfz	GRP_0	50	7	
8498	machine não está funcionando	i am unable to access the machine utilities to...	ufawcgob aowhxjky	GRP_62	102	17	
8499	an mehreren pc`s lassen sich verschiedene prgr...	an mehreren pc`s lassen sich verschiedene prgr...	kqvbrspl jyzoklfx	GRP_49	81	11	

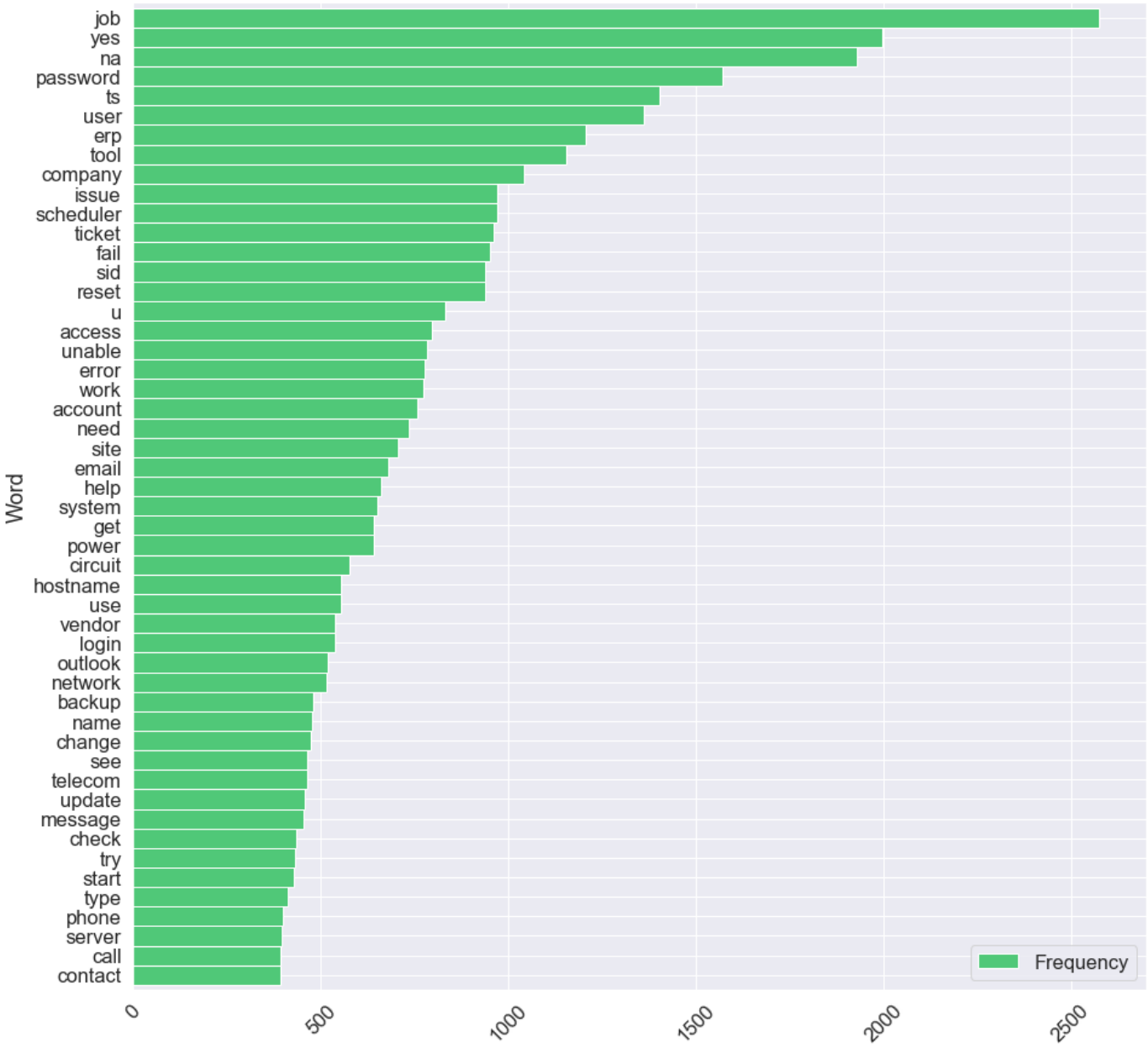
8427 rows × 18 columns

In [189…
```
dataset.sample(7)
```

Out[189…

	short_description	description	caller	group	char_length	word_length	short_char_leng
2300	windows password reset	windows password reset	dpajkrhy hwvjympt	GRP_0	22	3	
417	outlook is slow.	outlook is slow.	axdyfojg nyjlxbsk	GRP_0	16	3	
5945	blank call //gso	blank call //gso	rbozivdq gmlhrtvp	GRP_0	16	3	
722	email address in purchasing	from: dpuifqeo eglwsfkn \nsent: friday, octobe...	dfetvmzq brxavtzp	GRP_2	315	46	
5386	vpn vpn access	\n\nreceived from: qfcxbpht.oiykfzlr@gmail.com...	qfcxbpht oiykfzlr	GRP_0	209	27	
7501	unable to launch outlook	unable to launch outlook	ctxribfl hiwckyrn	GRP_0	24	4	
1077	termination for brthryian lsgthhuart	termination for brthryian lsgthhuart\n\nhello ...	lfikjasz tjbqcmvl	GRP_2	167	22	

In [190…
```
# top 50 most frequent words in text
top_N = 50

words = (dataset.cleaned_description.str.cat(sep=' ').split())
rslt = pd.DataFrame(Counter(words).most_common(top_N),
                    columns=['Word', 'Frequency']).set_index('Word')
rslt[:50].transpose()
sns.set(font_scale=1.5)  # scale up font size
rslt.sort_values(by='Frequency', ascending=True).plot(kind='barh', width=1, figsize=(15, 15), c
plt.xticks(rotation=45)
plt.savefig('dist_after_cleaning.png')
plt.show()
```



In [1]:
```
#importing the data
import pandas as pd
data = pd.read_excel('./data/cleaned_data.xlsx')
data.head()
```

Out[1]:

	short_description	description	caller	group	char_length	word_length	short_char_le
0	login issue	-verified user details.(employee# & manager na...	spxjnwir pjlcoqds	GRP_0	202	33	
1	outlook	\n\nreceived from: hmjdrvpb.komuaywn@gmail.com...	hmjdrvpb komuaywn	GRP_0	186	25	
2	cant log in to vpn	\n\nreceived from: eylqgodm.ybqkwiam@gmail.com...	eylqgodm ybqkwiam	GRP_0	79	11	
3	unable to access hr_tool page	unable to access hr_tool page	xbkucsvz gcpydteq	GRP_0	29	5	
4	skype error	skype error	owlgqjme qhcozdfx	GRP_0	12	2	

In [2]:
```
# reproducibility
import random
seed = 7
random.seed(seed)
```

In [3]:
```
#drop unwanted variables
data.drop(['description','char_length','word_length','short_char_length',
                    'short_word_length','group_code','char_length_bins'], axis = 1, inplace = Tr
```

In [4]:
```
data.head()
```

Out[4]:

	short_description	caller	group	description_keywords	short_description_keywords	cleaned_description	c
0	login issue	spxjnwir pjlcoqds	GRP_0	verified user details.	login issue	verify user detail name check user name ad res...	
1	outlook	hmjdrvpb komuaywn	GRP_0	appearing in my outlook calendar	outlook	team skype appear outlook calendar somebody ad...	
2	cant log in to vpn	eylqgodm ybqkwiam	GRP_0	log on to vpn	log in to vpn	log vpn	
3	unable to access hr_tool page	xbkucsvz gcpydteq	GRP_0	tool page	tool page	unable access hr tool page	
4	skype error	owlgqjme qhcozdfx	GRP_0	skype error	skype error	skype error	

In [5]:
```
from nltk.tokenize import word_tokenize
from string import punctuation
from nltk.corpus import stopwords
from nltk.stem import WordNetLemmatizer #word stemmer class
lemma = WordNetLemmatizer()
```

In [6]:
```
import nltk
nltk.download('words')
words = set(nltk.corpus.words.words())
nltk.download('stopwords')
from nltk.corpus import stopwords
nltk.download('wordnet')
from nltk.corpus import wordnet
import re
```

```
[nltk_data] Downloading package words to
[nltk_data]     C:\Users\surya\AppData\Roaming\nltk_data...
[nltk_data]   Package words is already up-to-date!
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\surya\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package wordnet to
[nltk_data]     C:\Users\surya\AppData\Roaming\nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
```

In [7]:
```
import re

def normalizer(text):
    text = " ".join(filter(lambda x: x[0]!= '@' , text.split()))
    text = re.sub('[^a-zA-Z]', ' ', text)
    text = text.lower()
    text = re.sub(' +', ' ', text).strip()
    text = text.split()
    text = [words for words in text if not words in set(stopwords.words('english'))]
    text = [lemma.lemmatize(word) for word in text]

    text = " ".join(text)
    return text
```

In [8]:
```
data['shrt_data'] = data['cleaned_short_description'].apply(normalizer)
```

In [9]:
```
data.head()
```

Out[9]:

	short_description	caller	group	description_keywords	short_description_keywords	cleaned_description	c
0	login issue	spxjnwir pjlcoqds	GRP_0	verified user details.	login issue	verify user detail name check user name ad res...	
1	outlook	hmjdrvpb komuaywn	GRP_0	appearing in my outlook calendar	outlook	team skype appear outlook calendar somebody ad...	
2	cant log in to vpn	eylqgodm ybqkwiam	GRP_0	log on to vpn	log in to vpn	log vpn	
3	unable to access hr_tool page	xbkucsvz gcpydteq	GRP_0	tool page	tool page	unable access hr tool page	
4	skype error	owlgqjme qhcozdfx	GRP_0	skype error	skype error	skype error	

In [10]:
```
## Combining the Description and Short_description Columnns
data['Combine_Descrption'] = data['shrt_data']+data['cleaned_description']
```

In [11]:
```
data.head()
```

Out[11]:

	short_description	caller	group	description_keywords	short_description_keywords	cleaned_description	c
0	login issue	spxjnwir pjlcoqds	GRP_0	verified user details.	login issue	verify user detail name check user name ad res...	
1	outlook	hmjdrvpb komuaywn	GRP_0	appearing in my outlook calendar	outlook	team skype appear outlook calendar somebody ad...	
2	cant log in to vpn	eylqgodm ybqkwiam	GRP_0	log on to vpn	log in to vpn	log vpn	
3	unable to access hr_tool page	xbkucsvz gcpydteq	GRP_0	tool page	tool page	unable access hr tool page	
4	skype error	owlgqjme qhcozdfx	GRP_0	skype error	skype error	skype error	

In [12]:
```
data['Combine_Descrption'][0]
```

Out[12]:
```
'login issueverify user detail name check user name ad reset password advise user login check c
aller confirm able login issue resolve'
```

In [13]:
```
#remover duplicate words in a sentence
def uniquify(string):
    output = []
    seen = set()
    for word in string.split():
        if word not in seen:
            output.append(word)
            seen.add(word)
    return ' '.join(output)
```

In [14]:
```
data['Combine_Descrption'] = data['Combine_Descrption'].apply(uniquify)
```

In [15]:
```
data['Combine_Descrption'][0]
```

Out[15]:
```
'login issueverify user detail name check ad reset password advise caller confirm able issue re
solve'
```

In [16]:
```
data=data[['group','caller','Combine_Descrption']]
```

In [17]:
```
data.head()
```

Out[17]:

	group	caller	Combine_Descrption
0	GRP_0	spxjnwir pjlcoqds	login issueverify user detail name check ad re...
1	GRP_0	hmjdrvpb komuaywn	outlookteam skype appear outlook calendar some...
2	GRP_0	eylqgodm ybqkwiam	ca nt log vpnlog vpn
3	GRP_0	xbkucsvz gcpydteq	unable access hr tool pageunable page
4	GRP_0	owlgqjme qhcozdfx	skype errorskype error

In [18]:
```
#collapsing the targets into 3 based on count frequency
def get_cat(GRP):

    if GRP in ['GRP_8','GRP_24','GRP_12','GRP_9','GRP_2','GRP_19','GRP_3','GRP_6','GRP_13','GRP
        return 'L2'

    elif GRP in ['GRP_17','GRP_31', 'GRP_7','GRP_34','GRP_26','GRP_40','GRP_28','GRP_41','GRP_3
                 'GRP_62','GRP_48','GRP_23','GRP_60','GRP_39','GRP_27','GRP_37','GRP_36','GRP_4
                 'GRP_43','GRP_32','GRP_66','GRP_68','GRP_38','GRP_63','GRP_56','GRP_58','GRP_7
                 'GRP_35']:
        return 'L3'
    else:
        return 'L1'
```

In [19]:
```
data['grp'] = data['group'].apply(get_cat)
```

In [20]:
```
data['grp'].value_counts(normalize=True) * 100
```

Out[20]:
```
L1    46.952087
L2    40.547913
L3    12.500000
Name: grp, dtype: float64
```

In [21]:
```
#chi square test
import scipy.stats as stats
data_crosstab = pd.crosstab(data['caller'],
                            data['grp'],
                           margins=True, margins_name="Total")
```

In [22]:
```
#HO: No relationship between caller and the target class
#HA: Signifacnt relationship between caller and target class

# significance level
alpha = 0.05

# Calcualtion of Chisquare test statistics
chi_square = 0
rows = data['caller'].unique()
columns = data['grp'].unique()
for i in columns:
    for j in rows:
        O = data_crosstab[i][j]
        E = data_crosstab[i]['Total'] * data_crosstab['Total'][j] / data_crosstab['Total']['Tot
        chi_square += (O-E)**2/E

# The p-value approach
print("Approach 1: The p-value approach to hypothesis testing in the decision rule")
p_value = 1 - stats.norm.cdf(chi_square, (len(rows)-1)*(len(columns)-1))
conclusion = "Failed to reject the null hypothesis."
```

```
if p_value <= alpha:
    conclusion = "Null Hypothesis is rejected."

print("chisquare-score is:", chi_square, " and p value is:", p_value)
print(conclusion)
```

```
Approach 1: The p-value approach to hypothesis testing in the decision rule
chisquare-score is: 10284.902651089269  and p value is: 0.0
Null Hypothesis is rejected.
```

In [23]:
```
#getting count freq of callers
count_freq = dict(data['caller'].value_counts())
```

In [24]:
```
data['count_freq'] = data['caller']
data['count_freq'] = data['count_freq'].map(count_freq)
```

In [25]:
```
import numpy as np
```

In [26]:
```
data['caller'] = np.where(data['count_freq']>1, 'Rep', 'No')
```

In [27]:
```
# creating a new variable callers that classifies caller into repetative and one off callers
data['caller'].value_counts(normalize=True) * 100
```

Out[27]:
```
Rep    82.732448
No     17.267552
Name: caller, dtype: float64
```

In [28]:
```
#chi square test on the new caller variable
data_crosstab = pd.crosstab(data['caller'],
                            data['grp'],
                           margins=True, margins_name="Total")
```

In [29]:
```
#HO: No relationship between caller and the target class
#HA: Signifacnt relationship between caller and target class

# significance level
alpha = 0.05

# Calcualtion of Chisquare test statistics
chi_square = 0
rows = data['caller'].unique()
columns = data['grp'].unique()
for i in columns:
    for j in rows:
        O = data_crosstab[i][j]
        E = data_crosstab[i]['Total'] * data_crosstab['Total'][j] / data_crosstab['Total']['Tot
        chi_square += (O-E)**2/E

# The p-value approach
print("Approach 1: The p-value approach to hypothesis testing in the decision rule")
p_value = 1 - stats.norm.cdf(chi_square, (len(rows)-1)*(len(columns)-1))
conclusion = "Failed to reject the null hypothesis."
if p_value <= alpha:
    conclusion = "Null Hypothesis is rejected."

print("chisquare-score is:", chi_square, " and p value is:", p_value)
print(conclusion)
```

```
Approach 1: The p-value approach to hypothesis testing in the decision rule
chisquare-score is: 166.28918317240448  and p value is: 0.0
Null Hypothesis is rejected.
```

Vizualizing the type of queries based on groups

In [30]:
```
from wordcloud import WordCloud, STOPWORDS
stopwords = set(STOPWORDS)
```

In [31]:
```
grp_0 = data[data["grp"] == 'L1']
```

```
grp_0.head()
```

Out[31]:

	group	caller	Combine_Descrption	grp	count_freq
0	GRP_0	No	login issueverify user detail name check ad re...	L1	1
1	GRP_0	Rep	outlookteam skype appear outlook calendar some...	L1	4
2	GRP_0	Rep	ca nt log vpnlog vpn	L1	3
3	GRP_0	Rep	unable access hr tool pageunable page	L1	3
4	GRP_0	Rep	skype errorskype error	L1	5

In [32]:
```
import warnings
warnings.filterwarnings('ignore')

import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
```

In [33]:
```
font = {'family': 'serif',
        'color':  'darkred',
        'weight': 'normal',
        'size': 16,
        }
All_words = ""
All_words += " ".join(grp_0['Combine_Descrption'])
wordcloud = WordCloud(background_color='white').generate(All_words) # width and height in the w
plt.figure(figsize=(15,15))
plt.title("WordCloud of Combine Description for L1 Group", fontdict=font)
plt.imshow(wordcloud, interpolation="bilinear")
plt.axis("off")
plt.show()
```

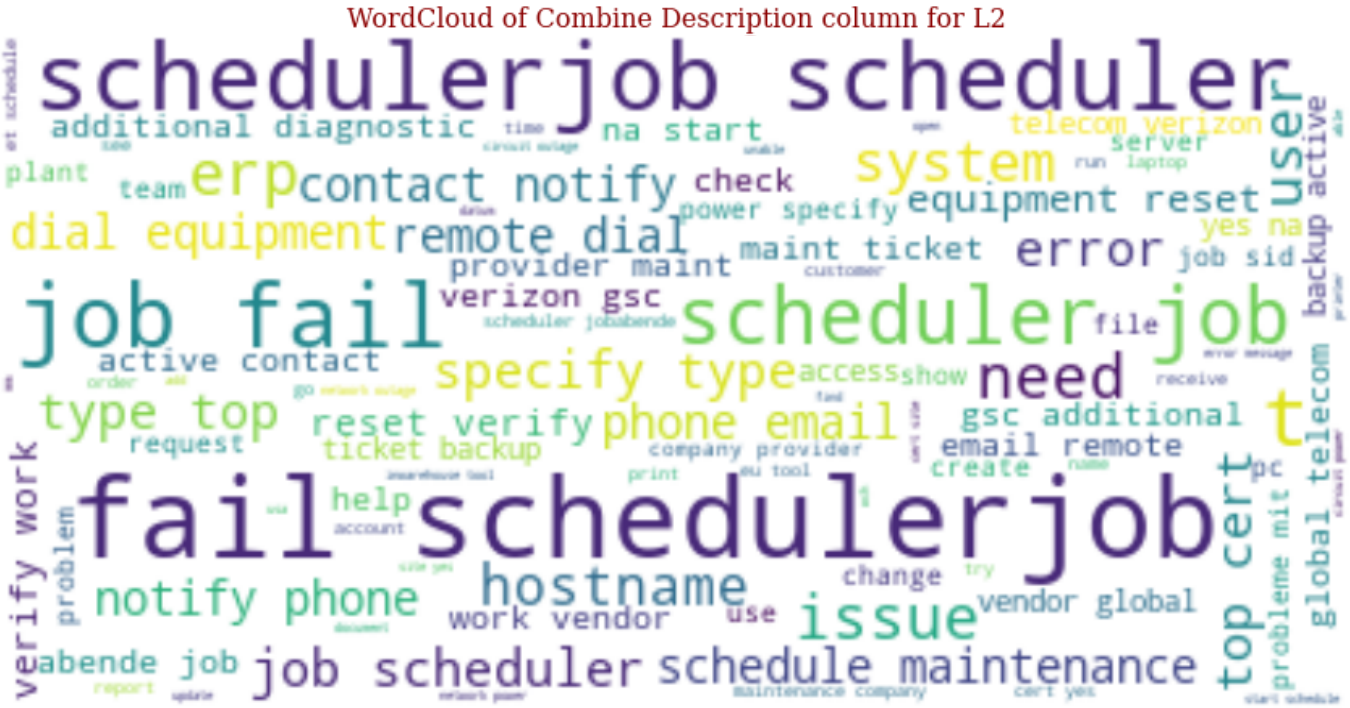
In [34]:
```
grp_L2 = data[data["grp"] == 'L2']
grp_L2.head()
```

Out[34]:

	group	caller	Combine_Descrption	grp	count_freq
17	GRP_3	Rep	undock pc screen come backundock back	L2	2
32	GRP_4	Rep	duplication network addressgentle two device t...	L2	6
43	GRP_5	Rep	reroute job printer issue need resolve todaypr...	L2	2

	group	caller	Combine_Descrption	grp	count_freq
47	GRP_6	Rep	job fail schedulerjob 1424 scheduler 09 06 00	L2	810
50	GRP_8	Rep	job mm zscr dly merktc fail schedulerjob zscr0...	L2	810

In [35]:
```
font = {'family': 'serif',
        'color':  'darkred',
        'weight': 'normal',
        'size': 16,
        }
All_words = ""
All_words += " ".join(grp_L2['Combine_Descrption'])
wordcloud = WordCloud(background_color='white').generate(All_words) # width and height in the w
plt.figure(figsize=(15,15))
plt.title("WordCloud of Combine Description column for L2", fontdict=font)
plt.imshow(wordcloud, interpolation="bilinear")
plt.axis("off")
plt.show()
```



In [36]:
```
grp_L3 = data[data["grp"] == 'L3']
grp_L3.head()
```

Out[36]:

	group	caller	Combine_Descrption	grp	count_freq
6	GRP_1	Rep	event critical hostname company com value moun...	L3	51
49	GRP_7	Rep	status change telephony softwareclose call age...	L3	3
83	GRP_11	Rep	engineering tool draw original pdf format show...	L3	2
140	GRP_15	Rep	channel partner receive multiple email erp urg...	L3	11
153	GRP_17	No	reset password use management tool resetget er...	L3	1

In [37]:
```
font = {'family': 'serif',
        'color':  'darkred',
        'weight': 'normal',
        'size': 16,
        }
All_words = ""
All_words += " ".join(grp_L3['Combine_Descrption'])
wordcloud = WordCloud(background_color='white').generate(All_words) # width and height in the w
plt.figure(figsize=(15,15))
plt.title("WordCloud of Combine Description column for L3", fontdict=font)
```

```
plt.imshow(wordcloud, interpolation="bilinear")
plt.axis("off")
plt.show()
```

In [38]:
```
data.drop(columns = ['count_freq'], axis = 1,inplace=True)
data.head()
```

Out[38]:

	group	caller	Combine_Descrption	grp
0	GRP_0	No	login issueverify user detail name check ad re...	L1
1	GRP_0	Rep	outlookteam skype appear outlook calendar some...	L1
2	GRP_0	Rep	ca nt log vpnlog vpn	L1
3	GRP_0	Rep	unable access hr tool pageunable page	L1
4	GRP_0	Rep	skype errorskype error	L1

Data pre-processing

In [39]:
```
from sklearn import preprocessing

le = preprocessing.LabelEncoder()
data["Label_grp"] = le.fit_transform(data["grp"])
y_classes_len = len(le.classes_)
le.classes_
print(y_classes_len)
```

```
3
```

In [40]:
```
data['caller']=np.where(data['caller']=='Rep', 1, 0)
```

In [41]:
```
data.head()
```

Out[41]:

	group	caller	Combine_Descrption	grp	Label_grp
0	GRP_0	0	login issueverify user detail name check ad re...	L1	0
1	GRP_0	1	outlookteam skype appear outlook calendar some...	L1	0
2	GRP_0	1	ca nt log vpnlog vpn	L1	0
3	GRP_0	1	unable access hr tool pageunable page	L1	0
4	GRP_0	1	skype errorskype error	L1	0

In [42]:
```
data.drop(columns = ['group','grp'], axis = 1,inplace=True)
data.head()
```

Out[42]:

	caller	Combine_Descrption	Label_grp
0	0	login issueverify user detail name check ad re...	0
1	1	outlookteam skype appear outlook calendar some...	0
2	1	ca nt log vpnlog vpn	0
3	1	unable access hr tool pageunable page	0
4	1	skype errorskype error	0

In [43]:
```
import pandas as pd
from sklearn.feature_extraction.text import CountVectorizer
```

In [44]:
```
def mytokenizer(x):
    return [y for y in x.split() if len(y) > 2]
```

In [45]:
```
vec = CountVectorizer(tokenizer=mytokenizer, min_df=0.005)
X = vec.fit_transform(data['Combine_Descrption'])
df = pd.DataFrame(X.toarray(), columns=vec.get_feature_names())
```

In [46]:
```
s1 = pd.Series(data['Label_grp'], name="Label_grp")
s2= pd.Series(data['caller'], name="caller")
```

In [47]:
```
df1=df.reset_index(drop=True)
s3=s1.reset_index(drop=True)
s4=s2.reset_index(drop=True)
```

In [48]:
```
result = pd.concat([df1, s3,s4], axis=1)
```

In [49]:
```
result.head()
```

Out[49]:

	2016	abende	able	access	account	action	active	add	additional	address	...	without	wle	work	wor
0	0	0	1	0	0	0	0	0	0	0	...	0	0	0	
1	0	0	0	0	0	0	0	0	0	0	...	0	0	0	
2	0	0	0	0	0	0	0	0	0	0	...	0	0	0	
3	0	0	0	1	0	0	0	0	0	0	...	0	0	0	
4	0	0	0	0	0	0	0	0	0	0	...	0	0	0	

5 rows × 427 columns

In [50]:
```
result = result.loc[:,~result.columns.duplicated()]
```

In [51]:
```
result.head()
```

Out[51]:

	2016	abende	able	access	account	action	active	add	additional	address	...	window	without	wle	w
0	0	0	1	0	0	0	0	0	0	0	...	0	0	0	
1	0	0	0	0	0	0	0	0	0	0	...	0	0	0	
2	0	0	0	0	0	0	0	0	0	0	...	0	0	0	
3	0	0	0	1	0	0	0	0	0	0	...	0	0	0	
4	0	0	0	0	0	0	0	0	0	0	...	0	0	0	

5 rows × 426 columns

In [52]:
```
#creating a copy of the train data and seperating the target column and the predictor variables

X=result.drop(['Label_grp'],axis=1)
y=result['Label_grp']
```

In [53]:
```
#loading required libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.preprocessing import binarize
from sklearn.metrics import confusion_matrix
from sklearn.linear_model import LogisticRegression
from sklearn.neighbors import KNeighborsClassifier
from scipy.stats import zscore
from sklearn.svm import SVC
from sklearn.metrics import confusion_matrix, classification_report,accuracy_score,f1_score
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import precision_recall_fscore_support
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from xgboost import XGBClassifier
from lightgbm import LGBMClassifier
from sklearn.ensemble import AdaBoostClassifier
from sklearn.ensemble import GradientBoostingClassifier
from sklearn.model_selection import StratifiedKFold,cross_val_score,KFold
from sklearn.model_selection import RandomizedSearchCV,GridSearchCV
from sklearn import metrics
import seaborn as sns
sns.set(color_codes=True)
%matplotlib inline
import warnings
warnings.filterwarnings('ignore')
```

In [54]:
```
# splitting data training dataset into train and test set for independent attributes
X_train, X_test, Y_train, Y_test =train_test_split(X,y, test_size=.30,random_state=105)
```

In [55]:
```
# Initializaing various classification algorithms with normal dataset and choosing the best mod

models = []
models.append(("LR", LogisticRegression()))
models.append(("KNN", KNeighborsClassifier()))
models.append(("GNB", GaussianNB()))
models.append(("SVM", SVC(kernel='linear',probability=True)))
models.append(("DT", DecisionTreeClassifier()))
models.append(("RF", RandomForestClassifier()))
models.append(("GBT", GradientBoostingClassifier()))
models.append(("XGB", XGBClassifier(verbosity=0)))
models.append(("LightGBM",LGBMClassifier()))

#testing models
results = []
names = []

for name, model in models:
    kfold = StratifiedKFold(n_splits=10, random_state=None,shuffle=True)
    cv_results = cross_val_score(model, X_train, Y_train, cv=kfold, scoring='roc_auc_ovo')
    results.append(cv_results)
    names.append(name)
    msg = '%s: %f%% (%f%%)' % (name, cv_results.mean()*100, cv_results.std()*100)
    print(msg)
```

```
LR: 83.071664% (1.429062%)
KNN: 78.526848% (2.182676%)
GNB: 74.838250% (1.969526%)
```

```
SVM: 79.743117% (1.366922%)
DT: 71.459423% (1.369076%)
RF: 85.569903% (1.172127%)
GBT: 83.958214% (0.826341%)
XGB: 85.167831% (1.669024%)
LightGBM: 85.069043% (0.811227%)
```

LightGBM with RandomsearchCV

In [56]:
```
from scipy.stats import randint as sp_randint
from scipy.stats import uniform as sp_uniform
```

In [57]:
```
param_test ={'num_leaves': sp_randint(6, 50),
             'min_child_samples': sp_randint(100, 500),
             'min_child_weight': [1e-5, 1e-3, 1e-2, 1e-1, 1, 1e1, 1e2, 1e3, 1e4],
             'subsample': sp_uniform(loc=0.2, scale=0.8),
             'colsample_bytree': sp_uniform(loc=0.4, scale=0.6),
             'reg_alpha': [0, 1e-1, 1, 2, 5, 7, 10, 50, 100],
             'reg_lambda': [0, 1e-1, 1, 5, 10, 20, 50, 100],
             'scale_pos_weight':[1,2,6,12]}

sample = 100

#n_estimators is set to a "large value". The actual number of trees build will depend on early
lgb = LGBMClassifier(max_depth=-1, random_state=31, silent=True, metric='multi_logloss', n_jobs
gs = RandomizedSearchCV(
    estimator=lgb, param_distributions=param_test,
    n_iter=sample,
    cv=5,
    refit=True,
    random_state=314,
    verbose=True)

gs.fit(X_train, Y_train)
gs.best_params_
```

```
Fitting 5 folds for each of 100 candidates, totalling 500 fits
[Parallel(n_jobs=1)]: Using backend SequentialBackend with 1 concurrent workers.
[Parallel(n_jobs=1)]: Done 500 out of 500 | elapsed:  9.7min finished
```

Out[57]:
```
{'colsample_bytree': 0.8914513227485243,
 'min_child_samples': 117,
 'min_child_weight': 0.01,
 'num_leaves': 7,
 'reg_alpha': 0.1,
 'reg_lambda': 5,
 'scale_pos_weight': 6,
 'subsample': 0.5804609677502011}
```

In [58]:
```
lgb=LGBMClassifier(colsample_bytree= 0.8914513227485243,
 min_child_samples= 117,
 min_child_weight= 0.01,
 num_leaves=7,
 reg_alpha= 0.1,
 reg_lambda= 5,
 scale_pos_weight= 6,
 subsample= 0.5804609677502011)
lgb.fit(X_train,Y_train)
```

Out[58]:
```
LGBMClassifier(colsample_bytree=0.8914513227485243, min_child_samples=117,
               min_child_weight=0.01, num_leaves=7, reg_alpha=0.1, reg_lambda=5,
               scale_pos_weight=6, subsample=0.5804609677502011)
```

In [59]:
```
modellgb1=lgb.score(X_train,Y_train)
print('Accuracy Score of Training Data: ', modellgb1)
```

```
Accuracy Score of Training Data:  0.7319552694002033
```

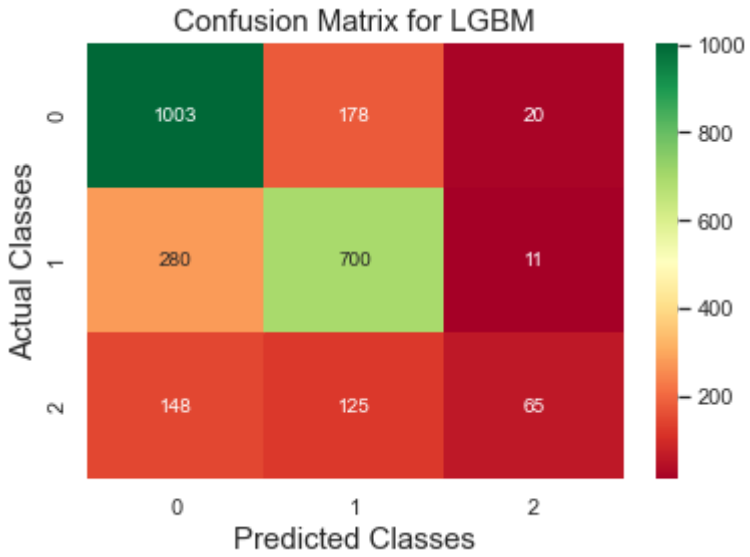
In [60]:
```
y_predictlg1= lgb.predict(X_test)
modellg1 = accuracy_score(Y_test, y_predictlg1)
print('Accuracy Score of Test Data:', modellg1)
```

```
Accuracy Score of Test Data: 0.6988142292490118
```

In [61]:
```
#printing classification report
print("Classification Report")
print(metrics.classification_report(Y_test, y_predictlg1, labels=[0,1,2]))
```

```
Classification Report
              precision    recall  f1-score   support

           0       0.70      0.84      0.76      1201
           1       0.70      0.71      0.70       991
           2       0.68      0.19      0.30       338

    accuracy                           0.70      2530
   macro avg       0.69      0.58      0.59      2530
weighted avg       0.70      0.70      0.68      2530
```

In [62]:
```
# visualizing confusion matrix
cm= confusion_matrix(Y_test, y_predictlg1)
plt.figure(figsize = (6, 4))
sns.heatmap(cm, annot = True, cmap = 'RdYlGn', fmt = 'd')
plt.ylabel('Actual Classes', fontsize = 15)
plt.xlabel('Predicted Classes', fontsize = 15)
plt.title('Confusion Matrix for LGBM', fontsize = 15);
```



Random forest with RandomsearchCV

In [63]:
```
# Number of trees in random forest
n_estimators = [int(x) for x in np.linspace(start = 50, stop = 500, num = 50)]
# Number of features to consider at every split
max_features = ['auto', 'sqrt']
# Maximum number of levels in tree
max_depth = [int(x) for x in np.linspace(10, 110, num = 11)]
max_depth.append(None)
# Minimum number of samples required to split a node
min_samples_split = range(2,100,5)
# Minimum number of samples required at each leaf node
min_samples_leaf = range(1,100,10)
# Method of selecting samples for training each tree
bootstrap = [True, False]

# Create the random grid
random_grid = {'n_estimators': n_estimators,
               'max_features': max_features,
               'max_depth': max_depth,
               'min_samples_split': min_samples_split,
               'min_samples_leaf': min_samples_leaf,
               'bootstrap': bootstrap,
                'criterion':['gini','entropy']}
```

In [64]:
```
rf = RandomForestClassifier()
rf_random = RandomizedSearchCV(estimator = rf, param_distributions = random_grid, cv = 5, verbo
rf_random.fit(X_train, Y_train)
rf_random.best_params_
```

```
Fitting 5 folds for each of 10 candidates, totalling 50 fits
[Parallel(n_jobs=-1)]: Using backend LokyBackend with 16 concurrent workers.
[Parallel(n_jobs=-1)]: Done   9 tasks      | elapsed:    8.6s
[Parallel(n_jobs=-1)]: Done  45 out of  50 | elapsed:   29.6s remaining:    3.2s
[Parallel(n_jobs=-1)]: Done  50 out of  50 | elapsed:   36.6s finished
```

Out[64]:
```
{'n_estimators': 463,
 'min_samples_split': 82,
 'min_samples_leaf': 1,
 'max_features': 'sqrt',
 'max_depth': 110,
 'criterion': 'gini',
 'bootstrap': False}
```

In [65]:
```
rf_grid1 = RandomForestClassifier(n_estimators=463,
 min_samples_split= 82,
 min_samples_leaf=1,
 max_features= 'sqrt',
 max_depth= 110,
 criterion= 'gini',
 bootstrap= False)
rf_grid1.fit(X_train, Y_train)
```

Out[65]:
```
RandomForestClassifier(bootstrap=False, max_depth=110, max_features='sqrt',
                       min_samples_split=82, n_estimators=463)
```

In [66]:
```
modelrfg1_score=rf_grid1.score(X_train,Y_train)
print('Accuracy Score of Training Data: ', modelrfg1_score)
```

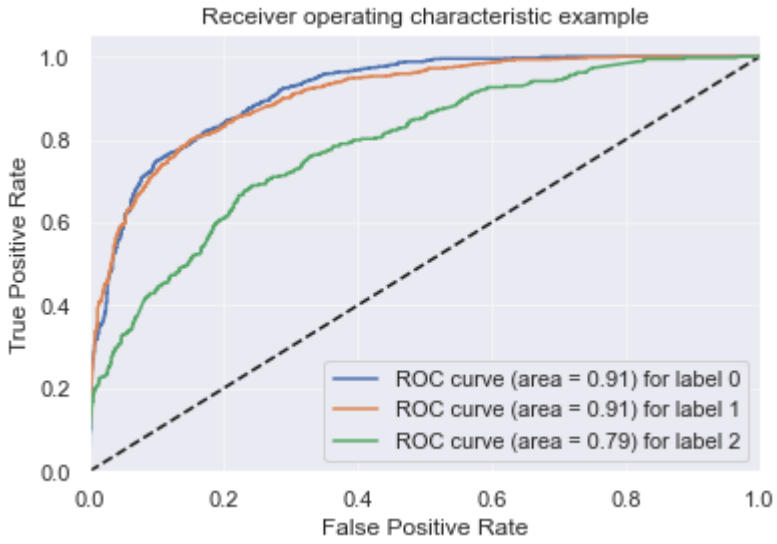
```
Accuracy Score of Training Data:  0.8542866824805151
```

In [67]:
```
y_predictrfg1= rf_grid1.predict(X_test)
modelrfg1_score = accuracy_score(Y_test, y_predictrfg1)
print('Accuracy Score of Test Data:', modelrfg1_score)
```

```
Accuracy Score of Test Data: 0.758893280632411
```

In [68]:
```
def plot_multiclass_roc(clf, X_test, y_test, n_classes, figsize=(17, 6)):
    y_score = clf.predict_proba(X_test)

    # structures
    fpr = dict()
    tpr = dict()
    roc_auc = dict()

    # calculate dummies once
    y_test_dummies = pd.get_dummies(y_test, drop_first=False).values
    for i in range(n_classes):
        fpr[i], tpr[i], _ = metrics.roc_curve(y_test_dummies[:, i], y_score[:, i])
        roc_auc[i] = metrics.auc(fpr[i], tpr[i])

    # roc for each class
    fig, ax = plt.subplots(figsize=figsize)
    ax.plot([0, 1], [0, 1], 'k--')
    ax.set_xlim([0.0, 1.0])
    ax.set_ylim([0.0, 1.05])
    ax.set_xlabel('False Positive Rate')
    ax.set_ylabel('True Positive Rate')
    ax.set_title('Receiver operating characteristic example')
    for i in range(n_classes):
        ax.plot(fpr[i], tpr[i], label='ROC curve (area = %0.2f) for label %i' % (roc_auc[i], i)
    ax.legend(loc="best")
    ax.grid(alpha=.4)
    sns.despine()
    plt.savefig('ROC_Curve_RF.png')
    plt.show()
```

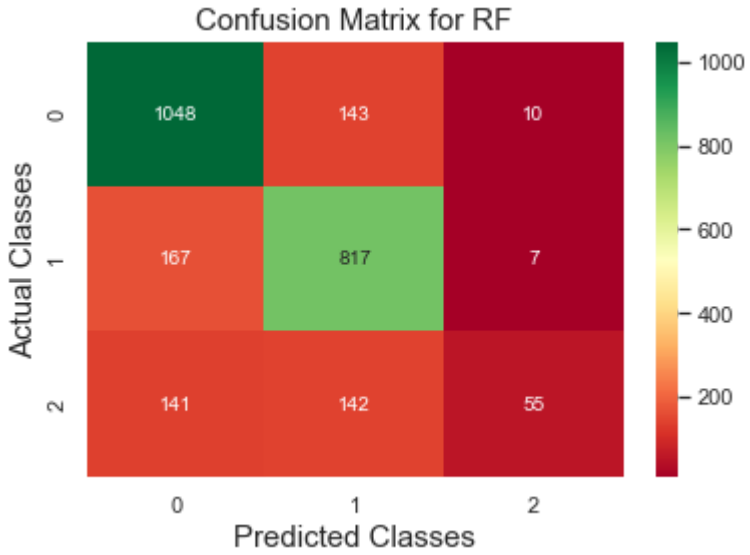
```
plot_multiclass_roc(rf_grid1, X_test, Y_test, n_classes=3, figsize=(6, 4))
```



In [69]:
```
#printing classification report
print("Classification Report")
print(metrics.classification_report(Y_test, y_predictrfg1, labels=[0,1,2]))
```

```
Classification Report
              precision    recall  f1-score   support

           0       0.77      0.87      0.82      1201
           1       0.74      0.82      0.78       991
           2       0.76      0.16      0.27       338

    accuracy                           0.76      2530
   macro avg       0.76      0.62      0.62      2530
weighted avg       0.76      0.76      0.73      2530
```

In [70]:
```
# visualizing confusion matrix
cm= confusion_matrix(Y_test, y_predictrfg1)
plt.figure(figsize = (6, 4))
sns.heatmap(cm, annot = True, cmap = 'RdYlGn', fmt = 'd')
plt.ylabel('Actual Classes', fontsize = 15)
plt.xlabel('Predicted Classes', fontsize = 15)
plt.title('Confusion Matrix for RF', fontsize = 15);
```



XGBoost with RandomsearchCV

In [71]:
```
xgb_para = {"learning_rate"    : [0.05, 0.10, 0.15, 0.20, 0.25, 0.30 ] ,
            "max_depth"        : [ 3, 4, 5, 6, 8, 10, 12, 15],
            "min_child_weight" : [ 1, 3, 5, 7 ],
            "gamma"            : [ 0.0, 0.1, 0.2 , 0.3, 0.4 ],
```

```
            "colsample_bytree" : [ 0.3, 0.4, 0.5 , 0.7 ]
           }

xgb = XGBClassifier()
xgb_hy = RandomizedSearchCV(estimator = xgb, param_distributions = xgb_para, cv = 5, verbose=2,
xgb_hy.fit(X_train, Y_train)
xgb_hy.best_params_
```

```
Fitting 5 folds for each of 10 candidates, totalling 50 fits
[Parallel(n_jobs=-1)]: Using backend LokyBackend with 16 concurrent workers.
[Parallel(n_jobs=-1)]: Done   9 tasks      | elapsed:  2.8min
[Parallel(n_jobs=-1)]: Done  45 out of  50 | elapsed:  8.5min remaining:   56.5s
[Parallel(n_jobs=-1)]: Done  50 out of  50 | elapsed:  8.6min finished
```

Out[71]:
```
{'min_child_weight': 1,
 'max_depth': 6,
 'learning_rate': 0.2,
 'gamma': 0.0,
 'colsample_bytree': 0.7}
```

In [72]:
```
xgb=XGBClassifier(min_child_weight=1,
 max_depth=6,
 learning_rate= 0.2,
 gamma= 0,
 colsample_bytree=0.7)
xgb.fit(X_train,Y_train)
```

Out[72]:
```
XGBClassifier(base_score=0.5, booster='gbtree', colsample_bylevel=1,
              colsample_bynode=1, colsample_bytree=0.7, gamma=0, gpu_id=-1,
              importance_type='gain', interaction_constraints='',
              learning_rate=0.2, max_delta_step=0, max_depth=6,
              min_child_weight=1, missing=nan, monotone_constraints='()',
              n_estimators=100, n_jobs=16, num_parallel_tree=1,
              objective='multi:softprob', random_state=0, reg_alpha=0,
              reg_lambda=1, scale_pos_weight=None, subsample=1,
              tree_method='exact', validate_parameters=1, verbosity=None)
```

In [73]:
```
modelxgb_score=xgb.score(X_train,Y_train)
print('Accuracy Score of Training Data: ', modelxgb_score)
```

```
Accuracy Score of Training Data:  0.8607251779057946
```

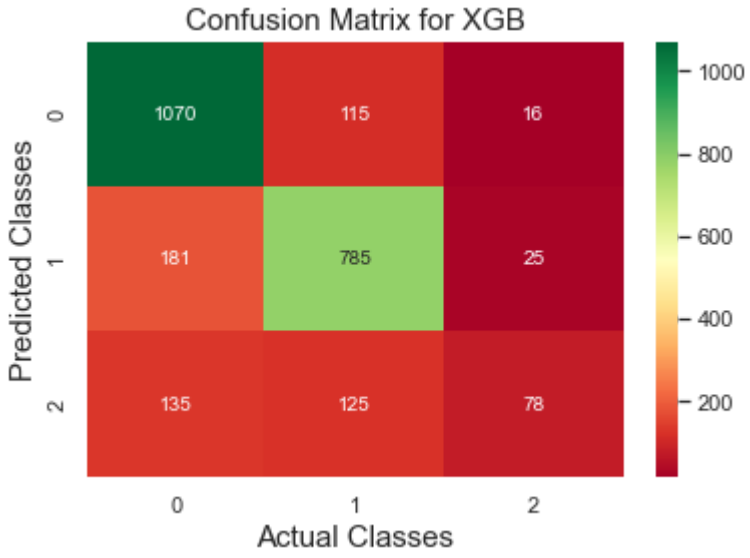
In [74]:
```
y_predictxg= xgb.predict(X_test)
modelxg_score = accuracy_score(Y_test, y_predictxg)
print('Accuracy Score of Test Data:', modelxg_score)
```

```
Accuracy Score of Test Data: 0.7640316205533597
```

In [75]:
```
#printing classification report
print("Classification Report")
print(metrics.classification_report(Y_test, y_predictxg, labels=[0,1,2]))
```

```
Classification Report
              precision    recall  f1-score   support

           0       0.77      0.89      0.83      1201
           1       0.77      0.79      0.78       991
           2       0.66      0.23      0.34       338

    accuracy                           0.76      2530
   macro avg       0.73      0.64      0.65      2530
weighted avg       0.75      0.76      0.74      2530
```

In [76]:
```
# visualizing confusion matrix
cm= confusion_matrix(Y_test, y_predictxg)
plt.figure(figsize = (6, 4))
sns.heatmap(cm, annot = True, cmap = 'RdYlGn', fmt = 'd')
plt.xlabel('Actual Classes', fontsize = 15)
plt.ylabel('Predicted Classes', fontsize = 15)
plt.title('Confusion Matrix for XGB', fontsize = 15);
```



Based on the word cloud we do notice that there is bit of an overlap between group L1 and L3, hence collapsing the targets further into 2 classes

In [77]:
```
def get_cat1(GRP):

    if GRP in [1]:
        return 1
    else:
        return 0
```

In [78]:
```
dd=data.copy()
```

In [79]:
```
dd['Label_grp'] = dd['Label_grp'].apply(get_cat1)
```

In [80]:
```
dd['Label_grp'].value_counts(normalize=True) * 100
```

Out[80]:
```
0    59.452087
1    40.547913
Name: Label_grp, dtype: float64
```

In [81]:
```
import pandas as pd
from sklearn.feature_extraction.text import CountVectorizer
```

In [82]:
```
dd.head()
```

Out[82]:

	caller	Combine_Descrption	Label_grp
0	0	login issueverify user detail name check ad re...	0
1	1	outlookteam skype appear outlook calendar some...	0
2	1	ca nt log vpnlog vpn	0
3	1	unable access hr tool pageunable page	0
4	1	skype errorskype error	0

In [83]:
```
vec = CountVectorizer(tokenizer=mytokenizer, min_df=0.005)
X = vec.fit_transform(dd['Combine_Descrption'])
df = pd.DataFrame(X.toarray(), columns=vec.get_feature_names())
```

In [84]:
```
s1 = pd.Series(dd['Label_grp'], name="Label_grp")
s2= pd.Series(dd['caller'], name="caller")
```

In [85]:
```
df1=df.reset_index(drop=True)
s3=s1.reset_index(drop=True)
s4=s2.reset_index(drop=True)
```

In [86]:
```
result = pd.concat([df1, s3,s4], axis=1)
```

In [87]:
```
result.head()
```

Out[87]:

	2016	abende	able	access	account	action	active	add	additional	address	...	without	wle	work	wor
0	0	0	1	0	0	0	0	0	0	0	...	0	0	0	
1	0	0	0	0	0	0	0	0	0	0	...	0	0	0	
2	0	0	0	0	0	0	0	0	0	0	...	0	0	0	
3	0	0	0	1	0	0	0	0	0	0	...	0	0	0	
4	0	0	0	0	0	0	0	0	0	0	...	0	0	0	

5 rows × 427 columns

In [88]:
```
result = result.loc[:,~result.columns.duplicated()]
```

In [89]:
```
result.head()
```

Out[89]:

	2016	abende	able	access	account	action	active	add	additional	address	...	window	without	wle	w
0	0	0	1	0	0	0	0	0	0	0	...	0	0	0	
1	0	0	0	0	0	0	0	0	0	0	...	0	0	0	
2	0	0	0	0	0	0	0	0	0	0	...	0	0	0	
3	0	0	0	1	0	0	0	0	0	0	...	0	0	0	
4	0	0	0	0	0	0	0	0	0	0	...	0	0	0	

5 rows × 426 columns

In [90]:
```
#creating a copy of the train data and seperating the target column and the predictor variables

X=result.drop(['Label_grp'],axis=1)
y=result['Label_grp']
```

In [91]:
```
# splitting data training dataset into train and test set for independent attributes
X_train, X_test, Y_train, Y_test =train_test_split(X,y, test_size=.30,random_state=12)
```

In [92]:
```
# Initializaing various classification algorithms with normal dataset and choosing the best mod

models = []
models.append(("LR", LogisticRegression()))
models.append(("KNN", KNeighborsClassifier()))
models.append(("GNB", GaussianNB()))
models.append(("SVM", SVC(kernel='linear',probability=True)))
models.append(("DT", DecisionTreeClassifier()))
models.append(("RF", RandomForestClassifier()))
models.append(("GBT", GradientBoostingClassifier()))
models.append(("XGB", XGBClassifier(verbosity=0)))
models.append(("LightGBM",LGBMClassifier()))

#testing models
results = []
names = []

for name, model in models:
    kfold = StratifiedKFold(n_splits=10, random_state=None,shuffle=True)
    cv_results = cross_val_score(model, X_train, Y_train, cv=kfold,scoring='roc_auc')
    results.append(cv_results)
```

```
    names.append(name)
    msg = '%s: %f%% (%f%%)' % (name, cv_results.mean()*100, cv_results.std()*100)
    print(msg)
```

```
LR: 88.448155% (1.250043%)
KNN: 84.695776% (2.007108%)
GNB: 84.779190% (1.925514%)
SVM: 86.350009% (2.227272%)
DT: 76.121059% (2.070598%)
RF: 90.490668% (0.905004%)
GBT: 88.714790% (1.090906%)
XGB: 89.995563% (1.472594%)
LightGBM: 90.197015% (1.267628%)
```

In [93]:
```
# Number of trees in random forest
n_estimators = [int(x) for x in np.linspace(start = 50, stop = 500, num = 50)]
# Number of features to consider at every split
max_features = ['auto', 'sqrt']
# Maximum number of levels in tree
max_depth = [int(x) for x in np.linspace(10, 110, num = 11)]
max_depth.append(None)
# Minimum number of samples required to split a node
min_samples_split = range(2,100,5)
# Minimum number of samples required at each leaf node
min_samples_leaf = range(1,100,10)
# Method of selecting samples for training each tree
bootstrap = [True, False]

# Create the random grid
random_grid = {'n_estimators': n_estimators,
               'max_features': max_features,
               'max_depth': max_depth,
               'min_samples_split': min_samples_split,
               'min_samples_leaf': min_samples_leaf,
               'bootstrap': bootstrap,
                'criterion':['gini','entropy']}
```

In [94]:
```
rf = RandomForestClassifier()
rf_random = RandomizedSearchCV(estimator = rf, param_distributions = random_grid, cv = 5, verbo
rf_random.fit(X_train, Y_train)
rf_random.best_params_
```

```
Fitting 5 folds for each of 10 candidates, totalling 50 fits
[Parallel(n_jobs=-1)]: Using backend LokyBackend with 16 concurrent workers.
[Parallel(n_jobs=-1)]: Done   9 tasks      | elapsed:    8.6s
[Parallel(n_jobs=-1)]: Done  45 out of  50 | elapsed:   29.3s remaining:    3.2s
[Parallel(n_jobs=-1)]: Done  50 out of  50 | elapsed:   35.1s finished
```

Out[94]:
```
{'n_estimators': 463,
 'min_samples_split': 82,
 'min_samples_leaf': 1,
 'max_features': 'sqrt',
 'max_depth': 110,
 'criterion': 'gini',
 'bootstrap': False}
```

In [95]:
```
rf_grid1 = RandomForestClassifier(n_estimators=463,
 min_samples_split= 82,
 min_samples_leaf=1,
 max_features= 'sqrt',
 max_depth= 110,
 criterion= 'gini',
 bootstrap= False)
rf_grid1.fit(X_train, Y_train)
```

Out[95]:
```
RandomForestClassifier(bootstrap=False, max_depth=110, max_features='sqrt',
                       min_samples_split=82, n_estimators=463)
```

In [96]:
```
modelrfg1_score=rf_grid1.score(X_train,Y_train)
print('Accuracy Score of Training Data: ', modelrfg1_score)
```

```
Accuracy Score of Training Data:  0.9039308708912233
```

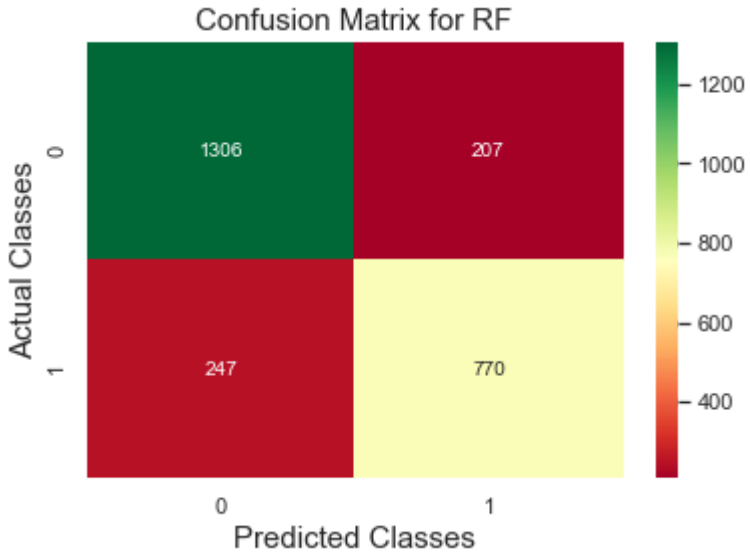
In [97]:
```
y_predictrfg1= rf_grid1.predict(X_test)
modelrfg1_score = accuracy_score(Y_test, y_predictrfg1)
print('Accuracy Score of Test Data:', modelrfg1_score)
```

```
Accuracy Score of Test Data: 0.8205533596837945
```

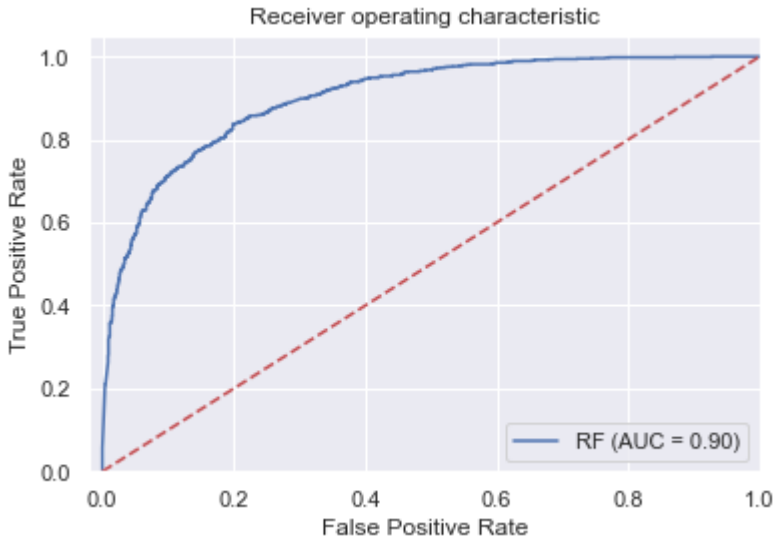
In [98]:
```
#printing classification report
print("Classification Report")
print(metrics.classification_report(Y_test, y_predictrfg1, labels=[0,1]))
```

```
Classification Report
              precision    recall  f1-score   support

           0       0.84      0.86      0.85      1513
           1       0.79      0.76      0.77      1017

    accuracy                           0.82      2530
   macro avg       0.81      0.81      0.81      2530
weighted avg       0.82      0.82      0.82      2530
```

In [99]:
```
# visualizing confusion matrix
cm= confusion_matrix(Y_test, y_predictrfg1)
plt.figure(figsize = (6, 4))
sns.heatmap(cm, annot = True, cmap = 'RdYlGn', fmt = 'd')
plt.ylabel('Actual Classes', fontsize = 15)
plt.xlabel('Predicted Classes', fontsize = 15)
plt.title('Confusion Matrix for RF', fontsize = 15);
```



In [105…
```
#Plotting ROC and AUC
probs = rf_grid1.predict_proba(X_test)
preds = probs[:,1]
fpr, tpr, threshold = metrics.roc_curve(Y_test, preds)
roc_auc_rfo = metrics.auc(fpr, tpr)
plt.figure(figsize=(6, 4))
plt.plot(fpr, tpr, label='RF (AUC = %0.2f)' % roc_auc_rfo)
plt.plot([0, 1], [0, 1],'r--')
plt.xlim([-0.02, 1.0])
plt.ylim([0.0, 1.05])
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('Receiver operating characteristic')
plt.legend(loc="lower right")
plt.show()
```



In [101…
```
i = np.arange(len(tpr)) # index for df
roc = pd.DataFrame({'fpr' : pd.Series(fpr, index=i),'tpr' : pd.Series(tpr, index = i), '1-fpr'
print(roc.loc[(roc.tf-0).abs().argsort()[:1]])
```

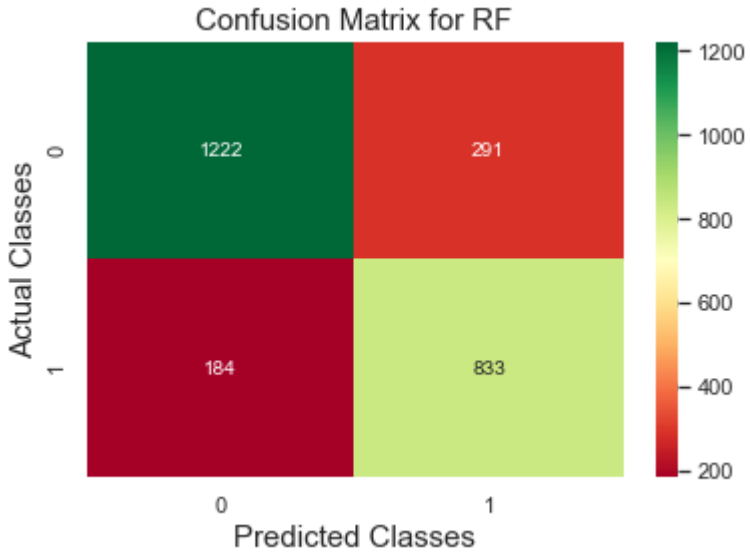
```
          fpr       tpr     1-fpr        tf  threshold
362  0.187707  0.813176  0.812293  0.000883   0.420479
```

In [102…
```
# store the predicted probabilities  for failed class
y_pred_prob = rf_grid1.predict_proba(X_test)[:, 1]
# predict fail if the predicted probability is greater than 0.43
from sklearn.preprocessing import binarize
y_pred_class = binarize([y_pred_prob], 0.41)[0]
```

In [103…
```
#printing classification report
print("Classification Report")
print(metrics.classification_report(Y_test, y_pred_class, labels=[0, 1]))
```

```
Classification Report
              precision    recall  f1-score   support

           0       0.87      0.81      0.84      1513
           1       0.74      0.82      0.78      1017

    accuracy                           0.81      2530
   macro avg       0.81      0.81      0.81      2530
weighted avg       0.82      0.81      0.81      2530
```

In [104…
```
# visualizing confusion matrix
cm= confusion_matrix(Y_test, y_pred_class)
plt.figure(figsize = (6, 4))
sns.heatmap(cm, annot = True, cmap = 'RdYlGn', fmt = 'd')
plt.ylabel('Actual Classes', fontsize = 15)
plt.xlabel('Predicted Classes', fontsize = 15)
plt.title('Confusion Matrix for RF', fontsize = 15);
```



- Conclusion: The best performing model when we consider 3 or 2 unique groups appears to be Random forest based on test accuracy

- **OBJECTIVE:**

Use the text data to build simple feed-forward Neural Nets and benchmark against the base ML models.

In [1]:
```
# imports

import os
import math
import random
import warnings
from time import time
from pathlib import Path
import pandas as pd, numpy as np
from pprint import pprint
import matplotlib.pyplot as plt
import seaborn as sns
from tqdm import tqdm
from collections import defaultdict, Counter
from sklearn.preprocessing import LabelEncoder
from wordcloud import WordCloud, STOPWORDS
import tensorflow

tqdm.pandas()
warnings.filterwarnings('ignore')
warnings.simplefilter(action='ignore', category=FutureWarning)
%matplotlib inline
```

```
C:\Users\surya\anaconda3\envs\full\lib\site-packages\tqdm\std.py:697: FutureWarning: The Panel
class is removed from pandas. Accessing it from the top-level namespace will also be removed in
the next version
  from pandas import Panel
```

In [2]:
```
# reproducibility
seed = 7
random.seed(seed)
tensorflow.random.set_seed(seed)
```

- **Import & Analyse the data.**

In [3]:
```
dataset = pd.read_excel('./data/cleaned_data.xlsx')
dataset.sample(10)
```

Out[3]:

	short_description	description	caller	group	char_length	word_length	short_cl
1867	set abc code erp 81807016 under me in crm.	set abc code erp 81807016 under me in crm. in ...	srhoeyza rkhuldgq	GRP_40	56	13	
3587	vip 1: please add me to the allowed sender lis...	please add me to the allowed sender list for t...	hkrecpfv kgwpbexv	GRP_26	66	13	
6957	business_client SID_1 search_server not working	hi, i can find documents in engineering tool S...	aqourvgz mkehgcdu	GRP_14	238	39	
4623	no response from other side.	no response from other side.	efbwiadp dicafxhv	GRP_0	28	5	
5862	wireless outage again-taiwan 0830	\n\nreceived from: ticqvhal.vgokzesi@gmail.com...	ticqvhal vgokzesi	GRP_4	110	10	
3595	computer crashed after a reboot	computer crashed after a reboot	zstkagwu jlyrhdcf	GRP_0	31	5	
2439	summary:when i run wip list and try to do my r...	summary:when i run wip list and try to do my r...	gdpxqyhj iapghvke	GRP_0	111	22	
3892	user unable tologin to vpn.	name:lizhwdoe mjudivse\nlanguage:\nbrowser:mic...	lizhwdoe mjudivse	GRP_0	160	15	
676	account unlock	account unlock	eboutzmn umzvbkfh	GRP_0	14	2	
2096	need access to folder	\n\nreceived from: umdyvbxo.qwzstijr@gmail.com...	umdyvbxo qwzstijr	GRP_12	113	15	

In [4]:
```
dataset.isna().sum()
```

Out[4]:
```
short_description              0
description                    0
caller                         0
group                          0
char_length                    0
word_length                    0
short_char_length              0
short_word_length              0
description_keywords           7
short_description_keywords    38
group_code                     0
char_length_bins               0
cleaned_description            0
cleaned_short_description      0
cleaned_char_length            0
cleaned_word_length            0
cleaned_short_char_length      0
cleaned_short_word_length      0
dtype: int64
```

In [5]:
```
dataset[dataset.isna().any(axis=1)].to_csv('./data/missing_keywords.csv')
dataset[dataset.isna().any(axis=1)]  # check rows with missing values
```

Out[5]:

	short_description	description	caller	group	char_length	word_leng
483	: k-bngell-cgdaytshqsd <k-bngell-cgdaytshqsd@c...	\n\nreceived from: lvxakohq.tsfnhowj@gmail.com...	lvxakohq tsfnhowj	GRP_19	210	

	short_description	description	caller	group	char_length	word_leng
708	id 04637	id 04637 printer have paper stuck up issue.	ongumpdz pjkrfmbc	GRP_19	43	
755	pc name	\n\nreceived from: koahsriq.wdugqatr@gmail.com...	koahsriq wdugqatr	GRP_28	323	
1114	id : 1064870825	id : 2175981936\n\nperson on other side discon...	efbwiadp dicafxhv	GRP_0	51	
1294	dn 9169508476,t/o 642392	\n\nreceived from: gjtyswkb.dpvaymxr@gmail.com...	gjtyswkb dpvaymxr	GRP_6	129	
1331	apac, company: multiple switches went down at ...	company-ap-chn-apac-company-fpsf-2960s-access-...	mnlazfsr mtqrkhnx	GRP_8	150	
1641	need help	\n\nreceived from: axcbfuqo.yiagubvh@gmail.com...	axcbfuqo yiagubvh	GRP_0	376	
1823	bgflmyar.xgufkidq@gmail.com wanted to check if...	bgflmyar.xgufkidq@gmail.com wanted to check if...	olckhmvx pcqobjnd	GRP_0	83	
2396	need your help!!	\n\nreceived from: ezwcpqrh.bnwqaglk@gmail.com...	ezwcpqrh bnwqaglk	GRP_0	571	
2436	changes in ad	hi, there,\nmy reporting line in the outlook o...	ywbnzxud qzwrynux	GRP_2	246	
2736	cann't do "mb31" for po115890552	there is a po 226901663 in plant_282.\nnow,we...	jerydwbn gdylnaue	GRP_45	100	
2952	\n\nreceived from: yzbjhmpw.vzrulkog@gmail.com...	\n\nreceived from: yzbjhmpw.vzrulkog@gmail.com...	yzbjhmpw vzrulkog	GRP_0	1167	1
2957	hp2热压炉数据传输卡,数据更新不出来,请帮我转给小贺	hp2热压炉数据传输卡,数据更新不出来,请帮我转给小贺	basqoyjx frvwhbse	GRP_30	27	
3215	help	\n\nreceived from: lanigpkq.qzhakunx@gmail.com...	lanigpkq qzhakunx	GRP_33	255	
3296	it help	\n\nreceived from: notwkdgr.zvmesjpt@gmail.com...	notwkdgr zvmesjpt	GRP_26	7467	4
3391	re: need a little help--please	\n\nreceived from: bcefayom.lzhwcgvb@gmail.com...	bcefayom lzhwcgvb	GRP_18	728	1
3392	re: need a little help--please	\n\nreceived from: smxoklny.hbecskgl@gmail.com...	khvzugxm yqfrcjwl	GRP_18	334	
3500	PR	create a purchase requisition with purchasing ...	ejvkzobl yijgokrn	GRP_29	198	
3509	lcowx216132	\n\nreceived from: zwirhcol.narzlmfw@gmail.com...	zwirhcol narzlmfw	GRP_0	204	
3510	lcow7404551	\n\nreceived from: zwirhcol.narzlmfw@gmail.com...	zwirhcol narzlmfw	GRP_0	234	

	short_description	description	caller	group	char_length	word_leng
3615	re: need a little help--please	\n\nreceived from: damuphws.arkulcoi@gmail.com...	damuphws arkulcoi	GRP_18	478	
3620	re: need a little help--please	\n\nreceived from: smxoklny.hbecskgl@gmail.com...	khvzugxm yqfrcjwl	GRP_18	131	
3681	mm#3342477 | mm#5270584 | mm#5270486 | mm#4166346	hi\n\nplease see below pricing team comments a...	kfhnmtgi boxmklnp	GRP_13	96	
3689	re: need a little help--please	\n\nreceived from: bcefayom.lzhwcgvb@gmail.com...	bcefayom lzhwcgvb	GRP_18	2292	3
4529	i am not able to connect to my regular printer...	x5380	koiapqbg teyldpkw	GRP_0	6	
4781	chg0034110	\n\nreceived from: afkstcev.utbnkyop@gmail.com...	afkstcev utbnkyop	GRP_0	226	
4802	ltcl8513156 - hgmx5q1 - e6420	rarty has this old laptop that he needs to log...	csmhykge mpxbjudw	GRP_3	399	
5040	hr_tool etime will not run after update ran la...	immediate need	nrmjhuox ktuyqewp	GRP_3	14	
5064	it help	\n\nreceived from: scjxobhd.ldypjkmf@gmail.com...	scjxobhd ldypjkmf	GRP_28	321	
5283	mm# 5260903 (kr230)	\n\nreceived from: hmjdrvpb.komuaywn@gmail.com...	hmjdrvpb komuaywn	GRP_29	171	
5416	答复: 35969737/2032252	\n\nreceived from: wqzarvhx.hfsojckw@gmail.com...	wqzarvhx hfsojckw	GRP_13	174	
5493	po - a4 4505633620	hello it,\n\nthere are 3 item linked with the...	bejcxvis anxmhwis	GRP_29	131	
5783	new cpp id can not request initiative. see im...	cphlme01\n	pfzxecbo ptygkvzl	GRP_21	9	
5791	s&op	\n\nreceived from: uyrpdvoq.mbzevtcx@gmail.com...	uyrpdvoq mbzevtcx	GRP_0	328	
5979	it help	\n\nreceived from: scjxobhd.ldypjkmf@gmail.com...	scjxobhd ldypjkmf	GRP_28	156	
5982	following up	hello it,\n\nplease can you block this email a...	pzybmcqd fxtemlyg	GRP_0	86	
6200	in the inbox always show there are several ema...	in the inbox always show there are several ema...	mqbxwpfn uclrqfxa	GRP_0	94	
6254	i have created 2 new material numbers but when...	mm#'s 7390081 and 6290061	xplwmiyr pifoldxr	GRP_29	25	
6610	awyw7217971	\n\nreceived from: utgszjrf.pacfvxzk@gmail.com...	utgszjrf pacfvxzk	GRP_19	243	

	short_description	description	caller	group	char_length	word_leng
6701	will not come up	not showixepyfbga wtqdyoin drive at all	hdfcwmag plxstkad	GRP_3	39	
7046	please help	\n\nreceived from: iqmhjlwr.jqmxaybi@gmail.com...	iqmhjlwr jqmxaybi	GRP_0	195	
7379	答复: help for mm#4866474 24800776	\n\nreceived from: windy.shi@company.com\n\nde...	tycludks cjofwigv	GRP_6	108	
7405	mm# 1876905	from: -kds sw11-services \nsent: tuesday, augu...	rxoynvgi ntgdsehl	GRP_6	243	
7657	can you please help	\n\nreceived from: smktofel.etsoirbw@gmail.com...	smktofel etsoirbw	GRP_55	216	

In [6]:
```
le = LabelEncoder()
dataset['group_code'] = le.fit_transform(dataset.group)
dataset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8432 entries, 0 to 8431
Data columns (total 18 columns):
 #   Column                      Non-Null Count  Dtype 
---  ------                      --------------  ----- 
 0   short_description           8432 non-null   object
 1   description                 8432 non-null   object
 2   caller                      8432 non-null   object
 3   group                       8432 non-null   object
 4   char_length                 8432 non-null   int64 
 5   word_length                 8432 non-null   int64 
 6   short_char_length           8432 non-null   int64 
 7   short_word_length           8432 non-null   int64 
 8   description_keywords        8425 non-null   object
 9   short_description_keywords  8394 non-null   object
 10  group_code                  8432 non-null   int32 
 11  char_length_bins            8432 non-null   int64 
 12  cleaned_description         8432 non-null   object
 13  cleaned_short_description   8432 non-null   object
 14  cleaned_char_length         8432 non-null   int64 
 15  cleaned_word_length         8432 non-null   int64 
 16  cleaned_short_char_length   8432 non-null   int64 
 17  cleaned_short_word_length   8432 non-null   int64 
dtypes: int32(1), int64(9), object(8)
memory usage: 1.1+ MB
```

In [7]:
```
le.classes_
```

Out[7]:
```
array(['GRP_0', 'GRP_1', 'GRP_10', 'GRP_11', 'GRP_12', 'GRP_13', 'GRP_14',
       'GRP_15', 'GRP_16', 'GRP_17', 'GRP_18', 'GRP_19', 'GRP_2',
       'GRP_20', 'GRP_21', 'GRP_22', 'GRP_23', 'GRP_24', 'GRP_25',
       'GRP_26', 'GRP_27', 'GRP_28', 'GRP_29', 'GRP_3', 'GRP_30',
       'GRP_31', 'GRP_32', 'GRP_33', 'GRP_34', 'GRP_35', 'GRP_36',
       'GRP_37', 'GRP_38', 'GRP_39', 'GRP_4', 'GRP_40', 'GRP_41',
       'GRP_42', 'GRP_43', 'GRP_44', 'GRP_45', 'GRP_46', 'GRP_47',
       'GRP_48', 'GRP_49', 'GRP_5', 'GRP_50', 'GRP_51', 'GRP_52',
       'GRP_53', 'GRP_54', 'GRP_55', 'GRP_56', 'GRP_57', 'GRP_58',
       'GRP_59', 'GRP_6', 'GRP_60', 'GRP_61', 'GRP_62', 'GRP_63',
       'GRP_64', 'GRP_65', 'GRP_66', 'GRP_67', 'GRP_68', 'GRP_69',
       'GRP_7', 'GRP_70', 'GRP_71', 'GRP_72', 'GRP_73', 'GRP_8', 'GRP_9'],
      dtype=object)
```

In [8]:
```
def merge_descriptions(row):
    merged_descr = np.nan
    if (row.cleaned_short_description == row.cleaned_description or
        str(row.description).startswith(str(row.cleaned_short_description))):
```

```
        merged_descr = str(row.cleaned_description)
    else:
        merged_descr = str(row.cleaned_short_description) + " " + str(row.cleaned_description)
    row['merged_description'] = str(merged_descr)
    return row

dataset = dataset.progress_apply(merge_descriptions, axis=1)
```

```
100%|██████████| 8432/8432 [00:08<00:00, 975.45it/s]
```

In [9]:
```
dataset[['cleaned_short_description', 'cleaned_description', 'merged_description']].sample(10)
```

Out[9]:

	cleaned_short_description	cleaned_description	merged_description
2192	kein datenabgleich zwischen eu tool und erp ge...	kein datenabgleich zwischen eu tool und erp ge...	kein datenabgleich zwischen eu tool und erp ge...
3095	business client work	unable access business client open business cl...	business client work unable access business cl...
458	job job 593 fail job scheduler 05 07 00	job job 593 fail job scheduler 05 07 00	job job 593 fail job scheduler 05 07 00
693	hostname 1325 drive flasng yellow message disp...	check hostname 1325 shop floor app server driv...	hostname 1325 drive flasng yellow message disp...
5953	login help hub	login help hub	login help hub
2405	prognose crm forecast plan dashbankrd work res...	dear one user affect get sale rep indicate ca ...	prognose crm forecast plan dashbankrd work res...
4868	job job 2555 fail job scheduler 22 00 00	job job 2555 fail job scheduler 22 00 00	job job 2555 fail job scheduler 22 00 00
745	reinstall hardcopy und eu tool lndypaqg	reinstall hardcopy und eu tool lndypaqg	reinstall hardcopy und eu tool lndypaqg
488	login issue	login issue verify user detail name check user...	login issue verify user detail name check user...
6663	probleme mit vpn client	hallo meine herren ich kann unsere vpn nicht b...	probleme mit vpn client hallo meine herren ich...

In [10]:
```
X = np.array(dataset.merged_description)
y = np.array(dataset.group_code)
X.shape, y.shape
```

Out[10]:
```
((8432,), (8432,))
```

In [11]:
```
from tensorflow.keras.utils import to_categorical
y_dummy_coded = to_categorical(y)
y[0], y_dummy_coded[0]
```

Out[11]:
```
(0,
 array([1., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
        0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
        0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
        0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
        0., 0., 0., 0., 0., 0.], dtype=float32))
```

In [12]:
```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y_dummy_coded, test_size=.2, random_stat
```

In [13]:
```
X_train.shape, X_test.shape, y_train.shape, y_test.shape
```

Out[13]:
```
((6745,), (1687,), (6745, 74), (1687, 74))
```

In [14]:
```
X_train[0], y_train[0]  # check sample
```

Out[14]:
```
('additional correction sale org 1278 company address phone number require germany move 1 sale
organisation address 1278 phone number fax number need reverse original phone fax number furth
0911 2 plant address plant 124 phone fax number need adjusted show germany central phone number
fax 3 company code address 5278 need revert back address furth germany detail attach ticket',
 array([0., 0., 0., 0., 0., 1., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
        0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
        0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
        0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
        0., 0., 0., 0., 0., 0.], dtype=float32))
```

In [15]:
```
# TODO: Check the distributions of groups in training and testing sets, i.e, if they vary too m
# stratify by y if required during splits
# or data augmentation to upsample minority classes to balance the group distributions
```

- **Tokenize and pad sequences**

In [16]:
```
# define params
NUM_WORDS = 20000
EMBEDDING_DIM = 300
MAX_LEN = 100 # dataset['word_length'].max()
MAX_LEN
```

Out[16]:
```
100
```

In [17]:
```
from tensorflow.keras.preprocessing.text import Tokenizer
from tensorflow.keras.preprocessing.sequence import pad_sequences


tokenizer = Tokenizer(num_words=NUM_WORDS)
tokenizer.fit_on_texts(X_train)
X_train_tokens = tokenizer.texts_to_sequences(X_train)
X_test_tokens = tokenizer.texts_to_sequences(X_test)
X_train_tokens[0], X_test_tokens[0]
```

Out[17]:
```
([184,
  2093,
  93,
  606,
  1609,
  10,
  148,
  45,
  116,
  169,
  146,
  375,
  15,
  93,
  6817,
  148,
  1609,
  45,
  116,
  1513,
  116,
  23,
  1729,
  674,
  45,
  1513,
  116,
  804,
  6818,
  30,
  75,
  148,
  75,
  2370,
  45,
  1513,
```

```
 116,
 23,
 4267,
 90,
 146,
 3354,
 45,
 116,
 1513,
 73,
 10,
 188,
 148,
 6819,
 23,
 1195,
 174,
 148,
 804,
 146,
 117,
 88,
 17],
[93, 2095, 280, 1029, 783, 355, 9, 2095, 1029, 1360, 2095, 211, 280])
```

In [18]:
```
y_train[0], y_test[0]
```

Out[18]:
```
(array([0., 0., 0., 0., 0., 1., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
        0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
        0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
        0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
        0., 0., 0., 0., 0., 0.], dtype=float32),
 array([0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
        0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
        0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
        0., 0., 0., 0., 0., 1., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
        0., 0., 0., 0., 0., 0.], dtype=float32))
```

In [19]:
```
# pad sequences to cut longer texts to a uniform length and pad the sentences that are shorter

# using just 20 words from each headline will severely limit the information that is
# available to the model and affect performance although the training will be faster
X_train_padded = pad_sequences(X_train_tokens,
                               padding='post',
                               truncating='post',
                               maxlen=MAX_LEN)
X_test_padded = pad_sequences(X_test_tokens,
                              padding='post',
                              truncating='post',
                              maxlen=MAX_LEN)


print(f'X train:  {X_train_padded.shape}\nX test:   {X_test_padded.shape}')
```

```
X train:  (6745, 100)
X test:   (1687, 100)
```

In [20]:
```
pprint(X_train_padded[0], compact=True)
```

```
array([ 184, 2093,   93,  606, 1609,   10,  148,   45,  116,  169,  146,
        375,   15,   93, 6817,  148, 1609,   45,  116, 1513,  116,   23,
       1729,  674,   45, 1513,  116,  804, 6818,   30,   75,  148,   75,
       2370,   45, 1513,  116,   23, 4267,   90,  146, 3354,   45,  116,
       1513,   73,   10,  188,  148, 6819,   23, 1195,  174,  148,  804,
        146,  117,   88,   17,    0,    0,    0,    0,    0,    0,    0,
          0,    0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
          0,    0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
          0,    0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
          0])
```

In [21]:
```
WORD_TO_INDEX = tokenizer.word_index
# pprint(WORD_TO_INDEX, compact=True)
pprint(list(WORD_TO_INDEX.keys())[:100], compact=True)
```

```
['job', 'yes', 'na', 'password', 'erp', 'tool', 'user', 'ts', 'issue',
 'company', 'sid', 'reset', 'access', 'scheduler', '1', '00', 'ticket',
 'unable', 'work', 'error', 'fail', 'account', 'need', 'email', 'site', 'help',
 'system', 'hostname', 'get', '2', 'login', 'circuit', 'power', 'outlook',
 'network', 'use', 'vendor', 'change', '34', 'update', 'name', 'message',
 'backup', 'see', 'phone', 'telecom', 'server', 'try', '10', 'able', 'outage',
 'log', 'check', 'new', 'problem', 'start', 'crm', 'engineering', 'request',
 'connect', 'call', 'usa', 'type', 'time', 'printer', 'order', 'report', 'vpn',
 'team', 'open', 'contact', 'skype', '3', 'lock', 'plant', 'et', 't', 'send',
 'create', '4', '5', 'window', 'file', 'pc', 'since', 'print', 'schedule',
 'attach', 'device', 'show', '8', 'maintenance', 'sale', '11', '12', 'receive',
 'abende', 'notify', '23', 'management']
```

In [22]:
```
VOCAB_SIZE = len(WORD_TO_INDEX) + 1
VOCAB_SIZE
```

Out[22]:
```
13790
```

In [23]:
```
# https://www.tensorflow.org/api_docs/python/tf/keras/preprocessing/sequence/pad_sequences
def retrieve_description_feat(x, mapping=WORD_TO_INDEX) -> str:
    # increment 3
    mapping = {k:(v + 3) for k, v in mapping.items()}
    mapping['<PAD>'] = 0
    mapping['<START>'] = 1
    mapping['<UNK>'] = 2
    inv_mapping = {v: k for k, v in mapping.items()}
    return str(" ".join(inv_mapping.get(i, '<NA>') for i in x))

retrieve_description_feat(X_test_padded[7])
```

Out[23]:
```
'1 22 team response complaint synchronize <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD>
<PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD>
<PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD>
<PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD>
<PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD>
<PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD>
<PAD> <PAD> <PAD> <PAD> <PAD>'
```

- ## GloVe Embeddings

In [24]:
```
EMBEDDING_DIM
```

Out[24]:
```
300
```

In [25]:
```
def get_embedding_matrix(embedding_dim=EMBEDDING_DIM):
    embeddings = defaultdict()
    if embedding_dim == 200:
        file_path = f'./data/glove.6B.{embedding_dim}d.txt'
    elif embedding_dim == 300:
        file_path = f'./data/glove.840B.{embedding_dim}d.txt'
    for l in open(file_path, encoding='utf-8'):
        word = l.split(" ")[0]
        embeddings[word] = np.asarray(l.split(" ")[1:], dtype='float32')

    embeddings = dict(embeddings)

    # create a weight matrix for words in training docs
    embedding_matrix = np.zeros((NUM_WORDS, embedding_dim))

    for word, idx in WORD_TO_INDEX.items():
        embedding_vector = embeddings.get(word)
        if embedding_vector is not None:
            embedding_matrix[idx] = embedding_vector

    return embedding_matrix
```

In [26]:
```
# use pre-trained glove embedding matrix to initialize weights in our model
```

```
embedding_matrix = get_embedding_matrix()
embedding_matrix.shape
```

Out[26]: (20000, 300)

- **Simple Feed-Forward Neural Net**

In [27]:
```
# !pip install livelossplot
from tensorflow.python.keras.models import Sequential
from sklearn.metrics import accuracy_score, confusion_matrix
from tensorflow.keras.regularizers import l2
from tensorflow.keras.constraints import max_norm, unit_norm
from tensorflow.python.keras.callbacks import LambdaCallback, EarlyStopping, ReduceLROnPlateau
from tensorflow.keras.layers import Flatten, Dense, Activation, BatchNormalization, Dropout, Em
```

In [28]:
```
NUM_CLASSES = len(le.classes_)
VOCAB_SIZE, MAX_LEN, EMBEDDING_DIM, NUM_CLASSES
```

Out[28]: (13790, 100, 300, 74)

In [29]:
```
# define model

model1 = Sequential([
    Embedding(input_dim=VOCAB_SIZE, output_dim=EMBEDDING_DIM, input_length=MAX_LEN),
    Flatten(),
    Dense(1024, activation = 'relu'),
    Dense(1024, activation = 'relu'),
    Dense(128, activation = 'relu'),
    Dense(NUM_CLASSES, activation = 'softmax')
])

model1.compile(
    loss = 'categorical_crossentropy',
    optimizer = 'adam',
    metrics = ['accuracy']
)
```

In [30]:
```
# Define Callbacks and a few helper functions

# simplify the training log
simple_log = LambdaCallback(
    on_epoch_end = lambda e, l: print(f" ~| Epoch: {e+1} | Validation Loss: {l['val_loss']:.5f}

# early stopping
early_stop = EarlyStopping(monitor='val_loss',
                           min_delta=0,
                           patience=7,
                           verbose=0,
                           restore_best_weights=True)

# learning rate reduction
lr_reduce_on_plateau = ReduceLROnPlateau(monitor='val_loss',
                                         patience=4,
                                         verbose=1,
                                         factor=0.4,
                                         min_lr=0.00001)

def plot_learning_curve(hist):
    sns.set()
    plt.figure(figsize=(5,5))
    train = hist.history['loss']
    val = hist.history['val_loss']
    epochs_run = range(1,len(train) + 1)
    sns.lineplot(epochs_run, train, marker = 'o', color = 'coral', label = 'Training Loss')
    sns.lineplot(epochs_run, val,  marker = '>', color = 'green', label = 'Validation Loss')
    plt.title("Loss vs. Epochs", fontsize = 20)
```

```
    plt.legend()
    plt.show()
```

In [31]:
```
X_train[0]
```

Out[31]:
```
'additional correction sale org 1278 company address phone number require germany move 1 sale o
rganisation address 1278 phone number fax number need reverse original phone fax number furth 0
911 2 plant address plant 124 phone fax number need adjusted show germany central phone number
fax 3 company code address 5278 need revert back address furth germany detail attach ticket'
```

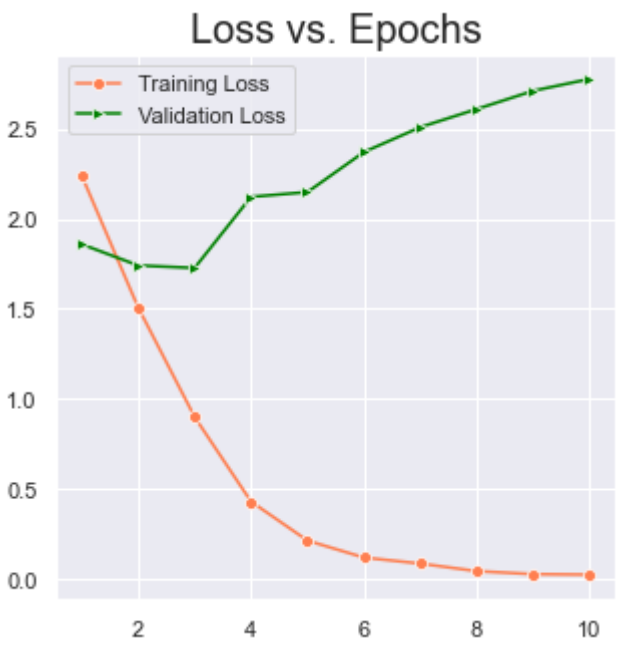
In [32]:
```
X_train.shape, y_train.shape, X_test.shape, y_test.shape
```

Out[32]:
```
((6745,), (6745, 74), (1687,), (1687, 74))
```

In [33]:
```
EPOCHS = 200
try:
    print("Training on GPU:")
    with tensorflow.device("gpu:0"):  # train on gpu
        h1 = model1.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)
except Exception as e:
    print(e)
    print("\nTraining on CPU:")
    h1 = model1.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)

print("\nTraining Done.")
```

```
Training on GPU:
 ~| Epoch: 1 | Validation Loss: 1.85944 >|>
 ~| Epoch: 2 | Validation Loss: 1.74150 >|>
 ~| Epoch: 3 | Validation Loss: 1.72742 >|>
 ~| Epoch: 4 | Validation Loss: 2.12246 >|>
 ~| Epoch: 5 | Validation Loss: 2.14895 >|>
 ~| Epoch: 6 | Validation Loss: 2.37142 >|>
 ~| Epoch: 7 | Validation Loss: 2.50962 >|>

Epoch 00007: ReduceLROnPlateau reducing learning rate to 0.0004000000189989805.
 ~| Epoch: 8 | Validation Loss: 2.61027 >|>
 ~| Epoch: 9 | Validation Loss: 2.71168 >|>
 ~| Epoch: 10 | Validation Loss: 2.77748 >|>

Training Done.
```

In [34]:
```
plot_learning_curve(h1)
```



In [35]:
```
loss, acc = model1.evaluate(X_test_padded, y_test)
print("Testing Loss: ", loss*100)
print("Testing Accuracy: ", acc*100)
```

```
53/53 [==============================] - 0s 8ms/step - loss: 1.8063 - accuracy: 0.6040
Testing Loss:  180.633807182312
Testing Accuracy:  60.40308475494385
```

- This model is clearly overfitting, we will add regularization to the next iteration

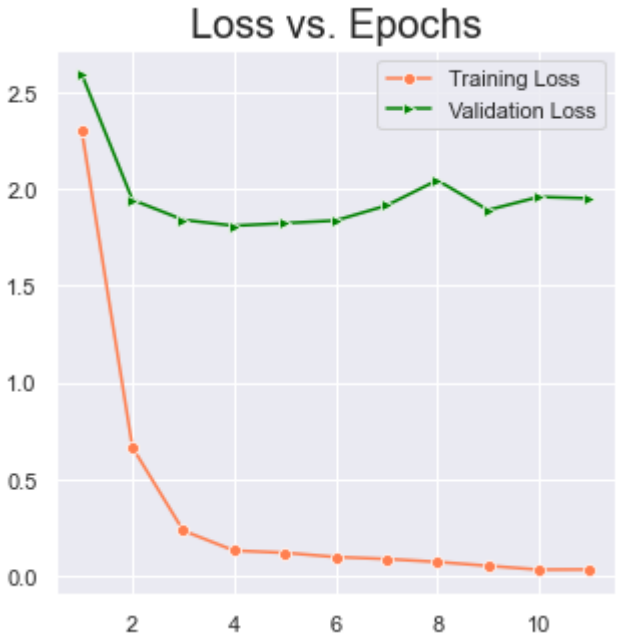
In [36]:
```
# define model

model2 = Sequential([
    Embedding(input_dim=VOCAB_SIZE, output_dim=EMBEDDING_DIM, input_length=MAX_LEN),
    Flatten(),
    Dense(256, activation = 'relu'),
    BatchNormalization(),
    Dense(256, activation = 'relu'),
    BatchNormalization(),
    Dense(NUM_CLASSES, activation = 'softmax')
])

model2.compile(
    loss = 'categorical_crossentropy',
    optimizer = 'adam',
    metrics = ['accuracy']
)
```

In [37]:
```
EPOCHS = 200
try:
    print("Training on GPU:")
    with tensorflow.device("gpu:0"):  # train on gpu
        h2 = model2.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)
except Exception as e:
    print(e)
    print("Training on CPU:")
    h2 = model2.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)
```

```
print("\nTraining Done.")
```

```
Training on GPU:
 ~| Epoch: 1 | Validation Loss: 2.59394 >|>
 ~| Epoch: 2 | Validation Loss: 1.94380 >|>
 ~| Epoch: 3 | Validation Loss: 1.84034 >|>
 ~| Epoch: 4 | Validation Loss: 1.80988 >|>
 ~| Epoch: 5 | Validation Loss: 1.82323 >|>
 ~| Epoch: 6 | Validation Loss: 1.83685 >|>
 ~| Epoch: 7 | Validation Loss: 1.91414 >|>
 ~| Epoch: 8 | Validation Loss: 2.04519 >|>

Epoch 00008: ReduceLROnPlateau reducing learning rate to 0.0004000000189989805.
 ~| Epoch: 9 | Validation Loss: 1.89107 >|>
 ~| Epoch: 10 | Validation Loss: 1.95908 >|>
 ~| Epoch: 11 | Validation Loss: 1.95138 >|>

Training Done.
```

In [38]:
```
plot_learning_curve(h2)
```



In [39]:
```
loss, acc = model2.evaluate(X_test_padded, y_test)
print("Testing Loss: ", loss*100)
print("Testing Accuracy: ", acc*100)
```

```
53/53 [==============================] - 0s 5ms/step - loss: 1.8225 - accuracy: 0.6343
Testing Loss:  182.25327730178833
Testing Accuracy:  63.42620253562927
```

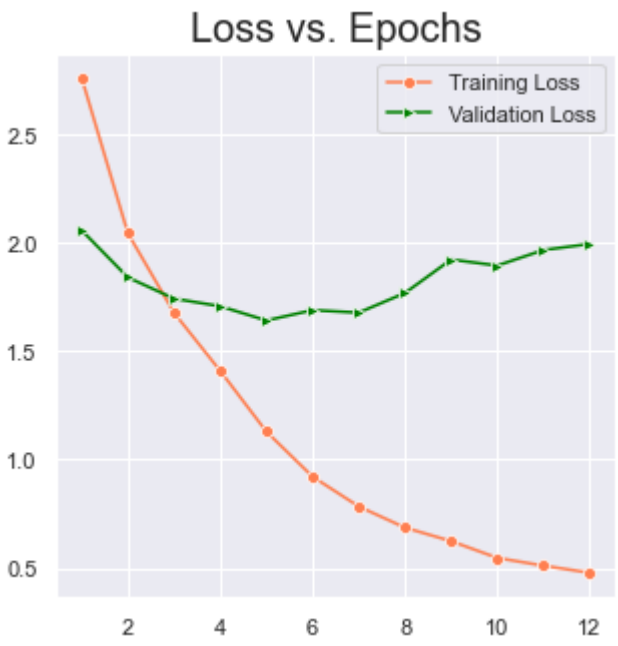
- **Add Dropout Layer**

In [40]:
```
# define model

model3 = Sequential([
    Embedding(input_dim=VOCAB_SIZE, output_dim=EMBEDDING_DIM, input_length=MAX_LEN),
    Flatten(),
    Dense(20, activation = 'relu'),
    Dropout(0.4),
    Dense(NUM_CLASSES, activation = 'softmax')
])

model3.compile(
    loss = 'categorical_crossentropy',
    optimizer = 'adam',
    metrics = ['accuracy']
)
```

In [41]:
```
EPOCHS = 200
try:
    print("Training on GPU:")
    with tensorflow.device("gpu:0"):  # train on gpu
        h3 = model3.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)
except Exception as e:
    print(e)
    print("Training on CPU:")
    h3 = model3.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)

print("\nTraining Done.")
```

```
Training on GPU:
 ~| Epoch: 1 | Validation Loss: 2.05753 >|>
 ~| Epoch: 2 | Validation Loss: 1.84023 >|>
 ~| Epoch: 3 | Validation Loss: 1.74239 >|>
 ~| Epoch: 4 | Validation Loss: 1.70713 >|>
 ~| Epoch: 5 | Validation Loss: 1.64135 >|>
 ~| Epoch: 6 | Validation Loss: 1.68905 >|>
 ~| Epoch: 7 | Validation Loss: 1.67758 >|>
 ~| Epoch: 8 | Validation Loss: 1.76820 >|>
 ~| Epoch: 9 | Validation Loss: 1.92260 >|>

Epoch 00009: ReduceLROnPlateau reducing learning rate to 0.0004000000189989805.
 ~| Epoch: 10 | Validation Loss: 1.89466 >|>
 ~| Epoch: 11 | Validation Loss: 1.96651 >|>
 ~| Epoch: 12 | Validation Loss: 1.99445 >|>

Training Done.
```

In [42]:
```
plot_learning_curve(h3)
```



In [43]:
```
loss, acc = model3.evaluate(X_test_padded, y_test)
print("Testing Accuracy: ", acc*100)
```

```
53/53 [==============================] - 0s 7ms/step - loss: 1.6926 - accuracy: 0.6473
Testing Accuracy:  64.73029255867004
```

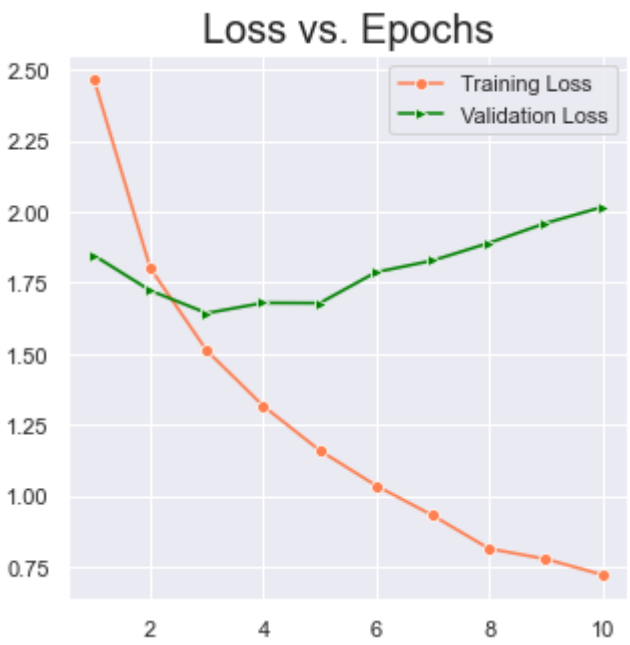
- **Use pre-trained embeddings**

In [44]:
```
# define model

model3 = Sequential([
    Embedding(input_dim=NUM_WORDS, output_dim=EMBEDDING_DIM, input_length=MAX_LEN, weights=[emb
    Flatten(),
    Dense(30, activation = 'relu'),
    Dropout(0.5),
    Dense(NUM_CLASSES, activation = 'softmax')
])

model3.compile(
    loss = 'categorical_crossentropy',
    optimizer = 'rmsprop',
    metrics = ['accuracy']
)
```

In [45]:
```
EPOCHS = 200
try:
    print("Training on GPU:")
    with tensorflow.device("gpu:0"):  # train on gpu
        h3 = model3.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)
except Exception as e:
    print(e)
    print("Training on CPU:")
    h3 = model3.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)

print("\nTraining Done.")
```

```
Training on GPU:
 ~| Epoch: 1 | Validation Loss: 1.84620 >|>
 ~| Epoch: 2 | Validation Loss: 1.72238 >|>
 ~| Epoch: 3 | Validation Loss: 1.64293 >|>
 ~| Epoch: 4 | Validation Loss: 1.67915 >|>
 ~| Epoch: 5 | Validation Loss: 1.67839 >|>
 ~| Epoch: 6 | Validation Loss: 1.78722 >|>
 ~| Epoch: 7 | Validation Loss: 1.82795 >|>

Epoch 00007: ReduceLROnPlateau reducing learning rate to 0.0004000000189989805.
 ~| Epoch: 8 | Validation Loss: 1.88995 >|>
 ~| Epoch: 9 | Validation Loss: 1.95935 >|>
 ~| Epoch: 10 | Validation Loss: 2.01581 >|>

Training Done.
```

In [46]:
```
plot_learning_curve(h3)
```



In [47]:
```
loss, acc = model3.evaluate(X_test_padded, y_test)
print("Testing Accuracy: ", acc*100)
```

```
53/53 [==============================] - 0s 7ms/step - loss: 1.7662 - accuracy: 0.6153
Testing Accuracy:  61.52934432029724
```

- **LSTM**

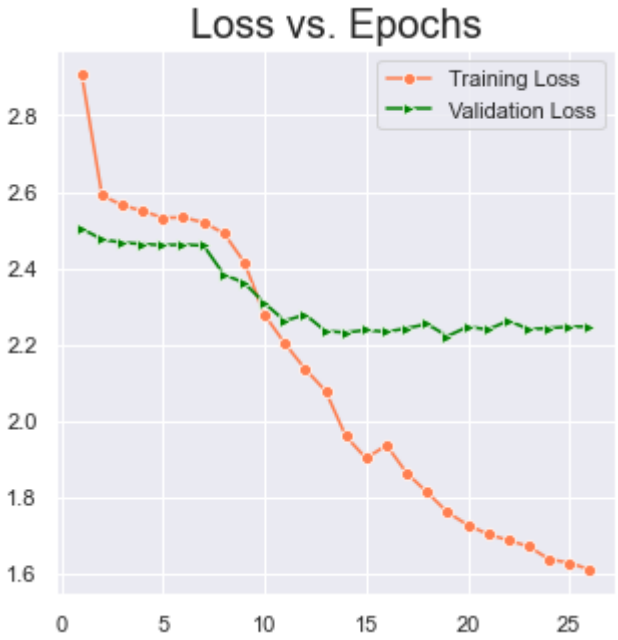
In [48]:
```
# define model

model4 = Sequential([
    Embedding(input_dim=VOCAB_SIZE, output_dim=EMBEDDING_DIM, input_length=MAX_LEN),
    LSTM(32),
    Dropout(0.4),
    Dense(NUM_CLASSES, activation = 'softmax')
])

model4.compile(
    loss = 'categorical_crossentropy',
    optimizer = 'adam',
    metrics = ['accuracy']
)
```

In [49]:
```
EPOCHS = 50
try:
    print("Training on GPU:")
    with tensorflow.device("gpu:0"):  # train on gpu
        h4 = model4.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)
except Exception as e:
    print(e)
    print("Training on CPU:")
    h4 = model4.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)

print("\nTraining Done.")
```

```
Training on GPU:
 ~| Epoch: 1 | Validation Loss: 2.50286 >|>
```

```
 ~| Epoch: 2 | Validation Loss: 2.47478 >|>
 ~| Epoch: 3 | Validation Loss: 2.46620 >|>
 ~| Epoch: 4 | Validation Loss: 2.46253 >|>
 ~| Epoch: 5 | Validation Loss: 2.45951 >|>
 ~| Epoch: 6 | Validation Loss: 2.46105 >|>
 ~| Epoch: 7 | Validation Loss: 2.45835 >|>
 ~| Epoch: 8 | Validation Loss: 2.38237 >|>
 ~| Epoch: 9 | Validation Loss: 2.36089 >|>
 ~| Epoch: 10 | Validation Loss: 2.30717 >|>
 ~| Epoch: 11 | Validation Loss: 2.26101 >|>
 ~| Epoch: 12 | Validation Loss: 2.27733 >|>
 ~| Epoch: 13 | Validation Loss: 2.23432 >|>
 ~| Epoch: 14 | Validation Loss: 2.23029 >|>
 ~| Epoch: 15 | Validation Loss: 2.23706 >|>
 ~| Epoch: 16 | Validation Loss: 2.23212 >|>
 ~| Epoch: 17 | Validation Loss: 2.24156 >|>
 ~| Epoch: 18 | Validation Loss: 2.25283 >|>

Epoch 00018: ReduceLROnPlateau reducing learning rate to 0.0004000000189989805.
 ~| Epoch: 19 | Validation Loss: 2.22072 >|>
 ~| Epoch: 20 | Validation Loss: 2.24564 >|>
 ~| Epoch: 21 | Validation Loss: 2.23823 >|>
 ~| Epoch: 22 | Validation Loss: 2.26147 >|>
 ~| Epoch: 23 | Validation Loss: 2.23908 >|>

Epoch 00023: ReduceLROnPlateau reducing learning rate to 0.00016000000759959222.
 ~| Epoch: 24 | Validation Loss: 2.24183 >|>
 ~| Epoch: 25 | Validation Loss: 2.24573 >|>
 ~| Epoch: 26 | Validation Loss: 2.24613 >|>

Training Done.
```

In [50]:
```
plot_learning_curve(h4)
```



In [51]:
```
loss, acc = model4.evaluate(X_test_padded, y_test)
print("Testing Accuracy: ", acc*100)
```

```
53/53 [==============================] - 0s 9ms/step - loss: 2.2392 - accuracy: 0.4991
Testing Accuracy:  49.91108477115631
```

- **Bi-Directional LSTM**

In [52]:
```
# define model

model4 = Sequential([
    Embedding(input_dim=VOCAB_SIZE, output_dim=EMBEDDING_DIM, input_length=MAX_LEN),
    Bidirectional(LSTM(32)),
    Dropout(0.4),
    Dense(NUM_CLASSES, activation = 'softmax')
```

```
])

model4.compile(
    loss = 'categorical_crossentropy',
    optimizer = 'rmsprop',
    metrics = ['accuracy']
)
```

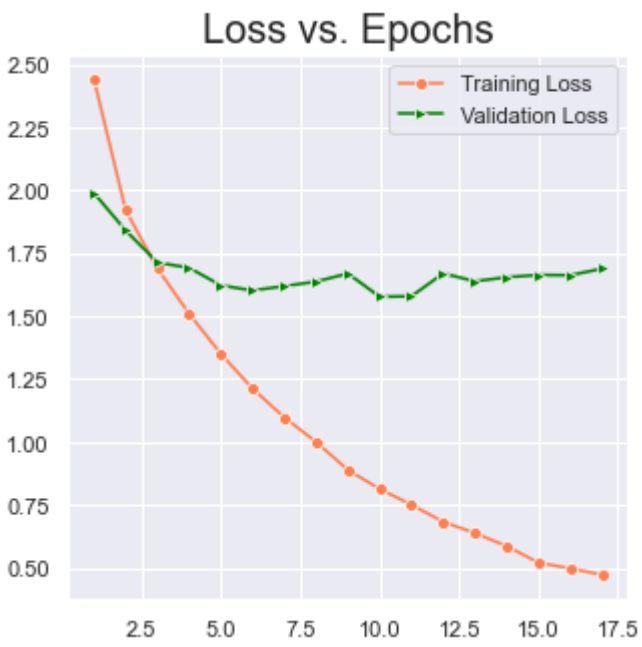
In [53]:

```
EPOCHS = 50
try:
    print("Training on GPU:")
    with tensorflow.device("gpu:0"):  # train on gpu
        h4 = model4.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)
except Exception as e:
    print(e)
    print("Training on CPU:")
    h4 = model4.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)

print("\nTraining Done.")
```

```
Training on GPU:
 ~| Epoch: 1 | Validation Loss: 1.98917 >|>
 ~| Epoch: 2 | Validation Loss: 1.84032 >|>
 ~| Epoch: 3 | Validation Loss: 1.71317 >|>
 ~| Epoch: 4 | Validation Loss: 1.69194 >|>
 ~| Epoch: 5 | Validation Loss: 1.62231 >|>
 ~| Epoch: 6 | Validation Loss: 1.60189 >|>
 ~| Epoch: 7 | Validation Loss: 1.61941 >|>
 ~| Epoch: 8 | Validation Loss: 1.63725 >|>
 ~| Epoch: 9 | Validation Loss: 1.66948 >|>
 ~| Epoch: 10 | Validation Loss: 1.57714 >|>
 ~| Epoch: 11 | Validation Loss: 1.57898 >|>
 ~| Epoch: 12 | Validation Loss: 1.66979 >|>
 ~| Epoch: 13 | Validation Loss: 1.63858 >|>
 ~| Epoch: 14 | Validation Loss: 1.65452 >|>

Epoch 00014: ReduceLROnPlateau reducing learning rate to 0.0004000000189989805.
 ~| Epoch: 15 | Validation Loss: 1.66364 >|>
 ~| Epoch: 16 | Validation Loss: 1.66259 >|>
 ~| Epoch: 17 | Validation Loss: 1.68847 >|>

Training Done.
```

In [54]:

```
plot_learning_curve(h4)
```



In [55]:
```
loss, acc = model4.evaluate(X_test_padded, y_test)
print("Testing Accuracy: ", acc*100)
```

```
53/53 [==============================] - 1s 16ms/step - loss: 1.5843 - accuracy: 0.6586
Testing Accuracy:  65.85655212402344
```

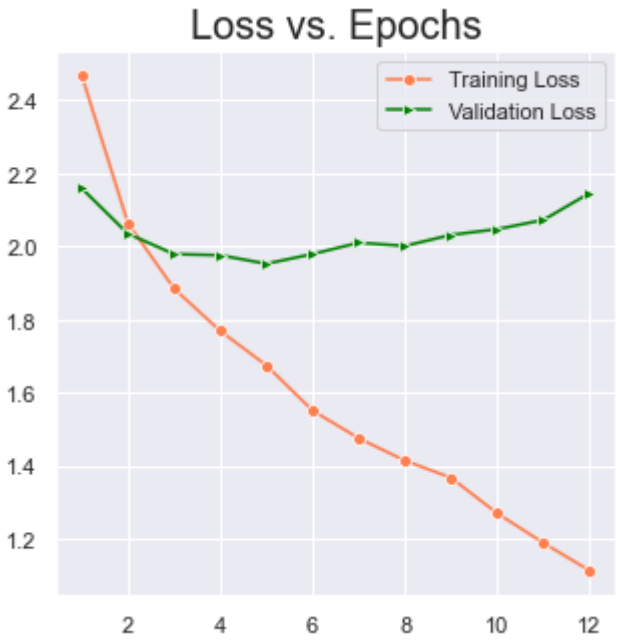
- **CNN (Dimensionality Reduction) + LSTM**

In [56]:
```
model5 = Sequential([
    Embedding(input_dim=VOCAB_SIZE, output_dim=256, input_length=MAX_LEN),
    Dropout(0.25),
    Conv1D(256, 5, padding = 'same', activation = 'relu', strides = 1),
    Conv1D(256, 5, padding = 'same', activation = 'relu', strides = 1),
    MaxPooling1D(pool_size = 2),
    Conv1D(64, 5, padding = 'same', activation = 'relu', strides = 1),
    MaxPooling1D(pool_size = 2),
    LSTM(75),
    Dense(NUM_CLASSES, activation = 'softmax')
])

model5.compile(
    loss = 'categorical_crossentropy',
    optimizer = 'adam',
    metrics = ['accuracy']
)
```

In [57]:
```
EPOCHS = 20
try:
    print("Training on GPU:")
    with tensorflow.device("gpu:0"):  # train on gpu
        h5 = model5.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)
except Exception as e:
    print(e)
    print("Training on CPU:")
    h5 = model5.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)

print("\nTraining Done.")
```

```
Training on GPU:
 ~| Epoch: 1 | Validation Loss: 2.15953 >|>
 ~| Epoch: 2 | Validation Loss: 2.03527 >|>
 ~| Epoch: 3 | Validation Loss: 1.97933 >|>
 ~| Epoch: 4 | Validation Loss: 1.97626 >|>
 ~| Epoch: 5 | Validation Loss: 1.95289 >|>
 ~| Epoch: 6 | Validation Loss: 1.97950 >|>
 ~| Epoch: 7 | Validation Loss: 2.01043 >|>
 ~| Epoch: 8 | Validation Loss: 2.00164 >|>
 ~| Epoch: 9 | Validation Loss: 2.03195 >|>

Epoch 00009: ReduceLROnPlateau reducing learning rate to 0.0004000000189989805.
 ~| Epoch: 10 | Validation Loss: 2.04752 >|>
 ~| Epoch: 11 | Validation Loss: 2.07280 >|>
 ~| Epoch: 12 | Validation Loss: 2.14650 >|>

Training Done.
```

In [58]:
```
plot_learning_curve(h5)
```



In [59]:
```
loss, acc = model5.evaluate(X_test_padded, y_test)
print("Testing Accuracy: ", acc*100)
```

```
53/53 [==============================] - 1s 14ms/step - loss: 1.9861 - accuracy: 0.5471
Testing Accuracy:  54.712510108947754
```

- **CNN (Dimensionality Reduction) + Bi-Directional LSTM**

In [60]:
```
model5 = Sequential([
    Embedding(input_dim=VOCAB_SIZE, output_dim=256, input_length=MAX_LEN),
    Dropout(0.25),
    Conv1D(256, 5, padding = 'same', activation = 'relu', strides = 1),
    Conv1D(256, 5, padding = 'same', activation = 'relu', strides = 1),
    MaxPooling1D(pool_size = 2),
    Conv1D(64, 5, padding = 'same', activation = 'relu', strides = 1),
    MaxPooling1D(pool_size = 2),
    Bidirectional(LSTM(75, recurrent_dropout=0.5)),
    Dense(NUM_CLASSES, activation = 'softmax')
])

model5.compile(
    loss = 'categorical_crossentropy',
    optimizer = 'adam',
    metrics = ['accuracy']
)
```

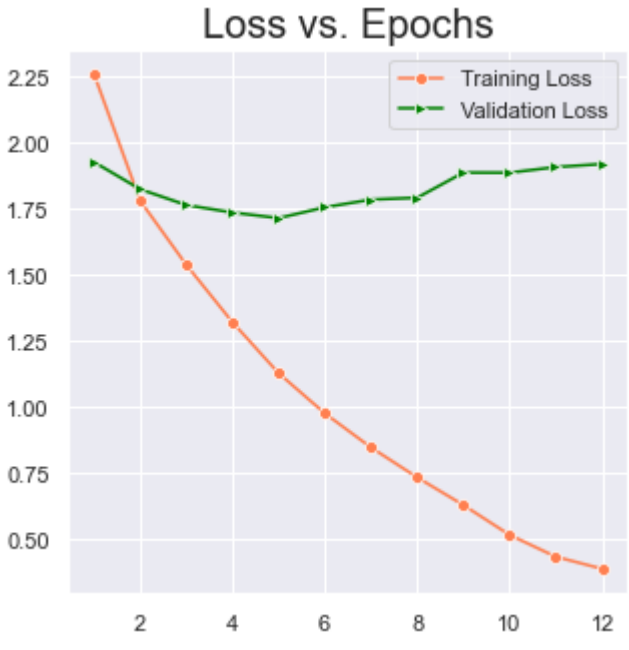
```
WARNING:tensorflow:Layer lstm_3 will not use cuDNN kernel since it doesn't meet the cuDNN kerne
l criteria. It will use generic GPU kernel as fallback when running on GPU
WARNING:tensorflow:Layer lstm_3 will not use cuDNN kernel since it doesn't meet the cuDNN kerne
```

```
l criteria. It will use generic GPU kernel as fallback when running on GPU
WARNING:tensorflow:Layer lstm_3 will not use cuDNN kernel since it doesn't meet the cuDNN kerne
l criteria. It will use generic GPU kernel as fallback when running on GPU
```

In [61]:
```
EPOCHS = 20
try:
    print("Training on GPU:")
    with tensorflow.device("gpu:0"):  # train on gpu
        h5 = model5.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)
except Exception as e:
    print(e)
    print("Training on CPU:")
    h5 = model5.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)

print("\nTraining Done.")
```

```
Training on GPU:
 ~| Epoch: 1 | Validation Loss: 1.92552 >|>
 ~| Epoch: 2 | Validation Loss: 1.82282 >|>
 ~| Epoch: 3 | Validation Loss: 1.76291 >|>
 ~| Epoch: 4 | Validation Loss: 1.73482 >|>
 ~| Epoch: 5 | Validation Loss: 1.71375 >|>
 ~| Epoch: 6 | Validation Loss: 1.75450 >|>
 ~| Epoch: 7 | Validation Loss: 1.78363 >|>
 ~| Epoch: 8 | Validation Loss: 1.79060 >|>
 ~| Epoch: 9 | Validation Loss: 1.88543 >|>

Epoch 00009: ReduceLROnPlateau reducing learning rate to 0.0004000000189989805.
 ~| Epoch: 10 | Validation Loss: 1.88504 >|>
 ~| Epoch: 11 | Validation Loss: 1.90626 >|>
 ~| Epoch: 12 | Validation Loss: 1.91783 >|>

Training Done.
```

In [62]:
```
plot_learning_curve(h5)
```



In [63]:
```
loss, acc = model5.evaluate(X_test_padded, y_test)
print("Testing Accuracy: ", acc*100)
```

```
53/53 [==============================] - 4s 80ms/step - loss: 1.7774 - accuracy: 0.5987
```

```
Testing Accuracy:  59.869593381881714
```

- **Use Tfidf vectors instead of Embedding Layer + Feature Selection**

In [72]:
```
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.model_selection import train_test_split

# using 75-25 split instead of 50-50 split as we need more data to train neural nets
X_train, X_test, y_train_vec, y_test_vec = train_test_split(X, y, test_size=0.2, random_state=s
print(f"Train dataset shape: {X_train.shape}, \nTest dataset shape: {X_test.shape}")
```

```
Train dataset shape: (6745,),
Test dataset shape: (1687,)
```

In [73]:
```
NGRAM_RANGE = (1, 2)
TOP_K = 20000
TOKEN_MODE = 'word'
MIN_DOC_FREQ = 2


kwargs = {
    'ngram_range' : NGRAM_RANGE,
    'dtype' : 'int32',
    'strip_accents' : 'unicode',
    'decode_error' : 'replace',
    'analyzer' : TOKEN_MODE,
    'min_df' : MIN_DOC_FREQ
}
vectorizer = TfidfVectorizer(**kwargs)
X_train_vec = vectorizer.fit_transform(X_train)
X_test_vec = vectorizer.transform(X_test)
print(f"Train dataset shape: {X_train_vec.shape}, \nTest dataset shape: {X_test_vec.shape}")
```

```
Train dataset shape: (6745, 17085),
Test dataset shape: (1687, 17085)
```

In [74]:
```
from sklearn.feature_selection import SelectKBest, f_classif

# Select best k features, with feature importance measured by f_classif
# Set k as 20000 or (if number of ngrams is less) number of ngrams
selector = SelectKBest(f_classif, k=min(TOP_K, X_train_vec.shape[1]))
selector.fit(X_train_vec, y_train_vec)
X_train_vec = selector.transform(X_train_vec).astype('float32')
X_test_vec = selector.transform(X_test_vec).astype('float32')
X_train_vec = X_train_vec.toarray()
X_test_vec = X_test_vec.toarray()

print(f"Train dataset shape: {X_train_vec.shape}, \nTest dataset shape: {X_test_vec.shape}")
```

```
Train dataset shape: (6745, 17085),
Test dataset shape: (1687, 17085)
```

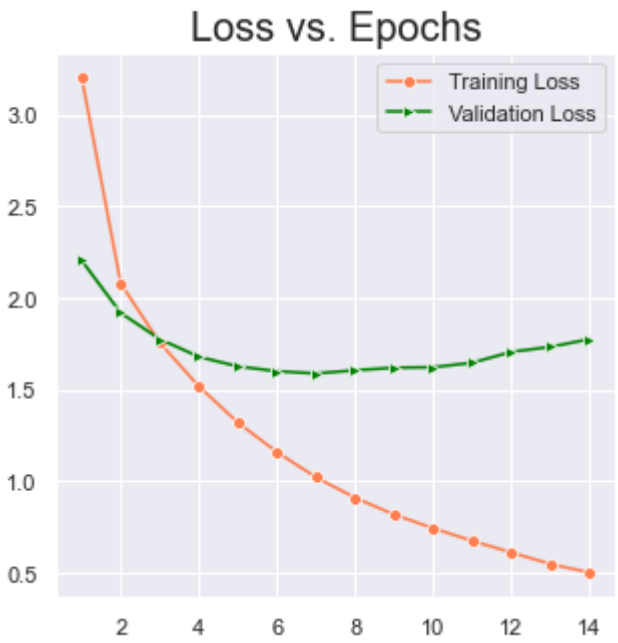
In [79]:
```
model6 = Sequential([
    Dense(64, activation='relu', input_shape=X_train_vec.shape[1:]),
    Dropout(0.2),
    Dense(16, activation='relu'),
    Dropout(0.2),
    Dense(NUM_CLASSES, activation='softmax')
])

model6.compile(
    loss = 'categorical_crossentropy',
    optimizer = 'adam',
    metrics = ['accuracy']
)
```

In [80]:
```
EPOCHS = 20
try:
    print("Training on GPU:")
```

```
    with tensorflow.device("gpu:0"):  # train on gpu
        h6 = model6.fit(
            X_train_vec, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop],
            verbose = False)
except Exception:
    print("Training on CPU:")
    h6 = model6.fit(
            X_train_vec, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop],
            verbose = False)

print("\nTraining Done.")
```

```
Training on GPU:
 ~| Epoch: 1 | Validation Loss: 2.20277 >|>
 ~| Epoch: 2 | Validation Loss: 1.91673 >|>
 ~| Epoch: 3 | Validation Loss: 1.77355 >|>
 ~| Epoch: 4 | Validation Loss: 1.67736 >|>
 ~| Epoch: 5 | Validation Loss: 1.62489 >|>
 ~| Epoch: 6 | Validation Loss: 1.59852 >|>
 ~| Epoch: 7 | Validation Loss: 1.58639 >|>
 ~| Epoch: 8 | Validation Loss: 1.60385 >|>
 ~| Epoch: 9 | Validation Loss: 1.61785 >|>
 ~| Epoch: 10 | Validation Loss: 1.62052 >|>
 ~| Epoch: 11 | Validation Loss: 1.64508 >|>
 ~| Epoch: 12 | Validation Loss: 1.70458 >|>
 ~| Epoch: 13 | Validation Loss: 1.73147 >|>
 ~| Epoch: 14 | Validation Loss: 1.77386 >|>

Training Done.
```

In [81]:
```
plot_learning_curve(h6)
```



In [82]:
```
loss, acc = model6.evaluate(X_test_vec, y_test)
print("Testing Accuracy: ", acc*100)
```

```
53/53 [==============================] - 0s 7ms/step - loss: 1.5654 - accuracy: 0.6680
Testing Accuracy:  66.80498123168945
```

- **Resultant Metrics:**

Model	Test Accuracy
Simple Feed-Forward Nerual Net	60.40

Feed-Forward NN + Batch Norm	63.43
Feed-Forward NN + Dropout	64.73
Feed-Forward NN + Pre-trained GLoVe embeddings	61.53
LSTM	49.91
Bi-Directional LSTM	65.87
Convolution Blocks (Dimensionality Reduction) + LSTM	54.71
Convolution Blocks (Dimensionality Reduction) + Bi-LSTM	59.87
Tfldf Vectors + Feature Selection + Feed-forward Neural Net	66.80

- **OBJECTIVE:**

Use the text data to build simple feed-forward Neural Nets and benchmark against the base ML models.

In [1]:
```
# imports

import os
import math
import random
import warnings
from time import time
from pathlib import Path
import pandas as pd, numpy as np
from pprint import pprint
import matplotlib.pyplot as plt
import seaborn as sns
from tqdm import tqdm
from collections import defaultdict, Counter
from sklearn.preprocessing import LabelEncoder
from wordcloud import WordCloud, STOPWORDS
import tensorflow

tqdm.pandas()
warnings.filterwarnings('ignore')
warnings.simplefilter(action='ignore', category=FutureWarning)
%matplotlib inline
```

```
C:\Users\surya\anaconda3\envs\full\lib\site-packages\tqdm\std.py:697: FutureWarning: The Panel
class is removed from pandas. Accessing it from the top-level namespace will also be removed in
the next version
  from pandas import Panel
```

In [2]:
```
# reproducibility
seed = 7
random.seed(seed)
tensorflow.random.set_seed(seed)
```

- **Import the clean data.**

In [3]:
```
dataset = pd.read_excel('./data/cleaned_data.xlsx')
dataset.sample(10)
```

Out[3]:

	short_description	description	caller	group	char_length	word_length	short
2600	job Job_534 failed in job_scheduler at: 10/01/...	received from: monitoring_tool@company.com\n\n...	bpctwhsn kzqsbmtp	GRP_8	105	11	

	short_description	description	caller	group	char_length	word_length	short
2113	4908206193/00001 dn required by 2 oclock est	\n\nreceived from: kaguhxwo.uoyipxqg@gmail.com...	kaguhxwo uoyipxqg	GRP_6	135	17	
5215	unable to connect to company wi-fi	unable to connect to company wi-fi	ntuhoafg bzwefjvk	GRP_0	35	6	
5068	netweaver -	\n\nreceived from: bcxpeuko.utorqehx@gmail.com...	bcxpeuko utorqehx	GRP_0	176	21	
4093	HostName_170-swap space on:HostName_170 is 75...	HostName_170- swap space on:HostName_170 is 75...	spxqmiry zpwgoqju	GRP_47	55	8	
6155	it help for engineering_tool and engineering_...	dear sir,\n\nplease help to download software ...	vxhyftae tbkyfdli	GRP_0	248	41	
5578	i need access to the deleted folder on collabo...	i can't find a mti certificate tracking form o...	hdfcwmag plxstkad	GRP_16	111	18	
4198	milano,italy: duplex mismatch gi2/0/1 on 1811...	duplex mismatch: duplex mode on interface giga...	mnlazfsr mtqrkhnx	GRP_4	531	66	
1556	grir issues plant_322 for ice alt. routes from...	email from maryhtutina bauuyternfeyt to athynd...	cbligfne wmoxktnj	GRP_53	443	63	
852	pc rqxw8515267 setup for remote company use ca...	pc rqxw8515267 setup for remote company use ca...	xweclugf qmhbjsyi	GRP_3	111	19	

In [4]:
```
dataset.isna().sum()
```

Out[4]:
```
short_description              0
description                    0
caller                         0
group                          0
char_length                    0
word_length                    0
short_char_length              0
short_word_length              0
description_keywords           7
short_description_keywords    38
group_code                     0
char_length_bins               0
cleaned_description            0
cleaned_short_description      0
cleaned_char_length            0
cleaned_word_length            0
cleaned_short_char_length      0
cleaned_short_word_length      0
dtype: int64
```

In [5]:
```
dataset.loc[dataset["group"] != 'GRP_0', 'group'] = 'Other'
dataset.loc[dataset["group"] == 'GRP_0', "group"] = 'Group 0'
```

In [6]:
```
dataset.group.value_counts()
```

Out[6]:
```
Other      4473
Group 0    3959
Name: group, dtype: int64
```

In [7]:
```
le = LabelEncoder()
dataset['group_code'] = le.fit_transform(dataset.group)
dataset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8432 entries, 0 to 8431
Data columns (total 18 columns):
 #   Column                       Non-Null Count  Dtype 
---  ------                       --------------  ----- 
 0   short_description            8432 non-null   object
 1   description                  8432 non-null   object
 2   caller                       8432 non-null   object
 3   group                        8432 non-null   object
 4   char_length                  8432 non-null   int64 
 5   word_length                  8432 non-null   int64 
 6   short_char_length            8432 non-null   int64 
 7   short_word_length            8432 non-null   int64 
 8   description_keywords         8425 non-null   object
 9   short_description_keywords   8394 non-null   object
 10  group_code                   8432 non-null   int32 
 11  char_length_bins             8432 non-null   int64 
 12  cleaned_description          8432 non-null   object
 13  cleaned_short_description    8432 non-null   object
 14  cleaned_char_length          8432 non-null   int64 
 15  cleaned_word_length          8432 non-null   int64 
 16  cleaned_short_char_length    8432 non-null   int64 
 17  cleaned_short_word_length    8432 non-null   int64 
dtypes: int32(1), int64(9), object(8)
memory usage: 1.1+ MB
```

In [8]:
```
le.classes_
```

Out[8]:
```
array(['Group 0', 'Other'], dtype=object)
```

In [9]:
```
def merge_descriptions(row):
    merged_descr = np.nan
    if (row.cleaned_short_description == row.cleaned_description or
        str(row.description).startswith(str(row.cleaned_short_description))):
        merged_descr = str(row.cleaned_description)
    else:
        merged_descr = str(row.cleaned_short_description) + " " + str(row.cleaned_description)
    row['merged_description'] = str(merged_descr)
    return row

dataset = dataset.progress_apply(merge_descriptions, axis=1)
```

```
100%|██████████| 8432/8432 [00:10<00:00, 823.10it/s]
```

In [10]:
```
dataset[['cleaned_short_description', 'cleaned_description', 'merged_description']].sample(10)
```

Out[10]:

	cleaned_short_description	cleaned_description	merged_description
6628	vpn connection issue	vpn connection issue connect user system use t...	vpn connection issue connect user system use t...
753	job bk biaprod fail job scheduler 05 12 00	job bk biaprod fail job scheduler 05 12 00	job bk biaprod fail job scheduler 05 12 00
5807	window printing issue need driver instal every...	window printing issue need driver instal every...	window printing issue need driver instal every...
7788	abende job job scheduler job 1148	abende job job scheduler job 1148 01 15 24	abende job job scheduler job 1148 abende job j...
2974	unable login ess protel	unable login ess portal	unable login ess protel unable login ess portal
6821	user unable login erp	user unable login erp	user unable login erp
2530	bex analyzer bex designer work	bex analyzer bex designer work	bex analyzer bex designer work

	cleaned_short_description	cleaned_description	merged_description
7194	setup new ws gonzale	setup new ws gonzale	setup new ws gonzale
6523	open pptx file attach email give repair error	open pptx file attach email give repair error	open pptx file attach email give repair error
7671	abende job job scheduler sid 38hotf	abende job job scheduler sid 38hotf 23 06 26	abende job job scheduler sid 38hotf abende job...

In [11]:
```
X = np.array(dataset.merged_description)
y = np.array(dataset.group_code)
X.shape, y.shape
```

Out[11]:
```
((8432,), (8432,))
```

In [12]:
```
from tensorflow.keras.utils import to_categorical
y_dummy_coded = to_categorical(y)
y[0], y_dummy_coded[0]
```

Out[12]:
```
(0, array([1., 0.], dtype=float32))
```

In [13]:
```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y_dummy_coded, test_size=.2, random_stat
```

In [14]:
```
X_train.shape, X_test.shape, y_train.shape, y_test.shape
```

Out[14]:
```
((6745,), (1687,), (6745, 2), (1687, 2))
```

In [15]:
```
X_train[0], y_train[0]  # check sample
```

Out[15]:
```
('additional correction sale org 1278 company address phone number require germany move 1 sale
organisation address 1278 phone number fax number need reverse original phone fax number furth
0911 2 plant address plant 124 phone fax number need adjusted show germany central phone number
fax 3 company code address 5278 need revert back address furth germany detail attach ticket',
 array([0., 1.], dtype=float32))
```

In [16]:
```
# TODO: Check the distributions of groups in training and testing sets, i.e, if they vary too m
# stratify by y if required during splits
# or data augmentation to upsample minority classes to balance the group distributions
```

- ## Tokenize and pad sequences

In [17]:
```
# define params
NUM_WORDS = 20000
EMBEDDING_DIM = 300
MAX_LEN = 100 # dataset['word_length'].max()
MAX_LEN
```

Out[17]:
```
100
```

In [18]:
```
from tensorflow.keras.preprocessing.text import Tokenizer
from tensorflow.keras.preprocessing.sequence import pad_sequences


tokenizer = Tokenizer(num_words=NUM_WORDS)
tokenizer.fit_on_texts(X_train)
X_train_tokens = tokenizer.texts_to_sequences(X_train)
X_test_tokens = tokenizer.texts_to_sequences(X_test)
X_train_tokens[0], X_test_tokens[0]
```

Out[18]:
```
([184,
  2093,
  93,
  606,
```

```
 1609,
 10,
 148,
 45,
 116,
 169,
 146,
 375,
 15,
 93,
 6817,
 148,
 1609,
 45,
 116,
 1513,
 116,
 23,
 1729,
 674,
 45,
 1513,
 116,
 804,
 6818,
 30,
 75,
 148,
 75,
 2370,
 45,
 1513,
 116,
 23,
 4267,
 90,
 146,
 3354,
 45,
 116,
 1513,
 73,
 10,
 188,
 148,
 6819,
 23,
 1195,
 174,
 148,
 804,
 146,
 117,
 88,
 17],
[93, 2095, 280, 1029, 783, 355, 9, 2095, 1029, 1360, 2095, 211, 280])
```

In [19]:
```
y_train[0], y_test[0]
```

Out[19]:
```
(array([0., 1.], dtype=float32), array([0., 1.], dtype=float32))
```

In [20]:
```
# pad sequences to cut longer texts to a uniform length and pad the sentences that are shorter

# using just 20 words from each headline will severely limit the information that is
# available to the model and affect performance although the training will be faster
X_train_padded = pad_sequences(X_train_tokens,
                               padding='post',
                               truncating='post',
                               maxlen=MAX_LEN)
X_test_padded = pad_sequences(X_test_tokens,
                              padding='post',
                              truncating='post',
                              maxlen=MAX_LEN)
```

```
print(f'X train:  {X_train_padded.shape}\nX test:   {X_test_padded.shape}')
```

```
X train:  (6745, 100)
X test:   (1687, 100)
```

In [21]:

```
pprint(X_train_padded[0], compact=True)
```

```
array([ 184, 2093,   93,  606, 1609,   10,  148,   45,  116,  169,  146,
        375,   15,   93, 6817,  148, 1609,   45,  116, 1513,  116,   23,
       1729,  674,   45, 1513,  116,  804, 6818,   30,   75,  148,   75,
       2370,   45, 1513,  116,   23, 4267,   90,  146, 3354,   45,  116,
       1513,   73,   10,  188,  148, 6819,   23, 1195,  174,  148,  804,
        146,  117,   88,   17,    0,    0,    0,    0,    0,    0,    0,
          0,    0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
          0,    0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
          0,    0,    0,    0,    0,    0,    0,    0,    0,    0,    0,
          0])
```

In [22]:

```
WORD_TO_INDEX = tokenizer.word_index
# pprint(WORD_TO_INDEX, compact=True)
pprint(list(WORD_TO_INDEX.keys())[:100], compact=True)
```

```
['job', 'yes', 'na', 'password', 'erp', 'tool', 'user', 'ts', 'issue',
 'company', 'sid', 'reset', 'access', 'scheduler', '1', '00', 'ticket',
 'unable', 'work', 'error', 'fail', 'account', 'need', 'email', 'site', 'help',
 'system', 'hostname', 'get', '2', 'login', 'circuit', 'power', 'outlook',
 'network', 'use', 'vendor', 'change', '34', 'update', 'name', 'message',
 'backup', 'see', 'phone', 'telecom', 'server', 'try', '10', 'able', 'outage',
 'log', 'check', 'new', 'problem', 'start', 'crm', 'engineering', 'request',
 'connect', 'call', 'usa', 'type', 'time', 'printer', 'order', 'report', 'vpn',
 'team', 'open', 'contact', 'skype', '3', 'lock', 'plant', 'et', 't', 'send',
 'create', '4', '5', 'window', 'file', 'pc', 'since', 'print', 'schedule',
 'attach', 'device', 'show', '8', 'maintenance', 'sale', '11', '12', 'receive',
 'abende', 'notify', '23', 'management']
```

In [23]:

```
VOCAB_SIZE = len(WORD_TO_INDEX) + 1
VOCAB_SIZE
```

Out[23]:

```
13790
```

In [24]:

```
# https://www.tensorflow.org/api_docs/python/tf/keras/preprocessing/sequence/pad_sequences
def retrieve_description_feat(x, mapping=WORD_TO_INDEX) -> str:
    # increment 3
    mapping = {k:(v + 3) for k, v in mapping.items()}
    mapping['<PAD>'] = 0
    mapping['<START>'] = 1
    mapping['<UNK>'] = 2
    inv_mapping = {v: k for k, v in mapping.items()}
    return str(" ".join(inv_mapping.get(i, '<NA>') for i in x))

retrieve_description_feat(X_test_padded[7])
```

Out[24]:

```
'1 22 team response complaint synchronize <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD>
<PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD>
<PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD>
<PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD>
<PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD>
<PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD>
<PAD> <PAD> <PAD> <PAD> <PAD>'
```

- ## GloVe Embeddings

In [25]:

```
EMBEDDING_DIM
```

Out[25]:

```
300
```

In [26]:

```
def get_embedding_matrix(embedding_dim=EMBEDDING_DIM):
```

```
    embeddings = defaultdict()
    if embedding_dim == 200:
        file_path = f'./data/glove.6B.{embedding_dim}d.txt'
    elif embedding_dim == 300:
        file_path = f'./data/glove.840B.{embedding_dim}d.txt'
    for l in open(file_path, encoding='utf-8'):
        word = l.split(" ")[0]
        embeddings[word] = np.asarray(l.split(" ")[1:], dtype='float32')

    embeddings = dict(embeddings)

    # create a weight matrix for words in training docs
    embedding_matrix = np.zeros((NUM_WORDS, embedding_dim))

    for word, idx in WORD_TO_INDEX.items():
        embedding_vector = embeddings.get(word)
        if embedding_vector is not None:
            embedding_matrix[idx] = embedding_vector

    return embedding_matrix
```

In [27]:
```
# use pre-trained glove embedding matrix to initialize weights in our model
embedding_matrix = get_embedding_matrix()
embedding_matrix.shape
```

Out[27]: (20000, 300)

- **Simple Feed-Forward Neural Net**

In [54]:
```
# !pip install livelossplot
from tensorflow.python.keras.models import Sequential
from sklearn.metrics import accuracy_score, confusion_matrix
from tensorflow.keras.regularizers import l2
from tensorflow.keras.constraints import max_norm, unit_norm
from tensorflow.python.keras.callbacks import LambdaCallback, EarlyStopping, ReduceLROnPlateau
from tensorflow.keras.layers import Flatten, Dense, Activation, BatchNormalization, Dropout, Em
```

In [29]:
```
NUM_CLASSES = len(le.classes_)
VOCAB_SIZE, MAX_LEN, EMBEDDING_DIM, NUM_CLASSES
```

Out[29]: (13790, 100, 300, 2)

In [30]:
```
# define model

model1 = Sequential([
    Embedding(input_dim=VOCAB_SIZE, output_dim=EMBEDDING_DIM, input_length=MAX_LEN),
    Flatten(),
    Dense(1024, activation = 'relu'),
    Dense(1024, activation = 'relu'),
    Dense(128, activation = 'relu'),
    Dense(NUM_CLASSES, activation = 'softmax')
])

model1.compile(
    loss = 'categorical_crossentropy',
    optimizer = 'adam',
    metrics = ['accuracy']
)
```

In [31]:
```
# Define Callbacks and a few helper functions

# simplify the training log
simple_log = LambdaCallback(
    on_epoch_end = lambda e, l: print(f" ~| Epoch: {e+1} | Validation Loss: {l['val_loss']:.5f}

# early stopping
```

```
early_stop = EarlyStopping(monitor='val_loss',
                           min_delta=0,
                           patience=7,
                           verbose=0,
                           restore_best_weights=True)

# learning rate reduction
lr_reduce_on_plateau = ReduceLROnPlateau(monitor='val_loss',
                                         patience=4,
                                         verbose=1,
                                         factor=0.4,
                                         min_lr=0.00001)

def plot_learning_curve(hist):
    sns.set()
    plt.figure(figsize=(5,5))
    train = hist.history['loss']
    val = hist.history['val_loss']
    epochs_run = range(1,len(train) + 1)
    sns.lineplot(epochs_run, train, marker = 'o', color = 'coral', label = 'Training Loss')
    sns.lineplot(epochs_run, val,  marker = '>', color = 'green', label = 'Validation Loss')
    plt.title("Loss vs. Epochs", fontsize = 20)
    plt.legend()
    plt.show()
```

In [32]:
```
X_train[0]
```

Out[32]:
```
'additional correction sale org 1278 company address phone number require germany move 1 sale o
rganisation address 1278 phone number fax number need reverse original phone fax number furth 0
911 2 plant address plant 124 phone fax number need adjusted show germany central phone number
fax 3 company code address 5278 need revert back address furth germany detail attach ticket'
```

In [33]:
```
X_train.shape, y_train.shape, X_test.shape, y_test.shape
```

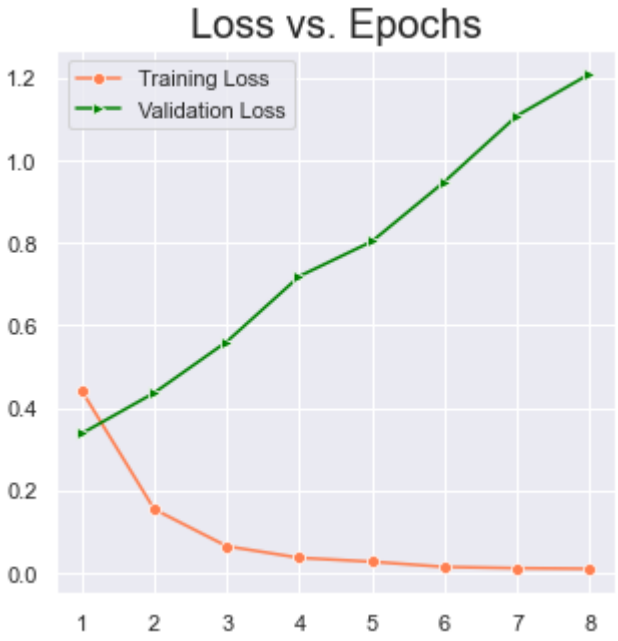
Out[33]:
```
((6745,), (6745, 2), (1687,), (1687, 2))
```

In [34]:
```
EPOCHS = 200
try:
    print("Training on GPU:")
    with tensorflow.device("gpu:0"):  # train on gpu
        h1 = model1.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)
except Exception as e:
    print(e)
    print("\nTraining on CPU:")
    h1 = model1.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)

print("\nTraining Done.")
```

```
Training on GPU:
 ~| Epoch: 1 | Validation Loss: 0.33744 >|>
 ~| Epoch: 2 | Validation Loss: 0.43625 >|>
 ~| Epoch: 3 | Validation Loss: 0.56031 >|>
 ~| Epoch: 4 | Validation Loss: 0.71875 >|>
 ~| Epoch: 5 | Validation Loss: 0.80328 >|>

Epoch 00005: ReduceLROnPlateau reducing learning rate to 0.0004000000189989805.
 ~| Epoch: 6 | Validation Loss: 0.94813 >|>
 ~| Epoch: 7 | Validation Loss: 1.10798 >|>
 ~| Epoch: 8 | Validation Loss: 1.20928 >|>
```

```
Training Done.
```

In [35]:
```
plot_learning_curve(h1)
```



In [36]:
```
loss, acc = model1.evaluate(X_test_padded, y_test)
print("Testing Loss: ", loss*100)
print("Testing Accuracy: ", acc*100)
```

```
53/53 [==============================] - 0s 6ms/step - loss: 0.3273 - accuracy: 0.8625
Testing Loss:  32.73244500160217
Testing Accuracy:  86.24777793884277
```

- This model is clearly overfitting, we will add regularization to the next iteration

- **Simple Feed-Forward Neural Net + Batch Normalization**

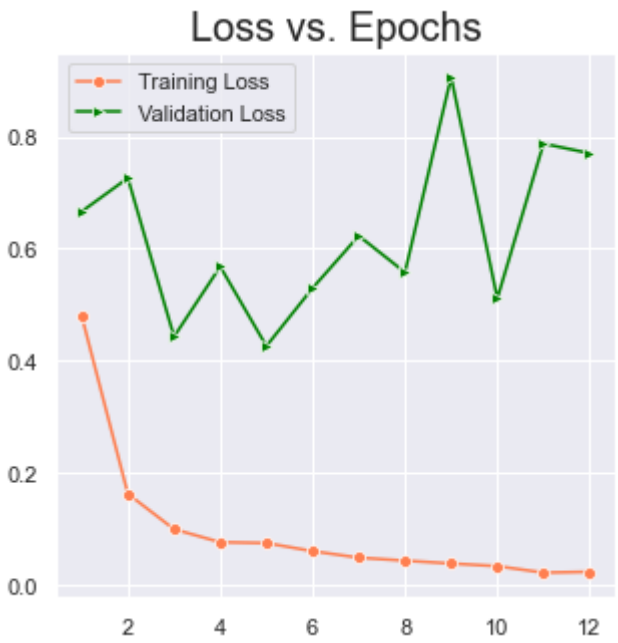
In [37]:
```
# define model

model2 = Sequential([
    Embedding(input_dim=VOCAB_SIZE, output_dim=EMBEDDING_DIM, input_length=MAX_LEN),
    Flatten(),
    Dense(256, activation = 'relu'),
    BatchNormalization(),
    Dense(256, activation = 'relu'),
    BatchNormalization(),
    Dense(NUM_CLASSES, activation = 'softmax')
])

model2.compile(
    loss = 'categorical_crossentropy',
    optimizer = 'adam',
    metrics = ['accuracy']
)
```

In [38]:
```
EPOCHS = 200
try:
    print("Training on GPU:")
    with tensorflow.device("gpu:0"):  # train on gpu
        h2 = model2.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)
except Exception as e:
```

```
    print(e)
    print("Training on CPU:")
    h2 = model2.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)

print("\nTraining Done.")
```

```
Training on GPU:
 ~| Epoch: 1 | Validation Loss: 0.66638 >|>
 ~| Epoch: 2 | Validation Loss: 0.72617 >|>
 ~| Epoch: 3 | Validation Loss: 0.44306 >|>
 ~| Epoch: 4 | Validation Loss: 0.56813 >|>
 ~| Epoch: 5 | Validation Loss: 0.42626 >|>
 ~| Epoch: 6 | Validation Loss: 0.52981 >|>
 ~| Epoch: 7 | Validation Loss: 0.62305 >|>
 ~| Epoch: 8 | Validation Loss: 0.55789 >|>
 ~| Epoch: 9 | Validation Loss: 0.90662 >|>

Epoch 00009: ReduceLROnPlateau reducing learning rate to 0.0004000000189989805.
 ~| Epoch: 10 | Validation Loss: 0.51133 >|>
 ~| Epoch: 11 | Validation Loss: 0.78763 >|>
 ~| Epoch: 12 | Validation Loss: 0.77035 >|>

Training Done.
```

In [39]:
```
plot_learning_curve(h2)
```



In [40]:
```
loss, acc = model2.evaluate(X_test_padded, y_test)
print("Testing Loss: ", loss*100)
print("Testing Accuracy: ", acc*100)
```

```
53/53 [==============================] - 0s 9ms/step - loss: 0.4056 - accuracy: 0.8376
Testing Loss:  40.56009352207184
Testing Accuracy:  83.75815153121948
```

- **Simple Feed-Forward Neural Net + Dropout**

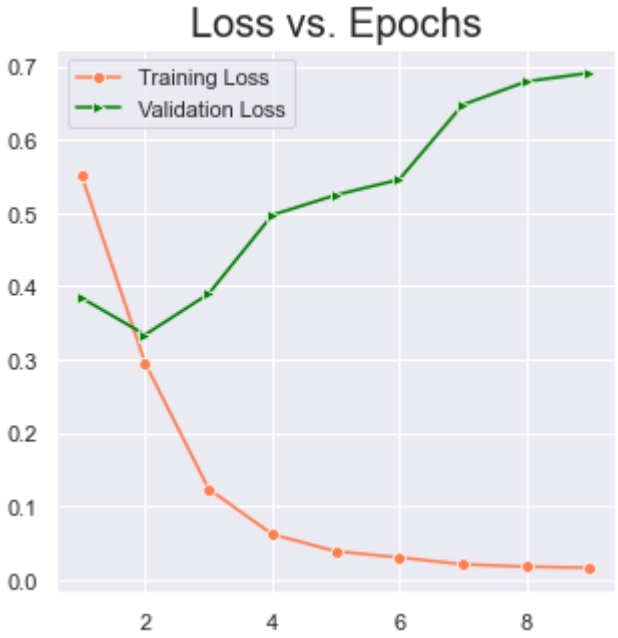
In [41]:
```
# define model

model3 = Sequential([
    Embedding(input_dim=VOCAB_SIZE, output_dim=EMBEDDING_DIM, input_length=MAX_LEN),
    Flatten(),
    Dense(20, activation = 'relu'),
    Dropout(0.4),
    Dense(NUM_CLASSES, activation = 'softmax')
```

```
])

model3.compile(
    loss = 'categorical_crossentropy',
    optimizer = 'adam',
    metrics = ['accuracy']
)
```

In [42]:
```
EPOCHS = 200
try:
    print("Training on GPU:")
    with tensorflow.device("gpu:0"):  # train on gpu
        h3 = model3.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)
except Exception as e:
    print(e)
    print("Training on CPU:")
    h3 = model3.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)

print("\nTraining Done.")
```

```
Training on GPU:
 ~| Epoch: 1 | Validation Loss: 0.38415 >|>
 ~| Epoch: 2 | Validation Loss: 0.33411 >|>
 ~| Epoch: 3 | Validation Loss: 0.38989 >|>
 ~| Epoch: 4 | Validation Loss: 0.49732 >|>
 ~| Epoch: 5 | Validation Loss: 0.52464 >|>
 ~| Epoch: 6 | Validation Loss: 0.54575 >|>

Epoch 00006: ReduceLROnPlateau reducing learning rate to 0.0004000000189989805.
 ~| Epoch: 7 | Validation Loss: 0.64775 >|>
 ~| Epoch: 8 | Validation Loss: 0.67945 >|>
 ~| Epoch: 9 | Validation Loss: 0.69136 >|>

Training Done.
```

In [43]:
```
plot_learning_curve(h3)
```



In [44]:
```
loss, acc = model3.evaluate(X_test_padded, y_test)
print("Testing Accuracy: ", acc*100)
```

```
53/53 [==============================] - 0s 4ms/step - loss: 0.3274 - accuracy: 0.8583
Testing Accuracy:  85.83284020423889
```

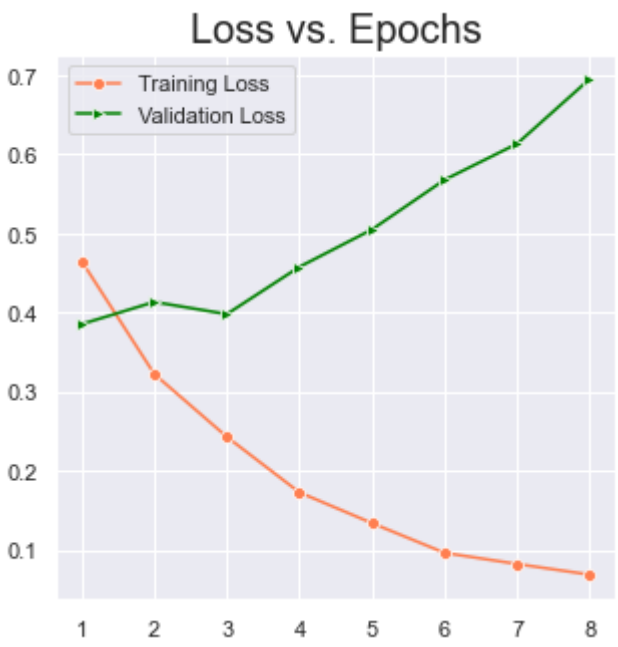
- **Use pre-trained embeddings**

In [45]:
```python
# define model

model3 = Sequential([
    Embedding(input_dim=NUM_WORDS, output_dim=EMBEDDING_DIM, input_length=MAX_LEN, weights=[emb
    Flatten(),
    Dense(30, activation = 'relu'),
    Dropout(0.5),
    Dense(NUM_CLASSES, activation = 'softmax')
])

model3.compile(
    loss = 'categorical_crossentropy',
    optimizer = 'rmsprop',
    metrics = ['accuracy']
)
```

In [46]:
```python
EPOCHS = 200
try:
    print("Training on GPU:")
    with tensorflow.device("gpu:0"):  # train on gpu
        h3 = model3.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)
except Exception as e:
    print(e)
    print("Training on CPU:")
    h3 = model3.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)

print("\nTraining Done.")
```

```
Training on GPU:
 ~| Epoch: 1 | Validation Loss: 0.38540 >|>
 ~| Epoch: 2 | Validation Loss: 0.41342 >|>
 ~| Epoch: 3 | Validation Loss: 0.39803 >|>
 ~| Epoch: 4 | Validation Loss: 0.45770 >|>
 ~| Epoch: 5 | Validation Loss: 0.50492 >|>

Epoch 00005: ReduceLROnPlateau reducing learning rate to 0.0004000000189989805.
 ~| Epoch: 6 | Validation Loss: 0.56770 >|>
 ~| Epoch: 7 | Validation Loss: 0.61330 >|>
 ~| Epoch: 8 | Validation Loss: 0.69508 >|>

Training Done.
```

In [47]:
```python
plot_learning_curve(h3)
```



In [48]:
```
loss, acc = model3.evaluate(X_test_padded, y_test)
print("Testing Accuracy: ", acc*100)
```

```
53/53 [==============================] - 0s 6ms/step - loss: 0.3927 - accuracy: 0.8275
Testing Accuracy:  82.75044560432434
```

- **LSTM**

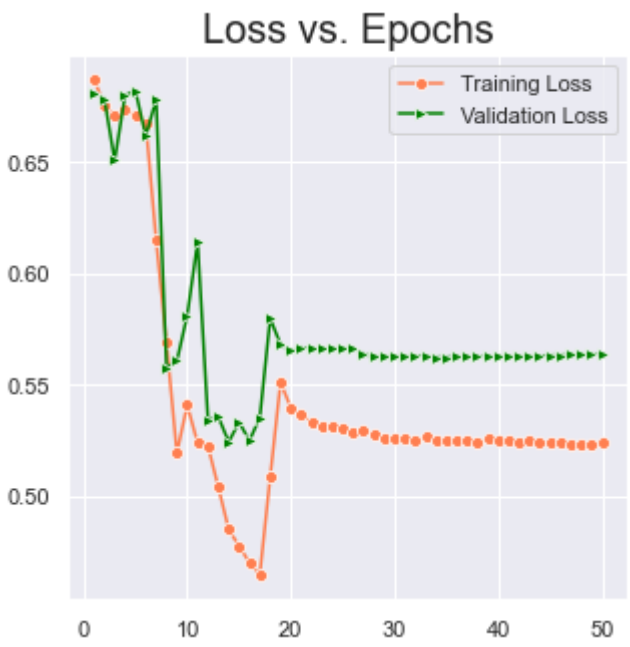
In [49]:
```
# define model

model4 = Sequential([
    Embedding(input_dim=VOCAB_SIZE, output_dim=EMBEDDING_DIM, input_length=MAX_LEN),
    LSTM(32),
    Dropout(0.4),
    Dense(NUM_CLASSES, activation = 'softmax')
])

model4.compile(
    loss = 'categorical_crossentropy',
    optimizer = 'adam',
    metrics = ['accuracy']
)
```

In [50]:
```
EPOCHS = 50
try:
    print("Training on GPU:")
    with tensorflow.device("gpu:0"):  # train on gpu
        h4 = model4.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, lr_reduce_on_plateau],
            verbose = False)
except Exception as e:
    print(e)
    print("Training on CPU:")
    h4 = model4.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, lr_reduce_on_plateau],
            verbose = False)

print("\nTraining Done.")
```

```
Training on GPU:
 ~| Epoch: 1 | Validation Loss: 0.68090 >|>
```

```
 ~| Epoch: 2 | Validation Loss: 0.67834 >|>
 ~| Epoch: 3 | Validation Loss: 0.65052 >|>
 ~| Epoch: 4 | Validation Loss: 0.67962 >|>
 ~| Epoch: 5 | Validation Loss: 0.68195 >|>
 ~| Epoch: 6 | Validation Loss: 0.66145 >|>
 ~| Epoch: 7 | Validation Loss: 0.67785 >|>

Epoch 00007: ReduceLROnPlateau reducing learning rate to 0.0004000000189989805.
 ~| Epoch: 8 | Validation Loss: 0.55761 >|>
 ~| Epoch: 9 | Validation Loss: 0.56050 >|>
 ~| Epoch: 10 | Validation Loss: 0.58099 >|>
 ~| Epoch: 11 | Validation Loss: 0.61384 >|>
 ~| Epoch: 12 | Validation Loss: 0.53396 >|>
 ~| Epoch: 13 | Validation Loss: 0.53542 >|>
 ~| Epoch: 14 | Validation Loss: 0.52415 >|>
 ~| Epoch: 15 | Validation Loss: 0.53307 >|>
 ~| Epoch: 16 | Validation Loss: 0.52474 >|>
 ~| Epoch: 17 | Validation Loss: 0.53513 >|>
 ~| Epoch: 18 | Validation Loss: 0.57956 >|>

Epoch 00018: ReduceLROnPlateau reducing learning rate to 0.00016000000759959222.
 ~| Epoch: 19 | Validation Loss: 0.56794 >|>
 ~| Epoch: 20 | Validation Loss: 0.56498 >|>
 ~| Epoch: 21 | Validation Loss: 0.56665 >|>
 ~| Epoch: 22 | Validation Loss: 0.56587 >|>

Epoch 00022: ReduceLROnPlateau reducing learning rate to 6.400000420399014e-05.
 ~| Epoch: 23 | Validation Loss: 0.56599 >|>
 ~| Epoch: 24 | Validation Loss: 0.56605 >|>
 ~| Epoch: 25 | Validation Loss: 0.56618 >|>
 ~| Epoch: 26 | Validation Loss: 0.56635 >|>

Epoch 00026: ReduceLROnPlateau reducing learning rate to 2.560000284574926e-05.
 ~| Epoch: 27 | Validation Loss: 0.56373 >|>
 ~| Epoch: 28 | Validation Loss: 0.56247 >|>
 ~| Epoch: 29 | Validation Loss: 0.56257 >|>
 ~| Epoch: 30 | Validation Loss: 0.56266 >|>

Epoch 00030: ReduceLROnPlateau reducing learning rate to 1.0240000847261399e-05.
 ~| Epoch: 31 | Validation Loss: 0.56270 >|>
 ~| Epoch: 32 | Validation Loss: 0.56276 >|>
 ~| Epoch: 33 | Validation Loss: 0.56280 >|>
 ~| Epoch: 34 | Validation Loss: 0.56219 >|>

Epoch 00034: ReduceLROnPlateau reducing learning rate to 1e-05.
 ~| Epoch: 35 | Validation Loss: 0.56223 >|>
 ~| Epoch: 36 | Validation Loss: 0.56229 >|>
 ~| Epoch: 37 | Validation Loss: 0.56236 >|>
 ~| Epoch: 38 | Validation Loss: 0.56242 >|>
 ~| Epoch: 39 | Validation Loss: 0.56248 >|>
 ~| Epoch: 40 | Validation Loss: 0.56256 >|>
 ~| Epoch: 41 | Validation Loss: 0.56263 >|>
 ~| Epoch: 42 | Validation Loss: 0.56272 >|>
 ~| Epoch: 43 | Validation Loss: 0.56281 >|>
 ~| Epoch: 44 | Validation Loss: 0.56288 >|>
 ~| Epoch: 45 | Validation Loss: 0.56298 >|>
 ~| Epoch: 46 | Validation Loss: 0.56307 >|>
 ~| Epoch: 47 | Validation Loss: 0.56317 >|>
 ~| Epoch: 48 | Validation Loss: 0.56327 >|>
 ~| Epoch: 49 | Validation Loss: 0.56340 >|>
 ~| Epoch: 50 | Validation Loss: 0.56350 >|>

Training Done.
```

In [51]:
```
plot_learning_curve(h4)
```



In [52]:
```
loss, acc = model4.evaluate(X_test_padded, y_test)
print("Testing Accuracy: ", acc*100)
```

```
53/53 [==============================] - 1s 10ms/step - loss: 0.5702 - accuracy: 0.6544
Testing Accuracy:  65.44161438941956
```

- **Bi-Directional LSTM**

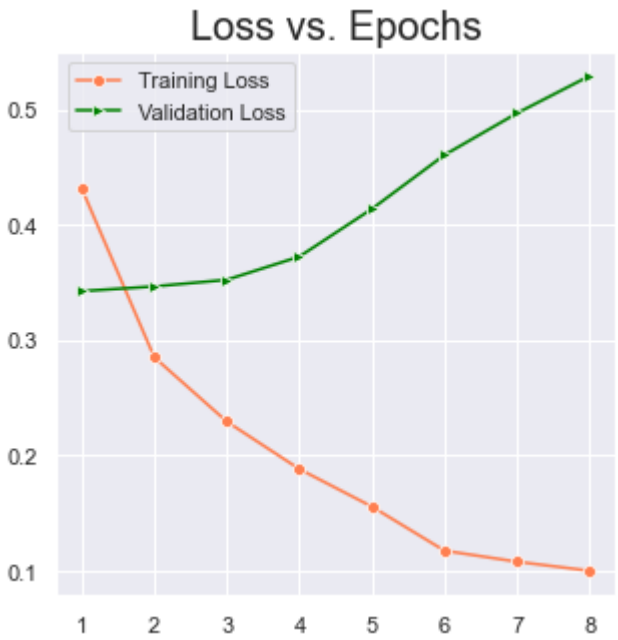
In [55]:
```
# define model

model4 = Sequential([
    Embedding(input_dim=VOCAB_SIZE, output_dim=EMBEDDING_DIM, input_length=MAX_LEN),
    Bidirectional(LSTM(32)),
    Dropout(0.4),
    Dense(NUM_CLASSES, activation = 'softmax')
])

model4.compile(
    loss = 'categorical_crossentropy',
    optimizer = 'rmsprop',
    metrics = ['accuracy']
)
```

In [56]:
```
EPOCHS = 50
try:
    print("Training on GPU:")
    with tensorflow.device("gpu:0"):  # train on gpu
        h4 = model4.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)
except Exception as e:
    print(e)
    print("Training on CPU:")
    h4 = model4.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)

print("\nTraining Done.")
```

```
Training on GPU:
 ~| Epoch: 1 | Validation Loss: 0.34246 >|>
```

```
 ~| Epoch: 2 | Validation Loss: 0.34657 >|>
 ~| Epoch: 3 | Validation Loss: 0.35232 >|>
 ~| Epoch: 4 | Validation Loss: 0.37244 >|>
 ~| Epoch: 5 | Validation Loss: 0.41386 >|>

Epoch 00005: ReduceLROnPlateau reducing learning rate to 0.0004000000189989805.
 ~| Epoch: 6 | Validation Loss: 0.46064 >|>
 ~| Epoch: 7 | Validation Loss: 0.49689 >|>
 ~| Epoch: 8 | Validation Loss: 0.52922 >|>

Training Done.
```

In [57]:
```
plot_learning_curve(h4)
```



In [58]:
```
loss, acc = model4.evaluate(X_test_padded, y_test)
print("Testing Accuracy: ", acc*100)
```

```
53/53 [==============================] - 1s 12ms/step - loss: 0.3354 - accuracy: 0.8524
Testing Accuracy:  85.24007201194763
```

- **CNN (Dimensionality Reduction) + LSTM**

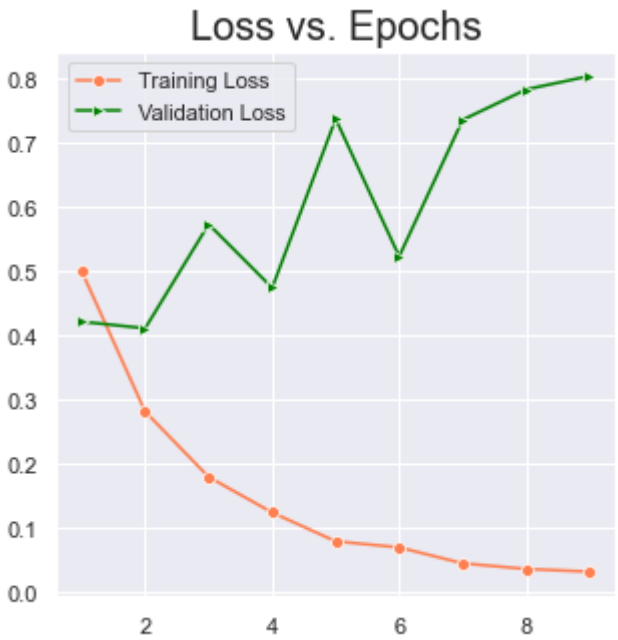
In [59]:
```
model5 = Sequential([
    Embedding(input_dim=VOCAB_SIZE, output_dim=256, input_length=MAX_LEN),
    Dropout(0.25),
    Conv1D(256, 5, padding = 'same', activation = 'relu', strides = 1),
    Conv1D(256, 5, padding = 'same', activation = 'relu', strides = 1),
    MaxPooling1D(pool_size = 2),
    Conv1D(64, 5, padding = 'same', activation = 'relu', strides = 1),
    MaxPooling1D(pool_size = 2),
    LSTM(75),
    Dense(NUM_CLASSES, activation = 'softmax')
])

model5.compile(
    loss = 'categorical_crossentropy',
    optimizer = 'adam',
    metrics = ['accuracy']
)
```

In [60]:
```
EPOCHS = 20
try:
    print("Training on GPU:")
    with tensorflow.device("gpu:0"):  # train on gpu
        h5 = model5.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
```

```
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)
except Exception as e:
    print(e)
    print("Training on CPU:")
    h5 = model5.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)

print("\nTraining Done.")
```

```
Training on GPU:
 ~| Epoch: 1 | Validation Loss: 0.42078 >|>
 ~| Epoch: 2 | Validation Loss: 0.41046 >|>
 ~| Epoch: 3 | Validation Loss: 0.57248 >|>
 ~| Epoch: 4 | Validation Loss: 0.47355 >|>
 ~| Epoch: 5 | Validation Loss: 0.73653 >|>
 ~| Epoch: 6 | Validation Loss: 0.52240 >|>

Epoch 00006: ReduceLROnPlateau reducing learning rate to 0.0004000000189989805.
 ~| Epoch: 7 | Validation Loss: 0.73506 >|>
 ~| Epoch: 8 | Validation Loss: 0.78235 >|>
 ~| Epoch: 9 | Validation Loss: 0.80333 >|>

Training Done.
```

In [61]:
```
plot_learning_curve(h5)
```



In [62]:
```
loss, acc = model5.evaluate(X_test_padded, y_test)
print("Testing Accuracy: ", acc*100)
```

```
53/53 [==============================] - 1s 12ms/step - loss: 0.3900 - accuracy: 0.8447
Testing Accuracy:  84.469473361969
```

- **CNN (Dimensionality Reduction) + Bi-Directional LSTM**

In [63]:
```
model5 = Sequential([
    Embedding(input_dim=VOCAB_SIZE, output_dim=256, input_length=MAX_LEN),
    Dropout(0.25),
    Conv1D(256, 5, padding = 'same', activation = 'relu', strides = 1),
    Conv1D(256, 5, padding = 'same', activation = 'relu', strides = 1),
    MaxPooling1D(pool_size = 2),
    Conv1D(64, 5, padding = 'same', activation = 'relu', strides = 1),
    MaxPooling1D(pool_size = 2),
    Bidirectional(LSTM(75, recurrent_dropout=0.5)),
    Dense(NUM_CLASSES, activation = 'softmax')
```

```
])

model5.compile(
    loss = 'categorical_crossentropy',
    optimizer = 'adam',
    metrics = ['accuracy']
)
```

```
WARNING:tensorflow:Layer lstm_3 will not use cuDNN kernel since it doesn't meet the cuDNN kerne
l criteria. It will use generic GPU kernel as fallback when running on GPU
WARNING:tensorflow:Layer lstm_3 will not use cuDNN kernel since it doesn't meet the cuDNN kerne
l criteria. It will use generic GPU kernel as fallback when running on GPU
WARNING:tensorflow:Layer lstm_3 will not use cuDNN kernel since it doesn't meet the cuDNN kerne
l criteria. It will use generic GPU kernel as fallback when running on GPU
```

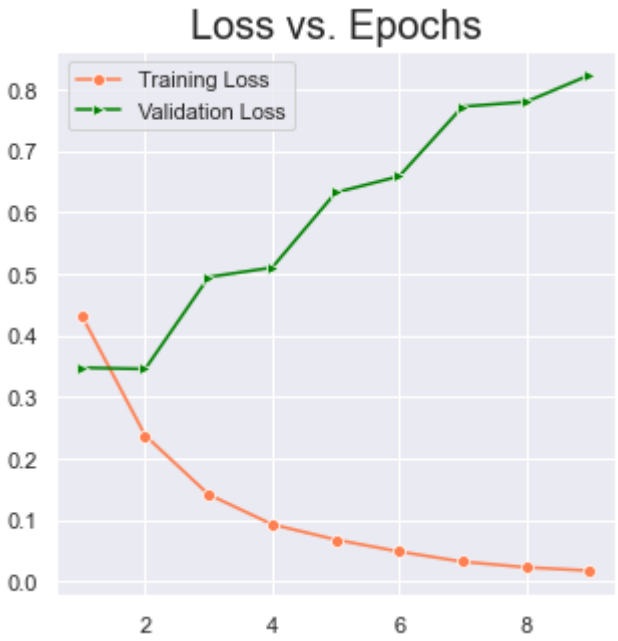
In [64]:

```
EPOCHS = 20
try:
    print("Training on GPU:")
    with tensorflow.device("gpu:0"):  # train on gpu
        h5 = model5.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)
except Exception as e:
    print(e)
    print("Training on CPU:")
    h5 = model5.fit(
            X_train_padded, y_train,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop, lr_reduce_on_plateau],
            verbose = False)

print("\nTraining Done.")
```

```
Training on GPU:
 ~| Epoch: 1 | Validation Loss: 0.34703 >|>
 ~| Epoch: 2 | Validation Loss: 0.34499 >|>
 ~| Epoch: 3 | Validation Loss: 0.49440 >|>
 ~| Epoch: 4 | Validation Loss: 0.50982 >|>
 ~| Epoch: 5 | Validation Loss: 0.63177 >|>
 ~| Epoch: 6 | Validation Loss: 0.65818 >|>

Epoch 00006: ReduceLROnPlateau reducing learning rate to 0.0004000000189989805.
 ~| Epoch: 7 | Validation Loss: 0.77149 >|>
 ~| Epoch: 8 | Validation Loss: 0.77925 >|>
 ~| Epoch: 9 | Validation Loss: 0.82289 >|>

Training Done.
```

In [65]:

```
plot_learning_curve(h5)
```



In [66]:
```
loss, acc = model5.evaluate(X_test_padded, y_test)
print("Testing Accuracy: ", acc*100)
```

```
53/53 [==============================] - 4s 77ms/step - loss: 0.3487 - accuracy: 0.8500
Testing Accuracy:  85.00296473503113
```

- **Use Tfidf vectors instead of Embedding Layer + Feature Selection**

In [67]:
```
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.model_selection import train_test_split

# using 75-25 split instead of 50-50 split as we need more data to train neural nets
X_train_vec, X_test_vec, y_train_vec, y_test_vec = train_test_split(X, y, test_size=0.2, random
print(f"Train dataset shape: {X_train_vec.shape}, \nTest dataset shape: {X_test_vec.shape}")
```

```
Train dataset shape: (6745,),
Test dataset shape: (1687,)
```

In [68]:
```
NGRAM_RANGE = (1, 2)
TOP_K = 10000
TOKEN_MODE = 'word'
MIN_DOC_FREQ = 2


kwargs = {
    'ngram_range' : NGRAM_RANGE,
    'dtype' : 'int32',
    'strip_accents' : 'unicode',
    'decode_error' : 'replace',
    'analyzer' : TOKEN_MODE,
    'min_df' : MIN_DOC_FREQ
}
vectorizer = TfidfVectorizer(**kwargs)
X_train_vec = vectorizer.fit_transform(X_train_vec)
X_test_vec = vectorizer.transform(X_test_vec)
print(f"Train dataset shape: {X_train_vec.shape}, \nTest dataset shape: {X_test_vec.shape}")
```

```
Train dataset shape: (6745, 17085),
Test dataset shape: (1687, 17085)
```

In [69]:
```
from sklearn.feature_selection import SelectKBest, f_classif

# Select best k features, with feature importance measured by f_classif
# Set k as 20000 or (if number of ngrams is less) number of ngrams
selector = SelectKBest(f_classif, k=min(TOP_K, X_train_vec.shape[1]))
selector.fit(X_train_vec, y_train_vec)
X_train_vec = selector.transform(X_train_vec).astype('float32')
X_test_vec = selector.transform(X_test_vec).astype('float32')
```

```
X_train_vec = X_train_vec.toarray()
X_test_vec = X_test_vec.toarray()

print(f"Train dataset shape: {X_train_vec.shape}, \nTest dataset shape: {X_test_vec.shape}")
```

```
Train dataset shape: (6745, 10000),
Test dataset shape: (1687, 10000)
```

In [70]:

```
model6 = Sequential([
    Dense(20, activation='relu', input_shape=X_train_vec.shape[1:]),
    Dropout(0.5),
    Dense(20, activation='relu'),
    Dropout(0.5),
    Dense(1, activation='sigmoid')
])

model6.compile(
    loss = 'binary_crossentropy',
    optimizer = 'rmsprop',
    metrics = ['accuracy']
)
```

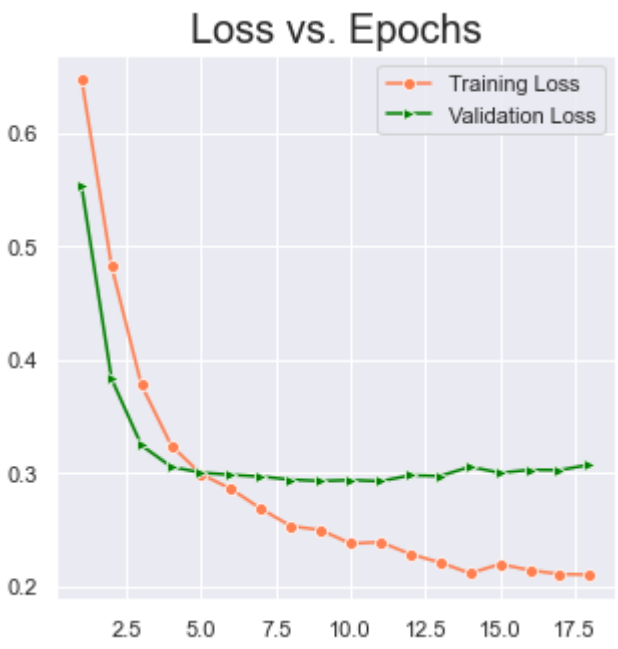
In [71]:

```
EPOCHS = 20
try:
    print("Training on GPU:")
    with tensorflow.device("gpu:0"):  # train on gpu
        h6 = model6.fit(
            X_train_vec, y_train_vec,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop],
            verbose = False)
except Exception:
    print("Training on CPU:")
    h6 = model6.fit(
            X_train_vec, y_train_vec,
            validation_split = 0.2,  # do not use the test data for validation to prevent data
            epochs = EPOCHS,
            callbacks = [simple_log, early_stop],
            verbose = False)

print("\nTraining Done.")
```

```
Training on GPU:
 ~| Epoch: 1 | Validation Loss: 0.55280 >|>
 ~| Epoch: 2 | Validation Loss: 0.38299 >|>
 ~| Epoch: 3 | Validation Loss: 0.32426 >|>
 ~| Epoch: 4 | Validation Loss: 0.30529 >|>
 ~| Epoch: 5 | Validation Loss: 0.30012 >|>
 ~| Epoch: 6 | Validation Loss: 0.29826 >|>
 ~| Epoch: 7 | Validation Loss: 0.29672 >|>
 ~| Epoch: 8 | Validation Loss: 0.29396 >|>
 ~| Epoch: 9 | Validation Loss: 0.29277 >|>
 ~| Epoch: 10 | Validation Loss: 0.29358 >|>
 ~| Epoch: 11 | Validation Loss: 0.29257 >|>
 ~| Epoch: 12 | Validation Loss: 0.29779 >|>
 ~| Epoch: 13 | Validation Loss: 0.29718 >|>
 ~| Epoch: 14 | Validation Loss: 0.30539 >|>
 ~| Epoch: 15 | Validation Loss: 0.29990 >|>
 ~| Epoch: 16 | Validation Loss: 0.30262 >|>
 ~| Epoch: 17 | Validation Loss: 0.30239 >|>
 ~| Epoch: 18 | Validation Loss: 0.30721 >|>

Training Done.
```

In [72]:

```
plot_learning_curve(h6)
```



In [73]:
```
loss, acc = model6.evaluate(X_test_vec, y_test_vec)
print("Testing Accuracy: ", acc*100)
```

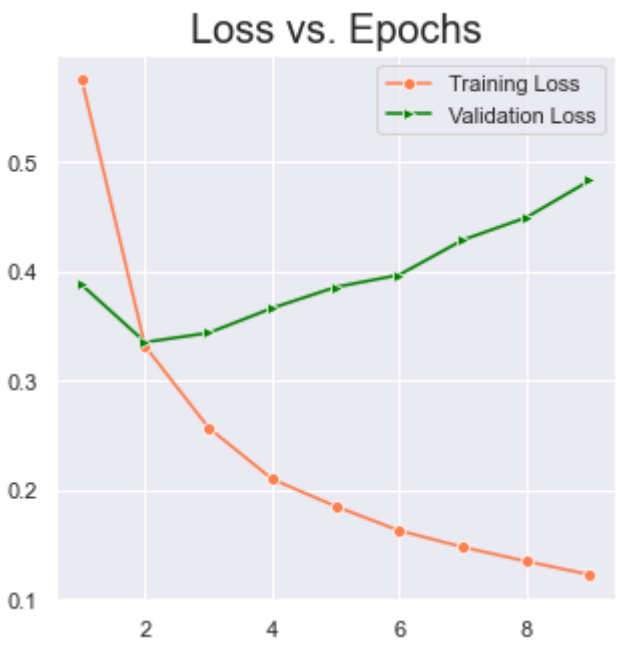
```
53/53 [==============================] - 0s 3ms/step - loss: 0.3555 - accuracy: 0.8577
Testing Accuracy:  85.77356338500977
```

- **Use Tfidf vectors instead of Embedding Layer + Feature Selection + Stratified KFold Training**

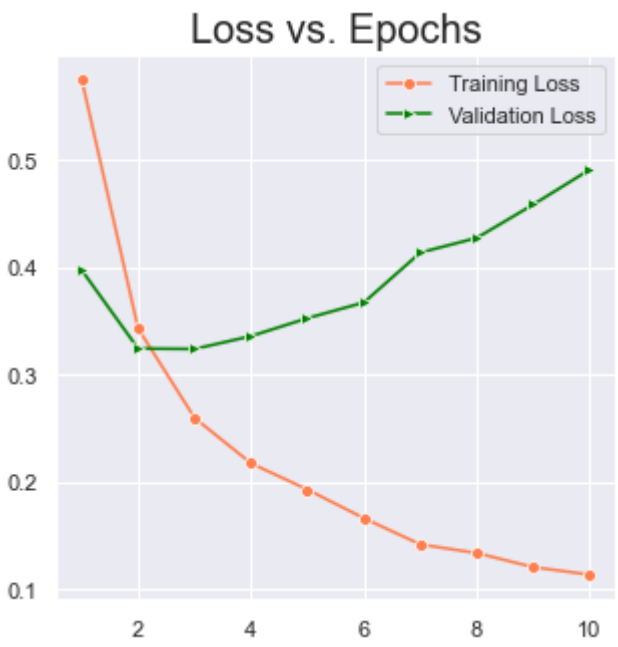
In [74]:
```
from pathlib import Path
from sklearn.model_selection import StratifiedKFold
from sklearn.feature_selection import SelectKBest, f_classif
from sklearn.feature_extraction.text import TfidfVectorizer


def get_model_name(k):
    return f'model_{k}.h5'

NUM_SPLITS = 6
EPOCHS = 20
save_dir = Path('./models/binary_classfiier/dl/merged_descr')
fold_var = 1
NGRAM_RANGE = (1, 2)
TOP_K = 10000
TOKEN_MODE = 'word'
MIN_DOC_FREQ = 2
NUM_CLASSES = 2

kwargs = {
        'ngram_range' : NGRAM_RANGE,
        'dtype' : 'int32',
        'strip_accents' : 'unicode',
        'decode_error' : 'replace',
        'analyzer' : TOKEN_MODE,
        'min_df' : MIN_DOC_FREQ
    }

val_accs = []
skf = StratifiedKFold(n_splits=NUM_SPLITS, shuffle=True, random_state=0)

for train_indices, test_indices in skf.split(X, y):
    X_train_split, X_test_split = X[train_indices], X[test_indices]
    y_train_split, y_test_split = y[train_indices], y[test_indices]
    vectorizer = TfidfVectorizer(**kwargs)
    X_train_vec = vectorizer.fit_transform(X_train_split)
    X_test_vec = vectorizer.transform(X_test_split)
    print(f"\nTrain dataset shape: {X_train_vec.shape}, \nTest dataset shape: {X_test_vec.shape
```

```
    selector = SelectKBest(f_classif, k=min(TOP_K, X_train_vec.shape[1]))
    selector.fit(X_train_vec, y_train_split)
    X_train_vec = selector.transform(X_train_vec).astype('float32')
    X_test_vec = selector.transform(X_test_vec).astype('float32')
    X_train_vec = X_train_vec.toarray()
    X_test_vec = X_test_vec.toarray()

    print(f"\nFeatures Train dataset shape: {X_train_vec.shape}, \nFeaturesTest dataset shape:
    model_ = None
    model_ = Sequential([
        Dense(20, activation='relu', input_shape=X_train_vec.shape[1:]),
        Dropout(0.5),
        Dense(20, activation='relu'),
        Dropout(0.5),
        Dense(1, activation='sigmoid')
    ])

    model_.compile(
        loss = 'binary_crossentropy',
        optimizer = 'adam',
        metrics = ['accuracy']
    )

    checkpoint = tensorflow.keras.callbacks.ModelCheckpoint(save_dir / get_model_name(fold_var)
                                                            monitor='val_accuracy',
                                                            verbose=1,
                                                            save_best_only=True,
                                                            mode='max')

    h_ = model_.fit(
            X_train_vec, y_train_split,
            validation_data = (X_test_vec, y_test_split),  # do not use the test data for valid
            epochs = EPOCHS,
            callbacks = [checkpoint, early_stop],
            verbose = False)

    model_.load_weights(save_dir / get_model_name(fold_var))
    plot_learning_curve(h_)
    loss, acc = model_.evaluate(X_test_vec, y_test_split)
    print("Testing Accuracy: ", acc*100)
    val_accs.append(acc)
    tensorflow.keras.backend.clear_session()
    fold_var += 1
```

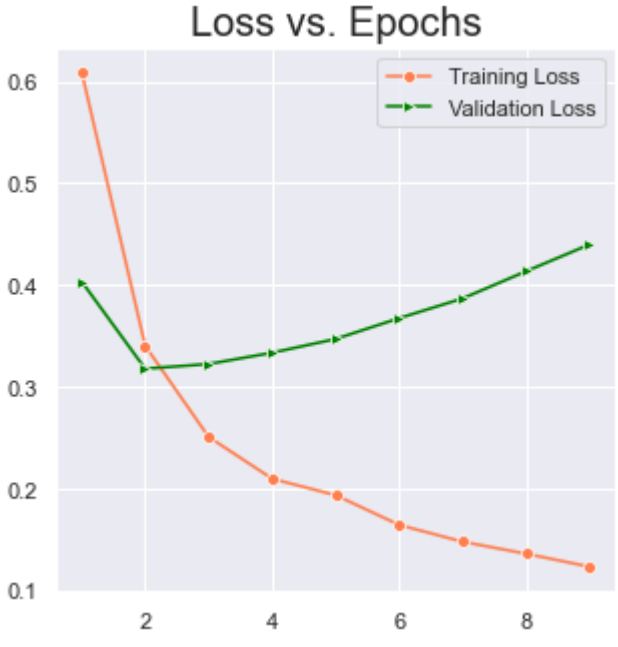
```
Train dataset shape: (7026, 17394),
Test dataset shape: (1406, 17394)

Features Train dataset shape: (7026, 10000),
FeaturesTest dataset shape: (1406, 10000)

Epoch 00001: val_accuracy improved from -inf to 0.85064, saving model to models\binary_classfii
er\dl\merged_descr\model_1.h5

Epoch 00002: val_accuracy improved from 0.85064 to 0.85989, saving model to models\binary_class
fiier\dl\merged_descr\model_1.h5

Epoch 00003: val_accuracy did not improve from 0.85989

Epoch 00004: val_accuracy improved from 0.85989 to 0.86060, saving model to models\binary_class
fiier\dl\merged_descr\model_1.h5

Epoch 00005: val_accuracy did not improve from 0.86060

Epoch 00006: val_accuracy did not improve from 0.86060

Epoch 00007: val_accuracy did not improve from 0.86060

Epoch 00008: val_accuracy did not improve from 0.86060

Epoch 00009: val_accuracy did not improve from 0.86060
```




```
44/44 [==============================] - 0s 7ms/step - loss: 0.3662 - accuracy: 0.8606
Testing Accuracy:  86.05974316596985

Train dataset shape: (7026, 16511),
Test dataset shape: (1406, 16511)

Features Train dataset shape: (7026, 10000),
FeaturesTest dataset shape: (1406, 10000)

Epoch 00001: val_accuracy improved from -inf to 0.84068, saving model to models\binary_classfii
er\dl\merged_descr\model_2.h5

Epoch 00002: val_accuracy improved from 0.84068 to 0.86060, saving model to models\binary_class
fiier\dl\merged_descr\model_2.h5

Epoch 00003: val_accuracy improved from 0.86060 to 0.86415, saving model to models\binary_class
fiier\dl\merged_descr\model_2.h5

Epoch 00004: val_accuracy did not improve from 0.86415

Epoch 00005: val_accuracy did not improve from 0.86415

Epoch 00006: val_accuracy did not improve from 0.86415

Epoch 00007: val_accuracy did not improve from 0.86415

Epoch 00008: val_accuracy did not improve from 0.86415

Epoch 00009: val_accuracy did not improve from 0.86415

Epoch 00010: val_accuracy did not improve from 0.86415
```

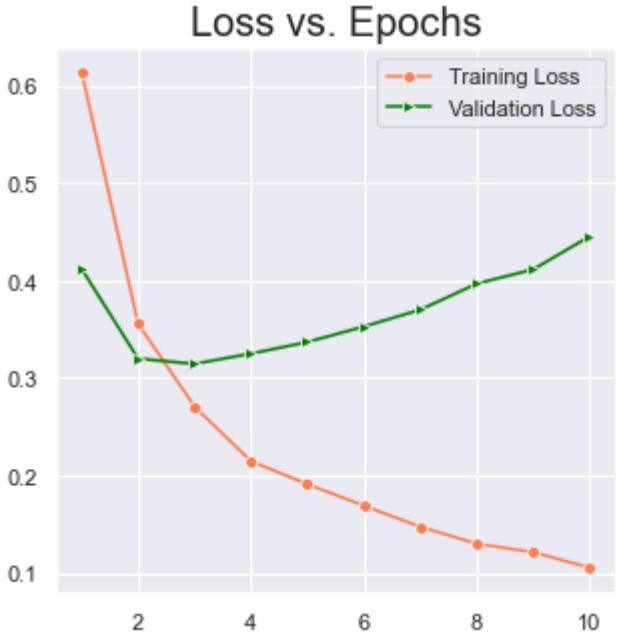



```
44/44 [==============================] - 0s 6ms/step - loss: 0.3239 - accuracy: 0.8642
Testing Accuracy:  86.41536235809326

Train dataset shape: (7027, 17654),
Test dataset shape: (1405, 17654)

Features Train dataset shape: (7027, 10000),
FeaturesTest dataset shape: (1405, 10000)

Epoch 00001: val_accuracy improved from -inf to 0.84413, saving model to models\binary_classfii
er\dl\merged_descr\model_3.h5

Epoch 00002: val_accuracy improved from 0.84413 to 0.85765, saving model to models\binary_class
fiier\dl\merged_descr\model_3.h5

Epoch 00003: val_accuracy did not improve from 0.85765

Epoch 00004: val_accuracy did not improve from 0.85765

Epoch 00005: val_accuracy did not improve from 0.85765

Epoch 00006: val_accuracy did not improve from 0.85765

Epoch 00007: val_accuracy did not improve from 0.85765

Epoch 00008: val_accuracy did not improve from 0.85765

Epoch 00009: val_accuracy did not improve from 0.85765
```



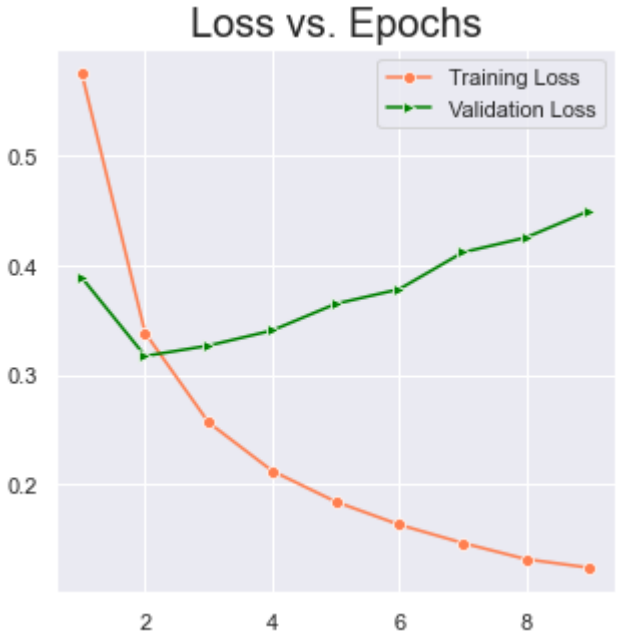

```
44/44 [==============================] - 0s 3ms/step - loss: 0.3175 - accuracy: 0.8577
Testing Accuracy:  85.76512336730957
```

```
Train dataset shape: (7027, 16823),
Test dataset shape: (1405, 16823)

Features Train dataset shape: (7027, 10000),
FeaturesTest dataset shape: (1405, 10000)

Epoch 00001: val_accuracy improved from -inf to 0.84342, saving model to models\binary_classfii
er\dl\merged_descr\model_4.h5

Epoch 00002: val_accuracy improved from 0.84342 to 0.84840, saving model to models\binary_class
fiier\dl\merged_descr\model_4.h5

Epoch 00003: val_accuracy improved from 0.84840 to 0.85480, saving model to models\binary_class
fiier\dl\merged_descr\model_4.h5

Epoch 00004: val_accuracy improved from 0.85480 to 0.85907, saving model to models\binary_class
fiier\dl\merged_descr\model_4.h5

Epoch 00005: val_accuracy improved from 0.85907 to 0.85979, saving model to models\binary_class
fiier\dl\merged_descr\model_4.h5

Epoch 00006: val_accuracy did not improve from 0.85979

Epoch 00007: val_accuracy did not improve from 0.85979

Epoch 00008: val_accuracy did not improve from 0.85979

Epoch 00009: val_accuracy did not improve from 0.85979

Epoch 00010: val_accuracy did not improve from 0.85979
```

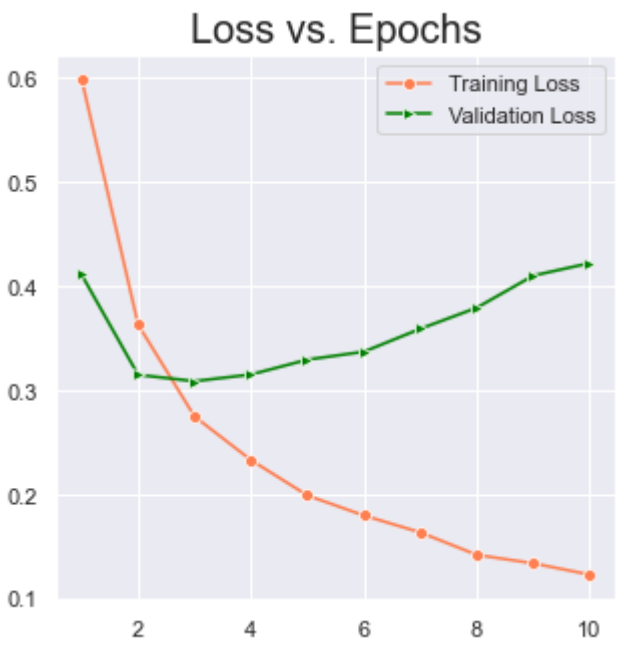



```
44/44 [==============================] - 0s 3ms/step - loss: 0.3369 - accuracy: 0.8598
Testing Accuracy:  85.97864508628845

Train dataset shape: (7027, 17284),
Test dataset shape: (1405, 17284)

Features Train dataset shape: (7027, 10000),
FeaturesTest dataset shape: (1405, 10000)

Epoch 00001: val_accuracy improved from -inf to 0.85979, saving model to models\binary_classfii
er\dl\merged_descr\model_5.h5

Epoch 00002: val_accuracy improved from 0.85979 to 0.86121, saving model to models\binary_class
fiier\dl\merged_descr\model_5.h5

Epoch 00003: val_accuracy did not improve from 0.86121

Epoch 00004: val_accuracy improved from 0.86121 to 0.86335, saving model to models\binary_class
fiier\dl\merged_descr\model_5.h5

Epoch 00005: val_accuracy did not improve from 0.86335
```

```
Epoch 00006: val_accuracy did not improve from 0.86335

Epoch 00007: val_accuracy did not improve from 0.86335

Epoch 00008: val_accuracy did not improve from 0.86335

Epoch 00009: val_accuracy did not improve from 0.86335
```




```
44/44 [==============================] - 0s 8ms/step - loss: 0.3407 - accuracy: 0.8633
Testing Accuracy:  86.3345205783844

Train dataset shape: (7027, 17779),
Test dataset shape: (1405, 17779)

Features Train dataset shape: (7027, 10000),
FeaturesTest dataset shape: (1405, 10000)

Epoch 00001: val_accuracy improved from -inf to 0.85552, saving model to models\binary_classfii
er\dl\merged_descr\model_6.h5

Epoch 00002: val_accuracy improved from 0.85552 to 0.87117, saving model to models\binary_class
fiier\dl\merged_descr\model_6.h5

Epoch 00003: val_accuracy improved from 0.87117 to 0.87829, saving model to models\binary_class
fiier\dl\merged_descr\model_6.h5

Epoch 00004: val_accuracy did not improve from 0.87829

Epoch 00005: val_accuracy did not improve from 0.87829

Epoch 00006: val_accuracy did not improve from 0.87829

Epoch 00007: val_accuracy did not improve from 0.87829

Epoch 00008: val_accuracy did not improve from 0.87829

Epoch 00009: val_accuracy did not improve from 0.87829

Epoch 00010: val_accuracy did not improve from 0.87829
```




```
44/44 [==============================] - 0s 7ms/step - loss: 0.3085 - accuracy: 0.8783
Testing Accuracy:  87.82917857170105
```

In [75]:
```
print("Testing Accuracy: ", np.mean(val_accs)*100)   # average k fold accuracy
```

```
Testing Accuracy:  86.3970955212911
```

- **Metrics:**

Model	Test Accuracy
Simple Feed-Forward Net using Embedding Layer	86.25%
Feed-Forward NN + Batch Norm	83.76%
Feed-Forward NN + Dropout	85.83%
Feed-Forward NN + Pre-trained GloVe embeddings	82.75%
LSTM	65.44%
Bi-Directional LSTM	85.24%
Convolution Blocks (Dimensionality Reduction) + LSTM	84.47%
Convolution Blocks (Dimensionality Reduction) + Bi-LSTM	85.00%
Tfidf Vectors + Feature Selection + Feed-forward Neural Net	85.77%
Stratified KFold Validation + Tfidf Vectors + Feature Selection + +Feed-forward Neural Net	86.40%