Final_Project_5

March 10, 2023

Tweets Uniqueness

[1]:
```
import sys
print(sys.version)
```

```
3.8.15 | packaged by conda-forge | (default, Nov 22 2022, 08:46:39)
[GCC 10.4.0]
```

[2]:
```
spark.version
```

[2]:
```
'3.1.3'
```

[3]:
```
import pandas as pd
import numpy as np
pd.set_option('display.max_colwidth', None)
pd.reset_option('display.max_rows')
from itertools import compress
from pyspark.sql.functions import *
from pyspark.sql.types import *
import seaborn as sns
import matplotlib.pyplot as plt
warnings.filterwarnings(action='ignore')
```

[4]:
```
from pyspark.sql import SparkSession
from pyspark import SparkContext
from pyspark.sql import SQLContext
from pyspark.sql import Row
from pyspark.sql.functions import col
```

[5]:
```
spark.conf.set("spark.sql.repl.eagerEval.enabled",True)
```

[6]:
```
%%time
twitter = spark.read.parquet('gs://chen26-bdp/original_data')
```

```
CPU times: user 9.06 ms, sys: 2.49 ms, total: 11.5 ms
Wall time: 8.36 s
```

```
23/03/10 02:42:39 WARN org.apache.spark.sql.catalyst.util.package: Truncated the
string representation of a plan since it was too large. This behavior can be
adjusted by setting 'spark.sql.debug.maxToStringFields'.
```

[7]:
```
keywords = ['college', 'high', 'university', 'students'
            , 'public', 'private', 'secondary', 'primary', 'education',␣
 ↪'undergraduate','graduate']
#filter out rows that do not contain words in keywords
twitter = twitter.withColumn('lower', lower(col('text')))
filter_twitter = twitter.filter(col('lower').rlike('|'.join(keywords)))

twitter_eng = filter_twitter.filter(col('lang') == 'en')
from pyspark.sql import functions as F
from pyspark.sql import types as t
from pyspark.sql.types import ArrayType, IntegerType, BooleanType

eng_ord=F.udf(lambda x: [ord(a) for a in x],t.ArrayType(IntegerType()))

def english_filter(x):
    for index in range(len(x)):
        if x[index] > 128:
            return False
        else:
            return True

filter_udf = F.udf(english_filter, BooleanType())
tweets = twitter_eng.filter(filter_udf(eng_ord('text')) == True)
```

Description

[8]:
```
!pip uninstall -y nltk
!pip install nltk --upgrade --no-cache-dir
```

```
Found existing installation: nltk 3.6.4
Uninstalling nltk-3.6.4:
  Successfully uninstalled nltk-3.6.4
WARNING: Running pip as the 'root' user can result in broken permissions
and conflicting behaviour with the system package manager. It is recommended to
use a virtual environment instead: https://pip.pypa.io/warnings/venv

Collecting nltk
  Downloading nltk-3.8.1-py3-none-any.whl (1.5 MB)
                             1.5/1.5 MB
26.1 MB/s eta 0:00:00a 0:00:01
Requirement already satisfied: tqdm in
/opt/conda/miniconda3/lib/python3.8/site-packages (from nltk) (4.64.1)
Requirement already satisfied: click in
```

```
/opt/conda/miniconda3/lib/python3.8/site-packages (from nltk) (7.1.2)
Collecting regex>=2021.8.3
  Downloading
regex-2022.10.31-cp38-cp38-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (772
kB)
                           772.3/772.3 kB
232.0 MB/s eta 0:00:00
Requirement already satisfied: joblib in
/opt/conda/miniconda3/lib/python3.8/site-packages (from nltk) (1.2.0)
Installing collected packages: regex, nltk
  Attempting uninstall: regex
    Found existing installation: regex 2021.4.4
    Uninstalling regex-2021.4.4:
      Successfully uninstalled regex-2021.4.4
Successfully installed nltk-3.8.1 regex-2022.10.31
WARNING: Running pip as the 'root' user can result in broken permissions
and conflicting behaviour with the system package manager. It is recommended to
use a virtual environment instead: https://pip.pypa.io/warnings/venv
```

[8]:
```
import re
import nltk
nltk.download('stopwords')
from nltk.corpus import stopwords
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data…
[nltk_data]   Package stopwords is already up-to-date!
```

[35]:
```
%%time
influencer = spark.read.parquet('gs://chen26-bdp/infl')
```

```
CPU times: user 3.1 ms, sys: 466 µs, total: 3.57 ms
Wall time: 386 ms
```

[13]:
```
u_df = influencer.filter(col('user_descrip').isNotNull())
```

[15]:
```
from pyspark.sql import functions as F
from pyspark.sql import types as t
from pyspark.sql.types import ArrayType, IntegerType, BooleanType

eng_ord=F.udf(lambda x: [ord(a) for a in x],t.ArrayType(IntegerType()))

def english_filter(x):
    for index in range(len(x)):
        if x[index] > 128:
            return False
```

```
        else:
            return True

filter_udf = F.udf(english_filter, BooleanType())
user_filt = u_df.filter(filter_udf(eng_ord('user_descrip')) == True)
```

[17]:
```
descr = user_filt.select('user_descrip')

d = descr.rdd.map(lambda x : x['user_descrip']).filter(lambda x: x is not None)

StopWords = stopwords.words("english")

tokens = d\
    .map( lambda document: document.strip().lower())\
    .map( lambda document: re.split(" ", document))\
    .map( lambda word: [x for x in word if x.isalnum()])\
    .map( lambda word: [x for x in word if len(x) > 3] )\
    .map( lambda word: [x for x in word if x not in StopWords])\
    #.zipWithIndex()
```

[18]:
```
tokens.take(5)
```

[18]:
```
[['army', 'armed', 'stand'],
 ['political',
  'science',
  'papers',
  'editor',
  'jjps',
  'board',
  'member',
  'ajcp'],
 ['political', 'analyst'],
 ['like',
  'scotland',
  'hibs',
  'andy',
  'murray',
  'tennis',
  'biden',
  'come',
  'ukraine',
  'could',
  'possibly'],
 ['plug', 'ward', 'theatre', 'data', 'bundles', 'partnerships']]
```

[20]:
```
descrCounts = tokens.flatMap(lambda x: x) \
    .map(lambda x: (x, 1)) \
    .reduceByKey(lambda x, y: x+y) \
    .map(lambda x:(x[1],x[0]))
dCountsSorted = descrCounts.sortByKey(ascending=False)
```

[27]:
```
d = dCountsSorted.map(lambda x : (x[0], x[1]))\
    .toDF(("count", "word"))
d.show(100)
```

```
+-----+-------------+
|count|         word|
+-----+-------------+
|45442|       school|
|36959|     official|
|32227|         high|
|32210|      twitter|
|28198|      account|
|23505|   university|
|22357|         love|
|19084|       sports|
|18511|      college|
|17831|         news|
|17030|        state|
|15803|    education|
|15395|        proud|
|15057|      student|
|14883|       follow|
|13930|       social|
|13691|     football|
|13475|        coach|
|13369|         life|
|13212|       former|
|13135|        views|
|12651|       public|
|12498|       tweets|
|11872|    community|
|11833|         like|
|11411|       people|
|10676|     director|
|10599|         team|
|10546|       health|
|10506|     research|
|10345|        media|
|10333|        world|
| 9921|         head|
```

9702	make
9594	class
9475	teacher
9359	basketball
9291	students
9284	business
9198	opinions
9114	writer
8896	national
8661	best
8613	member
8413	lover
8341	things
8284	2023
8231	good
8140	free
8081	since
8034	music
7982	science
7971	time
7809	political
7757	professor
7641	live
7630	page
7598	support
7521	home
7487	working
7419	learning
7268	human
7217	author
7195	retired
6993	content
6885	always
6846	every
6751	first
6617	professional
6612	work
6579	digital
6508	baseball
6391	father
6357	district
6145	also
6113	assistant
6044	help
5917	founder
5897	city
5811	black
5780	living

```
| 5761|     personal|
| 5744|     american|
| 5724|       editor|
| 5720|  development|
| 5580|international|
| 5534|       artist|
| 5516|         alum|
| 5509|      schools|
| 5434|    president|
| 5426|      history|
| 5425|        never|
| 5362|         real|
| 5362|        south|
| 5361|      husband|
| 5356|     reporter|
| 5336|       county|
| 5296|        local|
| 5267|        years|
| 5239|       global|
+-----+-------------+
only showing top 100 rows
```

[9]:
```
# categorize (roughly)
sport =␣
 ↪['baseball','basketball','tennis','team','coach','football','player','professional','espn',
        'season','nba','nfl','nhl','mlb']
news = ['abc','nbc','fox','cnn','bbc','news','media','reporter','editor']
gov =␣
 ↪['president','potus','biden','senator','congress','vp','flotus','chairman','minister']
celeb =␣
 ↪['musician','author','writer','actor','director','youtuber','instagram','entrepreneur','act
edu =␣
 ↪['university','school','professor','college','education','history','learning']
```

[45]:
```
influencer = influencer.withColumn('group', \
                      F.when((col('user_descrip').rlike('|'.join(sport))) |␣
 ↪(col('user_name').rlike('|'.join(sport))), 'Sports')\
                       .when((col('user_descrip').rlike('|'.join(news))) |␣
 ↪(col('user_name').rlike('|'.join(news))), 'News')\
                       .when((col('user_descrip').rlike('|'.join(gov))) |␣
 ↪(col('user_name').rlike('|'.join(gov))), 'Gov')\
                       .when((col('user_descrip').rlike('|'.join(edu))) |␣
 ↪(col('user_name').rlike('|'.join(edu))), 'Edu')\
                       .when((col('Max_reach') >= 100000)|(col('user_descrip').
 ↪rlike('|'.join(celeb))) | (col('user_name').rlike('|'.join(celeb))),␣
 ↪'Celebrity')
```

```
                    .otherwise('Other'))
```

[46]:
```
influencer
```

[46]:
```
+-----+------------------+-------------+---------+------------------------+--
-------+---------+
|count|           user_id|    user_name|Max_reach|
user_descrip|total_rct|    group|
+-----+------------------+-------------+---------+------------------------+--
-------+---------+
|    4|         484980949|  000Dillon000|     3504|     Independent, Chri…|
3|    Other|
|    2|1533841444225839104|    000Shub000|        1|                    null|
1|    Other|
|    1|         129785606|   000fukumoto|     3078|     Political science…|
8|    Other|
|    1|         348591369|     001Sardar|      680|                    null|
1|    Other|
|    1|        4636687154|       0027Woo|    58513|      PROBLEMATIC CON…|
15|    Other|
|  132|        2583636176|003rohitsunami|      211|                    null|
458|    Other|
|    1|1399826340099112964|   007Plangzak|       77|     Journalist, photo…|
4|Celebrity|
|    3|          43588936|      007__NIL|    12616|     I Like Scotland &…|
2|   Sports|
|    2|1317159854977462274|007enterprises|       78|     Your plug for War…|
1|    Other|
|    2|1286450080485126146|     007girl12|      457|     Child of God. Wal…|
2|    Other|
|    5|         276525537|        007mss|     3898|     Conservative, Chr…|
9|    Other|
|    2| 994464446486532097|   007saifalam|      498|     A Civil Engineer …|
1|    Other|
|    1|1557695863883014145|     009Ahmadi|       11|                    null|
4|    Other|
|    1|1395721565019459589|       00_EM01|     1047|     Too ambitious to …|
14|    Other|
|    1|        1396776462|       00daize|      527|     22 • nicki  • no…|
3|   Sports|
|    1|         246031716|   00nicolette|      571|     I'm So Meta, Even…|
6|    Other|
|    8| 726104454982836224|     00xxvv650|     6547|              |…
|121    Other|
|    5|1582606339330768898|        01DuaF|      389|                    null|
13|    Other|
|    1|1286742038923313152|     01rinette|      876|     intp   r…|
```

```
113|    Other|
|    2|         4613659461|       01sth02|    18440|     person / ltd. edi…|
11|      Edu|
+-----+-------------------+-------------+---------+-----------------------+--
-------+---------+
only showing top 20 rows
```

[47]:
```
influencer.select('group').groupby('group').count()
```

[47]:
```
+---------+------+
|    group| count|
+---------+------+
|   Sports| 58878|
|    Other|773736|
|      Edu| 49720|
|      Gov|  2534|
|Celebrity| 38575|
|     News| 35523|
+---------+------+
```

[ ]:
```
# focus on gov, edu, news
```

[48]:
```
influencer.filter(col('group') == 'Edu').orderBy('Max_reach', ascending = False)
```

[48]:
```
+-----+-------------------+--------------+---------+--------------------+------
---+-----+
|count|            user_id|     user_name|Max_reach|
user_descrip|total_rct|group|
+-----+-------------------+--------------+---------+--------------------+------
---+-----+
|    1|          369583954|TheNotoriousMMA|  9753739|Two division UFC …|
70|  Edu|
|    2|           17057271|     metmuseum|  4316369|Explore 5,000 yea…|
79|  Edu|
|    1|1349154719386775552|        FLOTUS|  3945634|First Lady of the…|
26422|  Edu|
|    5|          204151028|      DepEd_PH|  3738792|The executive age…|
56|  Edu|
|    3|           14592723|    MayoClinic|  2060614|An integrated cli…|
29|  Edu|
|    1|          171165627|    NBAHistory|  1769014|The history of th…|
411|  Edu|
|   19|           18247062| JudicialWatch|  1759738|A conservative no…|
```

```
5155|  Edu|
|   17|          148529707|        RBReich|  1585497|Berkeley professo…|
175297|  Edu|
|   17|           39585367|        Harvard|  1550580|Devoted to excell…|
4484|  Edu|
|    6|         1180533698|      fasc1nate|  1536303|Posting interesti…|
27384|  Edu|
|    2|          252018855|SudhanshuTrived|  1355422|Member of Rajya S…|
611|  Edu|
|   28|           48693594|  TeachForIndia|  1329574|Teach For India i…|
29|  Edu|
|    3|           15537451|     SLAMonline|  1323577|RESPECT THE GAME…|
414|  Edu|
|    9|           15460048|            MIT|  1260888|The Massachusetts…|
1056|  Edu|
|    2| 897843002785374210|Dr_YasminRashid|  1174929|Provincial Minist…|
1692|  Edu|
|    1|1444222873590243332| Letmespeak_org|  1143477|  Join our lan…|
7|  Edu|
|   53|1046084008742801408| collegemenfess|  1129750|Tempat nongkrong …|
12613|  Edu|
|    1|          153784150|    NASAhistory|  1067152|This is the NASA …|
173|  Edu|
|   10|          850107536|        codeorg|  1030564|http://Code.org i…|
1983|  Edu|
|    2|           25657401|   SarahBrownUK|  1000667|Chair @Theirworld…|
98|  Edu|
+-----+-------------------+---------------+---------+--------------------+------
---+-----+
only showing top 20 rows
```

[ ]:
```
# tweets rough categorize
```

[10]:
```
t_group = tweets.
 ↪select('user','created_at','extended_tweet','retweeted_status','text','id','retweeted_from'
t_group = t_group.withColumn("user_id", col("user").getItem("id")).\
            withColumn('user_name', col('user').getItem('screen_name')).\
            withColumn('user_descrip',col('user').getItem('description')).\
            withColumn('user_followerct',col('user').
 ↪getItem('followers_count')).\
            withColumn('verify_status', col('user').getItem('verified')).\
            select('user_id', 'user_name',␣
 ↪'user_descrip','user_followerct','verify_status',
          ␣
 ↪'created_at','text','extended_tweet','retweeted_status','id','retweeted_from')
t_group = t_group.withColumn('retweet_ct', col('retweeted_status').
 ↪getItem('retweet_count')).\
```

```
            withColumn('retweet', col('retweeted_status').getItem('retweeted')).
↪drop('retweeted_status')
```

[11]:
```
t_group = t_group.withColumn('group', \
                F.when(col('verify_status') == 'false', 'Other')
                 .when((col('user_descrip').rlike('|'.join(sport))) |␣
↪(col('user_name').rlike('|'.join(sport))), 'Sports')\
                 .when((col('user_descrip').rlike('|'.join(news))) |␣
↪(col('user_name').rlike('|'.join(news))), 'News')\
                 .when((col('user_descrip').rlike('|'.join(gov))) |␣
↪(col('user_name').rlike('|'.join(gov))), 'Gov')\
                 .when((col('user_descrip').rlike('|'.join(edu))) |␣
↪(col('user_name').rlike('|'.join(edu))), 'Edu')\
                 .when((col('user_followerct') >= 100000)|(col('user_descrip').
↪rlike('|'.join(celeb))) | (col('user_name').rlike('|'.join(celeb))),␣
↪'Celebrity')
                 .otherwise('Other'))
```

[12]:
```
edu_text = t_group.filter(col('group') == 'Edu').
↪select('id','user_id','user_name','user_descrip', 'text','group','retweet')
```

[60]:
```
edu_text
```

[60]:
```
+------------------+-----------------+--------------+--------------------+---
-----------------+-----+-------+
|                id|          user_id|     user_name|        user_descrip|
text|group|retweet|
+------------------+-----------------+--------------+--------------------+---
-----------------+-----+-------+
|1604280409147248640|         15537451|    SLAMonline|RESPECT THE GAME…|RT
@SLAMonline: I…|  Edu|  false|
|1610637827183632384|       2284718570|       UjuAnya|Professor. Field:…|RT
@NaijaFlyingDr…|  Edu|  false|
|1610637968665968641|        104985029|SchoolChoiceNow|American
Federati…|America's educati…|  Edu|   null|
|1610638319091650560|         80128797| VizcayaMuseum|Preserving the es…|RT
@Everglades_La…|  Edu|  false|
|1520045643179249665|        396752631|    ashleynmcb|covering Oakland
…|Catie Tombs is a …|  Edu|   null|
|1550120060781350912|780206406024691712|   neal_katyal|Supreme Court law…|So
excited that @…|  Edu|   null|
|1550120229769842689|        109608297|
MaudMaron|https://lnk.bio/m…|Healthy athletes …|  Edu|   null|
|1547724056526331912|         14416109|alexanderrusso|Founder of @thegr…|RT
@akilbello: PS…|  Edu|  false|
```

```
|1557721016713023489|         276989862|    wrightstate|Named for Ohio's
…|Safety first! The…|  Edu|   null|
|1557721048111529984|          17724276|            ucu|University and Co…|We
have hundreds …|  Edu|   null|
|1528699737414975489|          54626407| merrillcollege|The Philip Merril…|RT
@UMDsportsprof…|  Edu|  false|
|1578377103061372931|         320972641|  WichitaUSD259|The Wichita
Publi…|Join us at 6:30 p…|  Edu|   null|
|1549016279570325508|          17240179|   bereacollege|No tuition since
…|Berea College Bio…|  Edu|   null|
|1549016307173056512|         245499504|         Wonkhe|Home of the
highe…|Job: Deputy Vice-…|  Edu|   null|
|1549016385883459587|          29534841|        cfbhall|The Chick-fil-A
C…|@RPIII_Sports @Ga…|  Edu|   null|
|1516883139947208704|        1265114413|        UMKines|An international
…|Congrats to Hanna…|  Edu|   null|
|1524123868872462336|          15967775|       calstate|23 campuses. One …|RT
@SFSU: Startin…|  Edu|  false|
|1567913278780706822|          14311436|     cornellsun|Founded in 1880,
…|Since the closure…|  Edu|   null|
|1567913634671607808|          22495620| browardschools|Broward County
Pu…|Western High Scho…|  Edu|   null|
|1567913706927050752|           4268561|   jendeaderick|DED-er-ick. Autho…|RT
@aswinn: I'm v…|  Edu|  false|
+-------------------+-----------------+---------------+--------------------+---
-----------------+-----+-------+
only showing top 20 rows
```

[16]:
```
#edu original posts
ori_text = edu_text.filter(col('retweet').isNull())
```

[62]:
```
ori_text.count()
```

[62]:
```
24919
```

[17]:
```
from pyspark.sql import functions as F
from pyspark.sql import types as t
from pyspark.sql.types import ArrayType, IntegerType

df_text = ori_text.select('text')
eng_ord=F.udf(lambda x: [ord(a) for a in x],t.ArrayType(IntegerType()))

def english_filter(x):
    for index in range(len(x)):
        if x[index] > 128:
```

```
            return False
        else:
            return True

filter_udf = F.udf(english_filter, BooleanType())
text_filter = df_text.filter(filter_udf(eng_ord('text')) == True)
```

[21]:
```
text = text_filter.rdd.map(lambda x : x['text']).filter(lambda x: x is not None)

StopWords = stopwords.words("english")

tokens = text\
    .map( lambda document: document.strip().lower())\
    .map( lambda document: re.split(" ", document))\
    .map( lambda word: [x for x in word if x.isalnum()])\
    .map( lambda word: [x for x in word if len(x) > 3] )\
    .map( lambda word: [x for x in word if x not in StopWords])\
    .zipWithIndex()
```

[22]:
```
row = Row('text')
text_data=text.map(row).zipWithIndex().toDF(['text','id'])
text_data.show(5)
```

```
[Stage 5:>                                                          (0 + 1) / 1]

+--------------------+---+
|                text| id|
+--------------------+---+
|{America's educat…|  0|
|{Catie Tombs is a…|  1|
|{So excited that …|  2|
|{Healthy athletes…|  3|
|{Safety first! Th…|  4|
+--------------------+---+
only showing top 5 rows
```

[23]:
```
df_tokens = spark.createDataFrame(tokens, ["text_words",'id'])
df_tokens = df_tokens.filter(size('text_words') >= 1)
```

[67]:
```
df_tokens
```

[67]:
```
+--------------------+---+
|          text_words| id|
+--------------------+---+
|[education, syste…|  0|
|[catie, tombs, te…|  1|
|[excited, next, g…|  2|
|[healthy, athlete…|  3|
|[safety, wright, …|  4|
|[hundreds, thousa…|  5|
|[join, south, hig…|  6|
|[berea, college, …|  7|
|[deputy, universi…|  8|
|[great, radio, lo…|  9|
|[congrats, hannah…| 10|
|[since, closure, …| 11|
|[western, high, s…| 12|
|[senior, high, bo…| 13|
|[michigan, senate…| 14|
|[edition, focus, …| 15|
|[dean, college, p…| 16|
|[coming, campus, …| 17|
|[october, ramapo,…| 18|
|[richard, whitwor…| 19|
+--------------------+---+
only showing top 20 rows
```

[16]:
```
import re
import nltk
from pyspark.ml.feature import MinHashLSH
from pyspark.ml.feature import CountVectorizer,  IDF, CountVectorizerModel,␣
 ↪Tokenizer, RegexTokenizer, StopWordsRemover
from pyspark.sql import SparkSession
from pyspark import SparkContext
from pyspark.sql.functions import col
from pyspark import SparkContext
from pyspark.sql import SQLContext
from pyspark.sql import Row
import nltk
nltk.download('stopwords')
from nltk.corpus import stopwords
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data…
[nltk_data]   Package stopwords is already up-to-date!
```

[23]:
```
!pip install simhash
```

```
Collecting simhash
  Downloading simhash-2.1.2-py3-none-any.whl (4.7 kB)
```

```
Requirement already satisfied: numpy in
/opt/conda/miniconda3/lib/python3.8/site-packages (from simhash) (1.19.5)
Installing collected packages: simhash
Successfully installed simhash-2.1.2
WARNING: Running pip as the 'root' user can result in broken permissions
and conflicting behaviour with the system package manager. It is recommended to
use a virtual environment instead: https://pip.pypa.io/warnings/venv
```

[21]:
```python
from simhash import Simhash, SimhashIndex
```

[27]:
```python
vectorize = CountVectorizer(inputCol="text_words", outputCol="features",␣
↪minDF=1.0)
text_vectorize = vectorize.fit(df_tokens).transform(df_tokens)
```

[72]:
```python
text_vectorize.limit(5).toPandas()
```

[72]:
```
    text_words  \
0                                   [education, system, parents, using, school,
choice, voices]
1                    [catie, tombs, teacher, metwest, high, expulsions, metwest,
year, worried]
2                                                              [excited, next, going,
take, college]
3             [healthy, athletes, college, kids, young, people, different,
decision, regarding]
4  [safety, wright, state, university, police, department, provides, campus,
police, services]

   id  \
0   0
1   1
2   2
3   3
4   4

                    features
0  (1.0, 0.0, 0.0, 0.0, 0.0, 1.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 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, 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, 0.0, 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, 0.0, 0.0, …)
1  (0.0, 0.0, 0.0, 0.0, 1.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 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,
0.0, 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, 0.0, 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, …)
2  (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, 0.0, 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, 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, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, …)
3  (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, 0.0, 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, 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, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, …)
4  (0.0, 0.0, 0.0, 1.0, 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, 0.0, 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,
0.0, 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, 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, …)
```

[73]:

```
text_vectorize.select('features').limit(5).show(5)
```

```
[Stage 89:==================================================>(1322 + 1) / 1323]

+--------------------+
|            features|
+--------------------+
|(17595,[288,702,1…|
|(17595,[0,4,5,36,…|
|(17595,[119,124,1…|
|(17595,[3,42,101,…|
|(17595,[1,2,4,42,…|
+--------------------+
```

[10]:
```
mh = MinHashLSH(inputCol="features", outputCol="hashes", numHashTables=5)
model_text = mh.fit(text_vectorize)
text_hashed = mh.fit(text_vectorize).transform(text_vectorize)
```

[75]:
```
text_hashed.show(5)
```

```
[Stage 92:>                                                         (0 + 1) / 1]

+--------------------+---+--------------------+--------------------+
|          text_words| id|            features|              hashes|
+--------------------+---+--------------------+--------------------+
|[education, syste…|  0|(17595,[0,5,90,18…|[[4.7945584E7], […|
|[catie, tombs, te…|  1|(17595,[4,15,64,3…|[[2.56879666E8], …|
|[excited, next, g…|  2|(17595,[2,55,58,1…|[[1.1707592E8], […|
|[healthy, athlete…|  3|(17595,[2,85,146,…|[[2.99236534E8], …|
|[safety, wright, …|  4|(17595,[3,9,23,16…|[[6.56996916E8], …|
+--------------------+---+--------------------+--------------------+
only showing top 5 rows
```

[76]:
```
df_hashed_text = text_data.join(text_hashed, "id", how = 'left')
```

[ ]:
```
%%time
df_hashed_text.write.format("parquet").\
mode('overwrite').\
save('gs://chen26-bdp/text_hash')
```

[8]:
```
%%time
text_vectorize.write.format("parquet").\
mode('overwrite').\
save('gs://chen26-bdp/text_vectorize')
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
File <timed eval>:1

NameError: name 'text_vectorize' is not defined
```

[9]:
```
df_hashed_text = spark.read.parquet('gs://chen26-bdp/text_hash')
text_vectorize = spark.read.parquet('gs://chen26-bdp/text_vectorize')
```

[80]:
```
df_hashed_text
```

[80]:
```
+---+--------------------+--------------------+--------------------+------------
--------+
| id|                text|          text_words|            features|
hashes|
+---+--------------------+--------------------+--------------------+------------
--------+
|418|{A college intern…|[college,
letter,…|(17595,[2,80,138,…|[[5.75251383E8], …|
|415|{@clam_57 @FrankL…|[bailouts,
corpor…|(17595,[11,298,34…|[[5.201327E7], [5…|
|414|{Or this guy, who…|[looks, like,
hig…|(17595,[0,4,31,64…|[[4.7945584E7], […|
|427|{Dash '22, commit…|[dash,
committed,…|(17595,[5,43,550,…|[[1.75609176E8], …|
|406|{OMG. Reminds me …|[reminds,
another…|(17595,[0,4,201,2…|[[4.7945584E7], […|
|421|{Gifted education…|[gifted,
educatio…|(17595,[5,242,253…|[[1.42306314E8], …|
|412|{#UVM's Class of …|[class, 2026,
uni…|(17595,[3,25,448,…|[[8.30079935E8], …|
|417|{Kicking off a co…|[kicking,
communi…|(17595,[6,7,10,22…|[[2.30106198E8], …|
|420|{Do you want to m…|[want, make,
mark…|(17595,[2,38,63,4…|[[2.26069964E8], …|
|402|{The University o…|[university,
tole…|(17595,[3,54,87,1…|[[1.8678956E7], […|
|413|{Volunteer with u…|[volunteer,
help,…|(17595,[0,4,14,17…|[[4.7945584E7], […|
|424|{.@DrMikeHansen: …|[serving, high,
s…|(17595,[4,64,138,…|[[4.28918243E8], …|
|407|{OMG. Reminds me …|[reminds,
another…|(17595,[2,201,252…|[[7.2170851E7], […|
|430|{Welcome to all n…|[welcome,
returni…|(17595,[59,465,73…|[[3.14890701E8], …|
|405|{MI Senate passes…|[senate, passes,
…|(17595,[2,631,727…|[[1.42812809E8], …|
|434|{Choosing a major…|[choosing,
major,…|(17595,[2,55,115,…|[[1.9691946E7], […|
|404|{Thank you for th…|[thank, kind,
uni…|(17595,[3,34,57,8…|[[5.44473133E8], …|
|419|{Please help me c…|[please, help,
co…|(17595,[0,4,14,53…|[[4.7945584E7], […|
|423|{On November 18/1…|[november,
eleven…|(17595,[0,1,4,17,…|[[4.7945584E7], […|
|403|{Led by researche…|[researchers,
lou…|(17595,[113,350,4…|[[2.88639454E8], …|
+---+--------------------+--------------------+--------------------+------------
```

```
--------+
only showing top 20 rows
```

[19]:
```
jaccard_distance = 0.5

df_dups_text_50 = model_text.approxSimilarityJoin(text_hashed, text_hashed,␣
 ↪jaccard_distance).\
filter("datasetA.id < datasetB.id").select(
            col("distCol"),
             col("datasetA.id").alias("id_A"),
             col("datasetB.id").alias("id_B"),
            col('datasetA.text_words').alias('text_A'),
            col('datasetB.text_words').alias('text_B'))
```

[28]:
```
text_hashed.count()
```

[28]:
```
24880
```

[31]:
```
df_dups_text_50.limit(5).show()
```

```
[Stage 16:=====================================================>   (16 + 1) / 17]

+-------+-----+-----+--------------------+--------------------+
|distCol| id_A| id_B|              text_A|              text_B|
+-------+-----+-----+--------------------+--------------------+
|    0.0|14591|19340|[strongest, argum…|[strongest, argum…|
|    0.0| 6249|19928| [college, football]| [college, football]|
|    0.0| 4080| 5520|[opponents, schoo…|[opponents, schoo…|
|    0.0| 3158| 7905|[recent, study, l…|[recent, study, l…|
|    0.0|12767|19803|[homeless, makes,…|[homeless, makes,…|
+-------+-----+-----+--------------------+--------------------+
```

[22]:
```
df_dups_text_50 = df_dups_text_50.toPandas()
```

[27]:
```
df_dups_text_50
```

[27]:

	distCol	id_A	id_B
0	0.000000	3911	6356
1	0.000000	5933	11993
2	0.000000	14636	17149
3	0.357143	3911	14373

```
4     0.000000   5933   6477
…        …    …    …
6567  0.000000   3865  23105
6568  0.444444   4624  19065
6569  0.000000   5482   6869
6570  0.000000   1884  24269
6571  0.000000  19763  19795

       text_A  \
0     [starting, kansas, city, public, schools, board, directors, needs, join,
online, today]
1               [operating, hawaiian, schools, engaging, communities, transform,
educational]
2                            [care, solace, connects, school, staff, families,
cost, quickly]
3     [starting, kansas, city, public, schools, board, directors, needs, join,
online, today]
4               [operating, hawaiian, schools, engaging, communities, transform,
educational]
…
…
6567                           [federation, partnerships, university, london,
senate, house]
6568                              [wake, tech, graduates, earn, average, year,
high, school]
6569
[high, school]
6570                [used, college, education, solid, defense, poverty, writer,
loki, argues]
6571           [today, primary, school, national, offer, applied, online,
admissions, portal]

              text_B
0            [starting, kansas, city, public, schools, board, directors, needs,
join, online, today]
1                      [operating, hawaiian, schools, engaging, communities,
transform, educational]
2                                   [care, solace, connects, school, staff,
families, cost, quickly]
3     [kansas, city, public, schools, board, directors, needs, join, person,
online, give, feedback]
4                      [operating, hawaiian, schools, engaging, communities,
transform, educational]
…
…
6567                                  [federation, partnerships, university,
london, senate, house]
```

```
6568                                [high, school, graduates, average, earn,
annual, average, annual]
6569
[high, school]
6570                        [used, college, education, solid, defense, poverty,
writer, loki, argues]
6571                  [today, primary, school, national, offer, applied, online,
admissions, portal]

[6572 rows x 5 columns]
```

[15]:
```
sample = text_hashed.limit(10000)
```

[16]:
```
jaccard_distance = 0.5

df_dups_text_05 = model_text.approxSimilarityJoin(sample, sample,␣
 ↪jaccard_distance).\
filter("datasetA.id < datasetB.id").select(
            col("distCol"),
             col("datasetA.id").alias("id_A"),
             col("datasetB.id").alias("id_B"))
            #col('datasetA.text_words').alias('text_A'),
            #col('datasetB.text_words').alias('text_B'))
```

[17]:
```
records = sample.count()
dups_50_text_distinct = df_dups_text_05.select('id_A').distinct()
dups_50_text = dups_50_text_distinct.count()
uniques = records - dups_50_text

print ('Total records: ', records)
print ('Duplicate titles based on {', jaccard_distance, '} jaccard distance: ',␣
 ↪dups_50_text)
print ('Unique titles based on {', jaccard_distance, '} jaccard distance: ',␣
 ↪jaccard_distance, ': ', uniques)
```

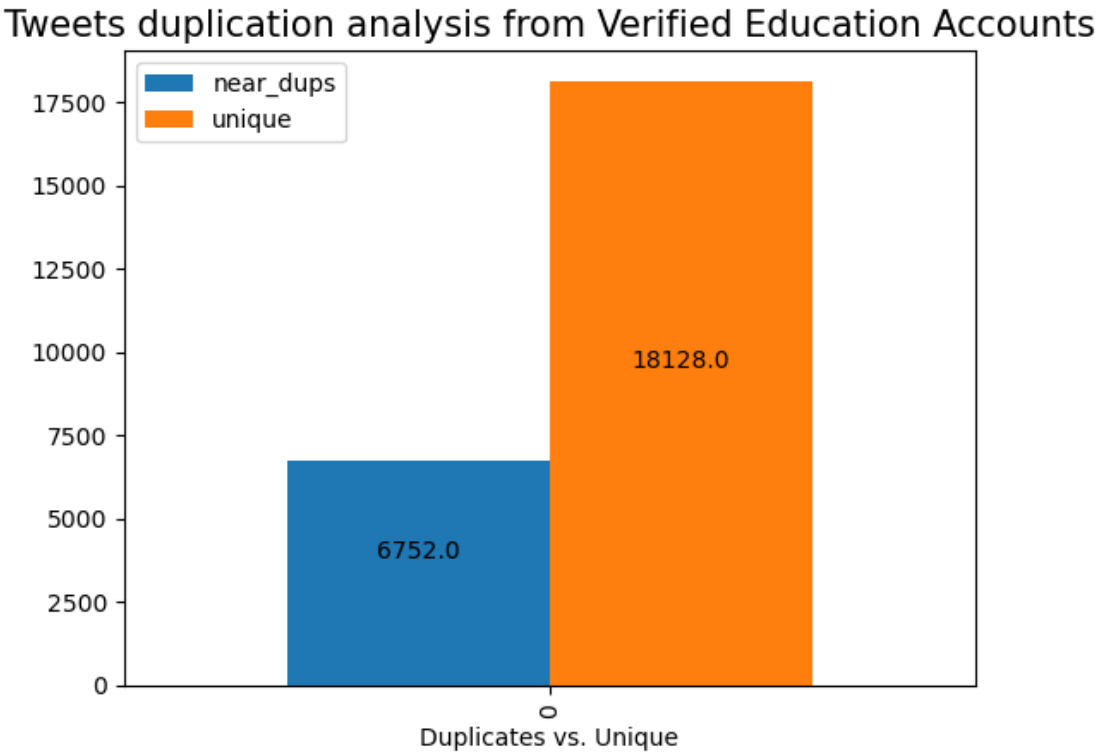
```
[Stage 18:>                                                         (0 + 1) / 1]

Total records:  10000
Duplicate titles based on { 0.5 } jaccard distance:  598
Unique titles based on { 0.5 } jaccard distance:  0.5 :  9402
```

[32]:
```
dups_df = pd.DataFrame.from_dict({'near_dups': [6752], 'unique': [18128]})

ax=dups_df.plot(kind = 'bar',y=['near_dups', 'unique'], fontsize=10,␣
 ↪color=['C0', 'C1'], align='center', width=0.8, xlabel="Duplicates vs.␣
 ↪Unique")
```

```
ax.set_title('Tweets duplication analysis from Verified Education Accounts',␣
 ↪fontsize=15)
for p in ax.patches:
       ax.annotate(format(p.get_height(), '.1f'),
                   (p.get_x() + p.get_width() / 2., p.get_height()/2),
                   ha = 'center', va = 'center',
                   xytext = (0, 9),
                   textcoords = 'offset points')
```



[ ]:
```
# verified group gov account
```

[41]:
```
gov_text = t_group.filter(col('group') == 'Gov').
 ↪select('id','user_id','user_name','user_descrip', 'text','group','retweet')
ori_gov = gov_text.filter(col('retweet').isNull())
```

[42]:
```
ori_gov.count()
```

[42]: 1036

[46]:
```
from pyspark.sql import functions as F
from pyspark.sql import types as t
from pyspark.sql.types import ArrayType, IntegerType

df_text = ori_gov.select('text')
eng_ord=F.udf(lambda x: [ord(a) for a in x],t.ArrayType(IntegerType()))

def english_filter(x):
    for index in range(len(x)):
        if x[index] > 128:
            return False
        else:
            return True

filter_udf = F.udf(english_filter, BooleanType())
text_filter = df_text.filter(filter_udf(eng_ord('text')) == True)
```

[47]:
```
text = text_filter.rdd.map(lambda x : x['text']).filter(lambda x: x is not None)

StopWords = stopwords.words("english")

tokens = text\
    .map( lambda document: document.strip().lower())\
    .map( lambda document: re.split(" ", document))\
    .map( lambda word: [x for x in word if x.isalnum()])\
    .map( lambda word: [x for x in word if len(x) > 3] )\
    .map( lambda word: [x for x in word if x not in StopWords])\
    .zipWithIndex()

row = Row('text')
text_data=text.map(row).zipWithIndex().toDF(['text','id'])
text_data.show(5)
```

```
[Stage 65:=================================================>       (7 + 1) / 8]

+--------------------+---+
|                text| id|
+--------------------+---+
|{@profmarkcollard…|  0|
|{Great day at @T…|  1|
|{Another ~$400 bi…|  2|
|{UO President Mic…|  3|
|{Per University o…|  4|
+--------------------+---+
only showing top 5 rows
```

[48]:
```
df_tokens_gov = spark.createDataFrame(tokens, ["text_words",'id'])
df_tokens_gov = df_tokens_gov.filter(size('text_words') >= 1)
```

[49]:
```
vectorize = CountVectorizer(inputCol="text_words", outputCol="features",␣
↪minDF=1.0)
text_vectorize_gov = vectorize.fit(df_tokens_gov).transform(df_tokens_gov)
```

[50]:
```
%%time
text_vectorize_gov.write.format("parquet").\
mode('overwrite').\
save('gs://chen26-bdp/text_vectorize_gov')
```

```
CPU times: user 1.06 s, sys: 254 ms, total: 1.31 s
Wall time: 6min 58s
```

[10]:
```
text_vectorize_gov = spark.read.parquet('gs://chen26-bdp/text_vectorize_gov')
```

[11]:
```
text_vectorize_gov
```

[11]:
```
+--------------------+---+--------------------+
|          text_words| id|            features|
+--------------------+---+--------------------+
|[norwalk, public,…| 57|(3078,[5,6,375],[…|
|[gets, decide, je…| 58|(3078,[2,13,206,2…|
|[henry, army, vet…| 59|(3078,[1,5,30,122…|
|[henry, army, vet…| 60|(3078,[1,5,30,122…|
|[henry, army, vet…| 61|(3078,[1,5,30,122…|
|[signed, srinivas…| 62|(3078,[2,4,128,58…|
|[says, cyber, cha…|962|(3078,[5,6,27,38,…|
|[notes, current, …|963|(3078,[1,5,36,132…|
|[invest, invest, …|964|(3078,[1,5,61,328…|
|[national, lung, …|965|(3078,[0,22,232,3…|
|[purchased, mine,…|332|(3078,[2,167,257,…|
|[thank, service, …|333|(3078,[2,13,59,54…|
|[back, late, high…|334|(3078,[1,3,11,194…|
|[colleges, reflec…|335|(3078,[12,155,500…|
|[root, cause, pro…| 12|(3078,[5,30,243,5…|
|[support, profess…| 13|(3078,[31,94,515,…|
```

```
|[quick, note, pay…| 14|(3078,[10,17,33,3…|
|[hillary, clinton…| 15|(3078,[160,201,24…|
|[honestly, little…|561|(3078,[1,3,40,86,…|
|[published, fall,…|562|(3078,[5,23,103,3…|
+--------------------+---+--------------------+
only showing top 20 rows
```

[13]:
```
mh = MinHashLSH(inputCol="features", outputCol="hashes", numHashTables=5)
model_text_gov = mh.fit(text_vectorize_gov)
text_hashed_gov = mh.fit(text_vectorize_gov).transform(text_vectorize_gov)
```

[16]:
```
jaccard_distance = 0.5

df_dups_text_gov_50 = model_text_gov.approxSimilarityJoin(text_hashed_gov,␣
 ↪text_hashed_gov, jaccard_distance).\
filter("datasetA.id < datasetB.id").select(
            col("distCol"),
             col("datasetA.id").alias("id_A"),
             col("datasetB.id").alias("id_B"),
            col('datasetA.text_words').alias('text_A'),
            col('datasetB.text_words').alias('text_B'))
```

[19]:
```
df_dups_text_gov_50_df = df_dups_text_gov_50.toPandas()
```

[20]:
```
df_dups_text_gov_50_df
```

[20]:

	distCol	id_A	id_B	\
0	0.000000	805	939	
1	0.000000	59	61	
2	0.000000	492	677	
3	0.333333	161	943	
4	0.333333	201	943	
5	0.200000	843	914	
6	0.333333	540	943	
7	0.250000	161	566	
8	0.333333	322	943	
9	0.400000	417	1001	
10	0.000000	292	825	
11	0.000000	212	758	
12	0.000000	631	846	
13	0.000000	60	61	
14	0.000000	741	742	
15	0.250000	524	812	
16	0.333333	235	773	
17	0.000000	117	429	

```
18  0.000000    59    60
19  0.250000   161   219
20  0.400000   262   566
21  0.000000   177   178
22  0.400000   219   566
23  0.000000   960   961
24  0.000000   951  1002
25  0.000000   508   825
26  0.000000   464   509
27  0.000000   935  1010
28  0.000000   537   933
29  0.400000   219   262
30  0.333333   943  1001
31  0.000000   283   456
32  0.000000   945  1000
33  0.333333   257   943
34  0.333333   812   943
35  0.000000   292   508
36  0.250000   161   262
37  0.333333   665   943

    text_A  \
0                                      [school, high, quality, like,
international, order]
1   [henry, army, veteran, school, teacher, running, congress, northern,
believed, public]
2       [abvp, delegation, leadership, national, general, secretary, chairman,
university]
3                                                                  [college,
football, best]
4                                                                 [watch, college,
football]
5                                       [programme, organised, college,
director, general]
6                                                                 [college,
football, dying]
7                                                                  [college,
football, best]
8                                                                [months, college,
football]
9                                                    [time, bring, back, college,
football]
10
[college]
11                                          [coming, barstool, live, state,
college, ohio]
12                                  [time, inclusive, school, protections, lgbtq,
```

```
students]
13  [henry, army, veteran, school, teacher, running, congress, northern,
believed, public]
14                                  [aldin, formally, introduced, head,
designated, former]
15                                                      [college, football,
drunk, today]
16
[college, kickers]
17
[high]
18  [henry, army, veteran, school, teacher, running, congress, northern,
believed, public]
19                                                                [college,
football, best]
20                                                       [uniform, best, college,
football]
21       [khaula, sawah, president, uossm, kentucky, graduate, leading, medical,
missions]
22                                                    [college, football,
rivalries, best]
23          [least, people, killed, mostly, young, practiced, university,
entrance, exams]
24          [pleased, release, thinking, public, conversation, today, professor,
crawford]
25
[college]
26                [marischal, college, basking, summer, looking, family, kids,
head, back]
27                                  [students, sage, ridge, school, week, explore,
pleasure]
28                                             [think, india, forum, students,
national]
29                                                  [college, football,
rivalries, best]
30                                                                    [college,
football]
31                                [congratulations, organized, equity, coast,
high, school]
32                     [celebration, smoking, department, partnership,
philippine, college]
33                                                                 [college,
football, take]
34                                                               [college,
football, drunk]
35
[college]
```

```
36                                                                  [college,
football, best]
37                                                                  [like, college,
football]

    text_B
0                                    [school, high, quality, like,
international, order]
1   [henry, army, veteran, school, teacher, running, congress, northern,
believed, public]
2       [abvp, delegation, leadership, national, general, secretary, chairman,
university]
3                                                                       [college,
football]
4                                                                       [college,
football]
5                                              [programme, organised, college,
director]
6                                                                       [college,
football]
7                                                        [college, football,
fans, best]
8                                                                       [college,
football]
9                                                                   [college,
football, back]
10
[college]
11                                         [coming, barstool, live, state,
college, ohio]
12                                 [time, inclusive, school, protections, lgbtq,
students]
13  [henry, army, veteran, school, teacher, running, congress, northern,
believed, public]
14                                 [aldin, formally, introduced, head,
designated, former]
15                                                               [college,
football, drunk]
16                                                              [college,
kickers, awesome]
17
[high]
18  [henry, army, veteran, school, teacher, running, congress, northern,
believed, public]
19                                                    [college, football,
rivalries, best]
20                                                         [college, football,
```

```
fans, best]
21       [khaula, sawah, president, uossm, kentucky, graduate, leading, medical,
missions]
22                                                          [college, football,
fans, best]
23          [least, people, killed, mostly, young, practiced, university,
entrance, exams]
24          [pleased, release, thinking, public, conversation, today, professor,
crawford]
25
[college]
26                [marischal, college, basking, summer, looking, family, kids,
head, back]
27                                  [students, sage, ridge, school, week, explore,
pleasure]
28                                              [think, india, forum, students,
national]
29                                                    [uniform, best, college,
football]
30                                                               [college,
football, back]
31                                  [congratulations, organized, equity, coast,
high, school]
32                      [celebration, smoking, department, partnership,
philippine, college]
33                                                                   [college,
football]
34                                                                   [college,
football]
35
[college]
36                                                    [uniform, best, college,
football]
37                                                                   [college,
football]
```

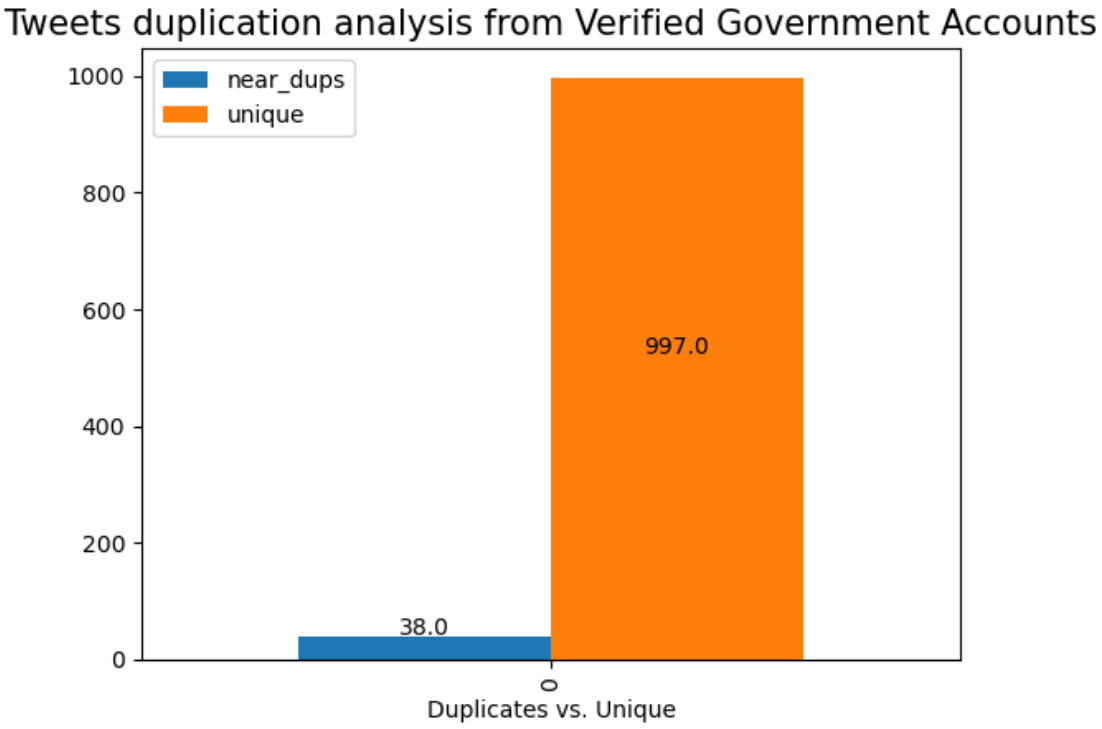
[21]:
```
text_hashed_gov.count()
```

[21]:
```
1035
```

[23]:
```
dups_df = pd.DataFrame.from_dict({'near_dups': [38], 'unique': [997]})

ax=dups_df.plot(kind = 'bar',y=['near_dups', 'unique'], fontsize=10,␣
↪color=['C0', 'C1'], align='center', width=0.8, xlabel="Duplicates vs.␣
↪Unique")
```

```
ax.set_title('Tweets duplication analysis from Verified Government Accounts',␣
 ↪fontsize=15)
for p in ax.patches:
       ax.annotate(format(p.get_height(), '.1f'),
                   (p.get_x() + p.get_width() / 2., p.get_height()/2),
                   ha = 'center', va = 'center',
                   xytext = (0, 9),
                   textcoords = 'offset points')
```



[ ]:
```
# all original text without grouping
```

[28]:
```
original = t_group.filter(col('retweet').isNull())
original = original.
 ↪select('user_id','user_name','verify_status','text','id','retweeted_from','retweet')
original = original.drop('retweet')
```

[29]:
```
original
```

[29]:
```
+------------------+--------------+-------------+--------------------------+-
-----------------+--------------------+
|           user_id|     user_name|verify_status|                      text|
```

```
id|      retweeted_from|
+------------------+--------------+------------+-------------------------+-
-----------------+--------------------+
|1101871913771692032|    SualihTemam|        false|#
#  https…|1604278587183415297|                null|
| 791367120617607168|   bluddystyles|        false|       the fact that I
g…|1604278598155702273|               null|
|         2613254236|    Rosaline536|        false|       Selling my 4x
The…|1604278614748528642|               null|
|1413848524517687296| kpopmisandrist|        false|       If you're
confusi…|1604278620993757186|               null|
|1437582409906655236| WillieETCarver|        false|       @bungalow3500
Tha…|1604278643693424643|               null|
|1449084294828371974|   Lann_Andrews|        false|       Homework |
nursin…|1604278651373199361|               null|
|1350274870827954180| prairiestatebn|        false|       Wasn't wild
about…|1604278654892183553|               null|
|1597799368262037504|   ZaxNewsStand|        false|       Former Alabama
Re…|1604278661267574785|               null|
|1422734857776533504| kevin_kershner|        false|       @YaOnlyLivvOnce
I…|1604278675108597760|               null|
|1230812852535070720|       slmjim44|        false|       @Choomb4TF
@SaSSa…|1604278675477860352|SaSSaBJJ @s1mpleO…|
| 830056431470641152| chicagohounds4|        false|       Just posted a
pho…|1604278676308299776| Grand Valley Sta…|
|          122918354|       bluesfor|        false|       @SpudRegis11
@Tgr…|1604278701407219712|Tgrace33 @LucyZel…|
|          224977551|        tdeb007|        false|       @Melmoo80
@Kirsti…|1604278736148275200|KirstiMiller30 I …|
|          192969072|     TAITCOMICS|        false|       Me in high
school…|1604278736798765056|               null|
|          147439783|       KGreco23|        false|       @miles_was_her
Ye…|1604278740065869824|               null|
|         1450384922|       SequoiaN|         true|       @Dalai_Mama_
Look…|1604278741248663552|               null|
|1595617317626564608|AmesburyBarbara|        false| #  #    The
fo…|1604278761951617024|               null|
|         2721511461|NatanielRodrig9|        false|  #
#  https://…|1604278768272838659|               null|
|1199408500801253377|        _V_C_U_|        false|       @thepeppas The
hi…|1604278778682810373|               null|
|1248887297505820678|    akinoe_ztmy|        false|#
#  https…|1604278793195003904|                null|
+------------------+--------------+------------+-------------------------+-
-----------------+--------------------+
only showing top 20 rows
```

[30]:
```
from pyspark.sql import functions as F
from pyspark.sql import types as t
from pyspark.sql.types import ArrayType, IntegerType

df_text = original.select('id', 'text')
eng_ord=F.udf(lambda x: [ord(a) for a in x],t.ArrayType(IntegerType()))

def english_filter(x):
    for index in range(len(x)):
        if x[index] > 128:
            return False
        else:
            return True

filter_udf = F.udf(english_filter, BooleanType())
text_filter = df_text.filter(filter_udf(eng_ord('text')) == True)
```

[31]:
```
d = text_filter.rdd.map(lambda x : x['text']).filter(lambda x: x is not None)
StopWords = stopwords.words("english")
# remove stop words
tokens = d\
        .map( lambda document: document.strip().lower())\
        .map( lambda document: re.sub("@[A-Za-z0-9_]+","", document))\
        .map( lambda document: re.sub(r'[^\w\s]', '', document))\
        .map( lambda document: re.split(" ", document))\
        .map( lambda word: [x for x in word if x.isalnum()])\
        .map( lambda word: [x for x in word if len(x) > 3] )\
        .map( lambda word: [x for x in word if x not in StopWords])
```

[32]:
```
t = tokens.zip(text_filter.select('id').rdd.flatMap(lambda x:x))
df_tokens = t.toDF(['tokens', 'id'])
eng = df_tokens.filter(size("tokens")>=1)
```

[64]:
```
eng
```

[64]:
```
+----------------------+------------------+
|                tokens|                id|
+----------------------+------------------+
|    [college, husband…|1604278587183415297|
|    [fact, asked, tra…|1604278598155702273|
|    [selling, 1975, t…|1604278614748528642|
|    [youre, confusing…|1604278620993757186|
|    [leftmost, sign, …|1604278643693424643|
```

```
|    [homework, nursin…|1604278651373199361|
|    [wasnt, wild, fru…|1604278654892183553|
|    [former, alabama,…|1604278661267574785|
|    [love, chicago, l…|1604278675108597760|
|    [fake, occupation…|1604278675477860352|
|    [posted, photo, g…|1604278676308299776|
|    [players, drafted…|1604278701407219712|
|    [sprinter, high, …|1604278736148275200|
|    [high, school, su…|1604278736798765056|
|    [pajama, pants, f…|1604278740065869824|
|    [look, body, spra…|1604278741248663552|
|[  , foundation…|1604278761951617024|
|    [last, adult, fam…|1604278768272838659|
|    [high, school, pe…|1604278778682810373|
|               [reveal]|1604278793195003904|
+----------------------+-------------------+
only showing top 20 rows
```

[33]:
```
import re
from pyspark.sql.functions import udf
from pyspark.sql.types import BooleanType

def filter_non_english(df, col_name):
    non_english_pattern = re.compile(r'[^\x00-\x7F]+')
    def contains_non_english_words(words):
        for word in words:
            if non_english_pattern.search(word):
                return True
        return False

    contains_non_english_udf = udf(contains_non_english_words, BooleanType())
    filtered_df = df.filter(~contains_non_english_udf(df[col_name]))

    return filtered_df

eng = filter_non_english(eng, 'tokens')
```

[19]:
```
eng
```

[19]:
```
+--------------------+-------------------+
|              tokens|                 id|
+--------------------+-------------------+
|[college, husband…|1604278587183415297|
|[youre, confusing…|1604278620993757186|
|[former, alabama,…|1604278661267574785|
```

```
|[players, drafted…|1604278701407219712|
|[sprinter, high, …|1604278736148275200|
|[pajama, pants, f…|1604278740065869824|
|[look, body, spra…|1604278741248663552|
|[last, adult, fam…|1604278768272838659|
|[high, school, pe…|1604278778682810373|
|            [reveal]|1604278793195003904|
|[hello, rate, sta…|1604278797494210560|
|[maybe, maybe, ut…|1604278802787377152|
|[push, reopen, sc…|1604278846479814656|
|[dont, want, elec…|1604278877924229121|
|     [college, grid]|1604278881187287040|
|  [drug, university]|1604278906437005312|
|[oberlin, college…|1604278926322200576|
|[bevis, stevenson…|1604278927182008320|
|[steward, harte, …|1604278954101444609|
|[qualification, n…|1604279015799681024|
+--------------------+-------------------+
only showing top 20 rows
```

[ ]:
```
eng.count()
```

[ ]:
```
13420117
```

[34]:
```
sample_text = eng.limit(10000)
```

[35]:
```
vectorize = CountVectorizer(inputCol="tokens", outputCol="features", minDF=1.0)
text_vectorize = vectorize.fit(sample_text).transform(sample_text)
mh = MinHashLSH(inputCol="features", outputCol="hashes", numHashTables=5)
text_model = mh.fit(text_vectorize)
text_hashed = mh.fit(text_vectorize).transform(text_vectorize)
```

[26]:
```
text_vectorize
```

[26]:
```
+--------------------+-------------------+--------------------+
|              tokens|                 id|            features|
+--------------------+-------------------+--------------------+
|[happened, powell…|1517190854393409537|(15641,[0,34,303,…|
|[shed, better, dr…|1517190869656293380|(15641,[0,82,83,1…|
|[public, school, …|1517190870121996288|(15641,[1,8,66,98…|
|[dont, kids, clif…|1517190905140334592|(15641,[0,10,17,1…|
|[name, given, hig…|1517190909619621896|(15641,[2,7,14,68…|
```

```
|[watching, gradua…|1517190912895557633|(15641,[1,56,73,7…|
|[name, richard, w…|1517190920818638848|(15641,[3,14,67,3…|
|[high, high, scho…|1517190954113015813|(15641,[1,2,4092]…|
|[nothing, high, s…|1517190954842607624|(15641,[1,2,157,1…|
|[dont, college, c…|1517190960769376257|(15641,[0,10,298]…|
|[nooooo, college,…|1517191018029924353|(15641,[0,4,146,2…|
|[high, schools, i…|1517191021838352384|(15641,[2,7,12616…|
|[theres, sunflowe…|1517191068160499712|(15641,[3,47,214,…|
|[ever, public, sc…|1517191070588620806|(15641,[1,8,78,83…|
|[university, stud…|1517191088980869121|(15641,[0,3,4,60,…|
|[professor, never…|1517191126288977923|(15641,[47,115,67…|
|[stophazaragenoci…|1517191161991073792|(15641,[1,2,238,2…|
|[accs, dean, stud…|1517191177774059521|(15641,[1,4,227,1…|
|[intelligent, peo…|1517191196724051971|(15641,[0,15,60,2…|
|[constantly, publ…|1517191209319546881|(15641,[1,8,40,18…|
+--------------------+-------------------+--------------------+
only showing top 20 rows
```

[ ]:
```
text_hashed
```

[ ]:
```
+--------------------+-------------------+--------------------+----------------
---+
|              tokens|                 id|            features|
hashes|
+--------------------+-------------------+--------------------+----------------
---+
|[done, college, f…|1579370883562745858|(15539,[0,77,142,…|[[4.7945584E7],
[…|
|[leave, college, …|1579370980761391107|(15539,[0,332,382…|[[4.7945584E7],
[…|
|[plssss, shame, l…|1579370983236206592|(15539,[0,1,5,23,…|[[4.7945584E7],
[…|
|[brush, means, an…|1579371025724477441|(15539,[1,2,36,15…|[[3.83000184E8],
…|
|[gilmeron, road, …|1579371039301468160|(15539,[2,858,300…|[[3.5836882E7],
[…|
|[main, reasons, b…|1579371053025210368|(15539,[11,87,227…|[[2.3237411E7],
[…|
|[keiko, torii, wo…|1579371087112306688|(15539,[3,176,220…|[[9.8549067E8],
[…|
|[hate, journey, c…|1579371151515856899|(15539,[0,147,559…|[[4.7945584E7],
[…|
|[pros, would, exp…|1579371242481946624|(15539,[26,1290,1…|[[3.68873365E8],
…|
|[fuck, university…|1579371351860973574|(15539,[3,185,294…|[[1.15057803E8],
```

```
…|
|[know, blessed, a…|1579371371238686720|(15539,[0,16,165,…|[[4.7945584E7],
[…|
|[high, school, gu…|1579371427224260608|(15539,[1,2,167,1…|[[2.59372826E8],
…|
|[engr, zohaib, fa…|1579371431846379520|(15539,[1,7,226,8…|[[2.71995886E8],
…|
|[destroyed, wings…|1579371459671392258|(15539,[0,255,846…|[[4.7945584E7],
[…|
|[terminal, babyfa…|1579371466013147138|(15539,[1,2,23,27…|[[3.331227E7],
[1…|
|[high, school, fr…|1579371539409309696|(15539,[1,2,122,2…|[[5.75251383E8],
…|
|[college, footbal…|1579371629536514049|(15539,[0,8,88,10…|[[4.7945584E7],
[…|
|[apparently, cent…|1579371630140469250|(15539,[1,14,21,2…|[[6.7121627E7],
[…|
|[people, expected…|1579371663678111744|(15539,[7,14,23,4…|[[2534020.0],
[2…|
|[revive, culture,…|1579371674574946304|(15539,[3,386,394…|[[8.0251182E7],
[…|
+--------------------+------------------+--------------------+-----------------
---+
only showing top 20 rows
```

[ ]:
```
jaccard_distance = 0.5
df_dups_text = text_model.approxSimilarityJoin(text_hashed, text_hashed,␣
 ↪jaccard_distance).\
    filter("datasetA.id < datasetB.id").select(col("distCol"),\
    col("datasetA.id").alias("id_A"),
    col("datasetB.id").alias("id_B"))

df_dups_text.show()
```

```
23/03/09 20:07:01 WARN org.apache.spark.scheduler.TaskSetManager: Lost task 0.0
in stage 32.0 (TID 7963) (hub-msca-chen26-bdp-chen26-w-0.c.chen26-bdp-class-
project.internal executor 21): org.apache.spark.SparkException: Failed to
execute user defined function(LSHModel$Lambda$3752/181969739:
(struct<type:tinyint,size:int,indices:array<int>,values:array<double>>) =>
array<struct<type:tinyint,size:int,indices:array<int>,values:array<double>>>)
        at org.apache.spark.sql.catalyst.expressions.GeneratedClass$GeneratedIte
ratorForCodegenStage5.processNext(Unknown Source)
        at org.apache.spark.sql.execution.BufferedRowIterator.hasNext(BufferedRo
wIterator.java:43)
        at org.apache.spark.sql.execution.WholeStageCodegenExec$$anon$1.hasNext(
WholeStageCodegenExec.scala:755)
        at scala.collection.Iterator$$anon$11.hasNext(Iterator.scala:491)
```

```
        at scala.collection.Iterator$ConcatIterator.hasNext(Iterator.scala:224)
        at scala.collection.Iterator$$anon$10.hasNext(Iterator.scala:460)
        at org.apache.spark.sql.catalyst.expressions.GeneratedClass$GeneratedIte
ratorForCodegenStage6.processNext(Unknown Source)
        at org.apache.spark.sql.execution.BufferedRowIterator.hasNext(BufferedRo
wIterator.java:43)
        at org.apache.spark.sql.execution.WholeStageCodegenExec$$anon$1.hasNext(
WholeStageCodegenExec.scala:755)
        at org.apache.spark.sql.execution.SparkPlan.$anonfun$getByteArrayRdd$1(S
parkPlan.scala:345)
        at
org.apache.spark.rdd.RDD.$anonfun$mapPartitionsInternal$2(RDD.scala:898)
        at
org.apache.spark.rdd.RDD.$anonfun$mapPartitionsInternal$2$adapted(RDD.scala:898)
        at
org.apache.spark.rdd.MapPartitionsRDD.compute(MapPartitionsRDD.scala:52)
        at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scala:373)
        at org.apache.spark.rdd.RDD.iterator(RDD.scala:337)
        at org.apache.spark.scheduler.ResultTask.runTask(ResultTask.scala:90)
        at org.apache.spark.scheduler.Task.run(Task.scala:131)
        at
org.apache.spark.executor.Executor$TaskRunner.$anonfun$run$3(Executor.scala:505)
        at org.apache.spark.util.Utils$.tryWithSafeFinally(Utils.scala:1439)
        at org.apache.spark.executor.Executor$TaskRunner.run(Executor.scala:508)
        at
java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1149)
        at
java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624)
        at java.lang.Thread.run(Thread.java:750)
Caused by: java.lang.IllegalArgumentException: requirement failed: Must have at
least 1 non zero entry.
        at scala.Predef$.require(Predef.scala:281)
        at
org.apache.spark.ml.feature.MinHashLSHModel.hashFunction(MinHashLSH.scala:61)
        at
org.apache.spark.ml.feature.LSHModel.$anonfun$transform$1(LSH.scala:101)
        … 23 more

23/03/09 20:07:15 WARN org.apache.spark.scheduler.TaskSetManager: Lost task 0.0
in stage 34.0 (TID 7971) (hub-msca-chen26-bdp-chen26-sw-lm67.c.chen26-bdp-class-
project.internal executor 19): org.apache.spark.SparkException: Failed to
execute user defined function(LSHModel$Lambda$3752/181969739:
(struct<type:tinyint,size:int,indices:array<int>,values:array<double>>) =>
array<struct<type:tinyint,size:int,indices:array<int>,values:array<double>>>)
        at org.apache.spark.sql.catalyst.expressions.GeneratedClass$GeneratedIte
ratorForCodegenStage7.processNext(Unknown Source)
        at org.apache.spark.sql.execution.BufferedRowIterator.hasNext(BufferedRo
wIterator.java:43)
```

```
        at org.apache.spark.sql.execution.WholeStageCodegenExec$$anon$1.hasNext(
WholeStageCodegenExec.scala:755)
        at scala.collection.Iterator$$anon$11.hasNext(Iterator.scala:491)
        at scala.collection.Iterator$ConcatIterator.hasNext(Iterator.scala:224)
        at scala.collection.Iterator$$anon$10.hasNext(Iterator.scala:460)
        at org.apache.spark.sql.catalyst.expressions.GeneratedClass$GeneratedIte
ratorForCodegenStage8.agg_doAggregateWithKeys_0$(Unknown Source)
        at org.apache.spark.sql.catalyst.expressions.GeneratedClass$GeneratedIte
ratorForCodegenStage8.processNext(Unknown Source)
        at org.apache.spark.sql.execution.BufferedRowIterator.hasNext(BufferedRo
wIterator.java:43)
        at org.apache.spark.sql.execution.WholeStageCodegenExec$$anon$1.hasNext(
WholeStageCodegenExec.scala:755)
        at org.apache.spark.sql.execution.SparkPlan.$anonfun$getByteArrayRdd$1(S
parkPlan.scala:345)
        at
org.apache.spark.rdd.RDD.$anonfun$mapPartitionsInternal$2(RDD.scala:898)
        at
org.apache.spark.rdd.RDD.$anonfun$mapPartitionsInternal$2$adapted(RDD.scala:898)
        at
org.apache.spark.rdd.MapPartitionsRDD.compute(MapPartitionsRDD.scala:52)
        at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scala:373)
        at org.apache.spark.rdd.RDD.iterator(RDD.scala:337)
        at org.apache.spark.scheduler.ResultTask.runTask(ResultTask.scala:90)
        at org.apache.spark.scheduler.Task.run(Task.scala:131)
        at
org.apache.spark.executor.Executor$TaskRunner.$anonfun$run$3(Executor.scala:505)
        at org.apache.spark.util.Utils$.tryWithSafeFinally(Utils.scala:1439)
        at org.apache.spark.executor.Executor$TaskRunner.run(Executor.scala:508)
        at
java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1149)
        at
java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624)
        at java.lang.Thread.run(Thread.java:750)
Caused by: java.lang.IllegalArgumentException: requirement failed: Must have at
least 1 non zero entry.
        at scala.Predef$.require(Predef.scala:281)
        at
org.apache.spark.ml.feature.MinHashLSHModel.hashFunction(MinHashLSH.scala:61)
        at
org.apache.spark.ml.feature.LSHModel.$anonfun$transform$1(LSH.scala:101)
        … 24 more

23/03/09 20:07:29 WARN org.apache.spark.scheduler.TaskSetManager: Lost task 0.4
in stage 34.0 (TID 7975) (hub-msca-chen26-bdp-chen26-w-0.c.chen26-bdp-class-
project.internal executor 21): org.apache.spark.SparkException: Failed to
execute user defined function(LSHModel$$Lambda$3752/181969739:
(struct<type:tinyint,size:int,indices:array<int>,values:array<double>>) =>
```

```
array<struct<type:tinyint,size:int,indices:array<int>,values:array<double>>>)
        at org.apache.spark.sql.catalyst.expressions.GeneratedClass$GeneratedIte
ratorForCodegenStage7.processNext(Unknown Source)
        at org.apache.spark.sql.execution.BufferedRowIterator.hasNext(BufferedRo
wIterator.java:43)
        at org.apache.spark.sql.execution.WholeStageCodegenExec$$anon$1.hasNext(
WholeStageCodegenExec.scala:755)
        at scala.collection.Iterator$$anon$11.hasNext(Iterator.scala:491)
        at scala.collection.Iterator$ConcatIterator.hasNext(Iterator.scala:224)
        at scala.collection.Iterator$$anon$10.hasNext(Iterator.scala:460)
        at org.apache.spark.sql.catalyst.expressions.GeneratedClass$GeneratedIte
ratorForCodegenStage8.agg_doAggregateWithKeys_0$(Unknown Source)
        at org.apache.spark.sql.catalyst.expressions.GeneratedClass$GeneratedIte
ratorForCodegenStage8.processNext(Unknown Source)
        at org.apache.spark.sql.execution.BufferedRowIterator.hasNext(BufferedRo
wIterator.java:43)
        at org.apache.spark.sql.execution.WholeStageCodegenExec$$anon$1.hasNext(
WholeStageCodegenExec.scala:755)
        at org.apache.spark.sql.execution.SparkPlan.$anonfun$getByteArrayRdd$1(S
parkPlan.scala:345)
        at
org.apache.spark.rdd.RDD.$anonfun$mapPartitionsInternal$2(RDD.scala:898)
        at
org.apache.spark.rdd.RDD.$anonfun$mapPartitionsInternal$2$adapted(RDD.scala:898)
        at
org.apache.spark.rdd.MapPartitionsRDD.compute(MapPartitionsRDD.scala:52)
        at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scala:373)
        at org.apache.spark.rdd.RDD.iterator(RDD.scala:337)
        at org.apache.spark.scheduler.ResultTask.runTask(ResultTask.scala:90)
        at org.apache.spark.scheduler.Task.run(Task.scala:131)
        at
org.apache.spark.executor.Executor$TaskRunner.$anonfun$run$3(Executor.scala:505)
        at org.apache.spark.util.Utils$.tryWithSafeFinally(Utils.scala:1439)
        at org.apache.spark.executor.Executor$TaskRunner.run(Executor.scala:508)
        at
java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1149)
        at
java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624)
        at java.lang.Thread.run(Thread.java:750)
Caused by: java.lang.IllegalArgumentException: requirement failed: Must have at
least 1 non zero entry.
        at scala.Predef$.require(Predef.scala:281)
        at
org.apache.spark.ml.feature.MinHashLSHModel.hashFunction(MinHashLSH.scala:61)
        at
org.apache.spark.ml.feature.LSHModel.$anonfun$transform$1(LSH.scala:101)
        … 24 more
```

```
+------------------+------------------+------------------+
|           distCol|               id_A|               id_B|
+------------------+------------------+------------------+
|               0.4|1517190870121996288|1622547699458351104|
|0.16666666666666663|1517190897548550144|1517191706734649347|
|0.16666666666666663|1517190897548550144|1517191145939517441|
|0.33333333333333337|1517190954113015813|1549424569630576641|
|0.33333333333333337|1517190954113015813|1564108139527487488|
|0.33333333333333337|1517190954113015813|1575539471067410432|
|0.33333333333333337|1517190954113015813|1568809101152370689|
|0.33333333333333337|1517190954113015813|1554421877942140929|
|0.33333333333333337|1517190954113015813|1594744950474035200|
|0.33333333333333337|1517190954113015813|1622498125817454592|
|0.16666666666666663|1517191145939517441|1517191706734649347|
|0.19999999999999996|1517191196724051971|1517191330623066112|
|               0.4|1517191317884973059|1566648298819158022|
| 0.1333333333333333|1517191434855714822|1517191450043289600|
|0.15384615384615385|1517191526673424384|1517191605215809537|
|0.15384615384615385|1517191544012374018|1517191669904408576|
|0.33333333333333337|1517191559338360837|1564109295364591617|
|0.33333333333333337|1517191559338360837|1602841075453829121|
|               0.4|1517191629748248577|1566649077957369856|
|0.33333333333333337|1517191740872171521|1549424569630576641|
+------------------+------------------+------------------+
only showing top 20 rows
```

[ ]:
```
# original news tweets
```

[12]:
```
news_text = t_group.filter(col('group') == 'News').
 ↪select('id','user_id','user_name','user_descrip', 'text','group','retweet')
ori_news = news_text.filter(col('retweet').isNull())
ori_news = ori_news.sample(False, 10000/len(ori_news.collect()), 1000)
```

[13]:
```
from pyspark.sql import functions as F
from pyspark.sql import types as t
from pyspark.sql.types import ArrayType, IntegerType

df_text_news = ori_news.select('text')
eng_ord=F.udf(lambda x: [ord(a) for a in x],t.ArrayType(IntegerType()))

def english_filter(x):
    for index in range(len(x)):
        if x[index] > 128:
```

```
            return False
        else:
            return True

filter_udf = F.udf(english_filter, BooleanType())
news_text_filter = df_text_news.filter(filter_udf(eng_ord('text')) == True)
```

[ ]:
```
news_text = news_text_filter.rdd.map(lambda x : x['text']).filter(lambda x: x␣
↪is not None)

StopWords = stopwords.words("english")

tokens = news_text\
    .map( lambda document: document.strip().lower())\
    .map( lambda document: re.split(" ", document))\
    .map( lambda word: [x for x in word if x.isalnum()])\
    .map( lambda word: [x for x in word if len(x) > 3] )\
    .map( lambda word: [x for x in word if x not in StopWords])\
    .zipWithIndex()

row = Row('text')
news_text_data=news_text.map(row).zipWithIndex().toDF(['text','id'])
```

[17]:
```
df_tokens_news = spark.createDataFrame(tokens, ["text_words",'id'])
df_tokens_news = df_tokens_news.filter(size('text_words') >= 1)
vectorize = CountVectorizer(inputCol="text_words", outputCol="features",␣
↪minDF=1.0)
text_vectorize_news = vectorize.fit(df_tokens_news).transform(df_tokens_news)
```

[28]:
```
from pyspark.ml.linalg import Vector
from pyspark.sql.functions import udf
from pyspark.sql.types import BooleanType

def is_not_empty_vector(vect: Vector) -> bool:
    if_not_empty = vect.numNonzeros > 0
    return if_not_empty

is_nonzero_vector_udf = udf(is_not_empty_vector, BooleanType())

text_vectorize_news = text_vectorize_news.withColumn("notEmpty",␣
↪is_nonzero_vector_udf("features"))
```

[29]:
```
mh = MinHashLSH(inputCol="features", outputCol="hashes", numHashTables=5)
model_text_news = mh.fit(text_vectorize_news)
text_hashed_news = mh.fit(text_vectorize_news).transform(text_vectorize_news)
```

[30]:
```
jaccard_distance = 0.5

df_dups_text_news_50 = model_text_news.approxSimilarityJoin(text_hashed_news,␣
 ↪text_hashed_news, jaccard_distance).\
filter("datasetA.id < datasetB.id").select(
            col("distCol"),
             col("datasetA.id").alias("id_A"),
             col("datasetB.id").alias("id_B"))
```

[ ]:
```

```