In [18]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
import re
import os
```

In [2]:

```
df = pd.read_csv('train.csv')
print("Number of data points:",df.shape[0])
```

```
Number of data points: 404290
```

In [3]:

```
df.head()
```

Out[3]:

	id	qid1	qid2	question1	question2	is_duplicate
0	0	1	2	What is the step by step guide to invest in sh...	What is the step by step guide to invest in sh...	0
1	1	3	4	What is the story of Kohinoor (Koh-i-Noor) Dia...	What would happen if the Indian government sto...	0
2	2	5	6	How can I increase the speed of my internet co...	How can Internet speed be increased by hacking...	0
3	3	7	8	Why am I mentally very lonely? How can I solve...	Find the remainder when [math]23^{24}[/math] i...	0
4	4	9	10	Which one dissolve in water quikly sugar, salt...	Which fish would survive in salt water?	0

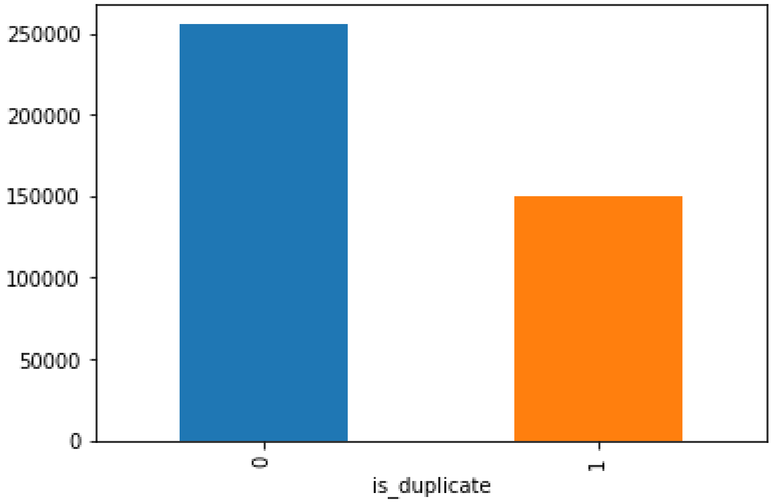
In [4]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 404290 entries, 0 to 404289
Data columns (total 6 columns):
id              404290 non-null int64
qid1            404290 non-null int64
qid2            404290 non-null int64
question1       404289 non-null object
question2       404288 non-null object
is_duplicate    404290 non-null int64
dtypes: int64(4), object(2)
memory usage: 18.5+ MB
```

In [5]:
```
df.groupby('is_duplicate')['id'].count().plot.bar()
```

Out[5]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x1cf3edfe6d8>
```



In [6]:
```
print("Total pairs for training:\n {}".format(len(df)))
```

```
Total pairs for training:
 404290
```

In [7]:
```
qids = pd.Series(df['qid1'].tolist() + df['qid2'].tolist())
```

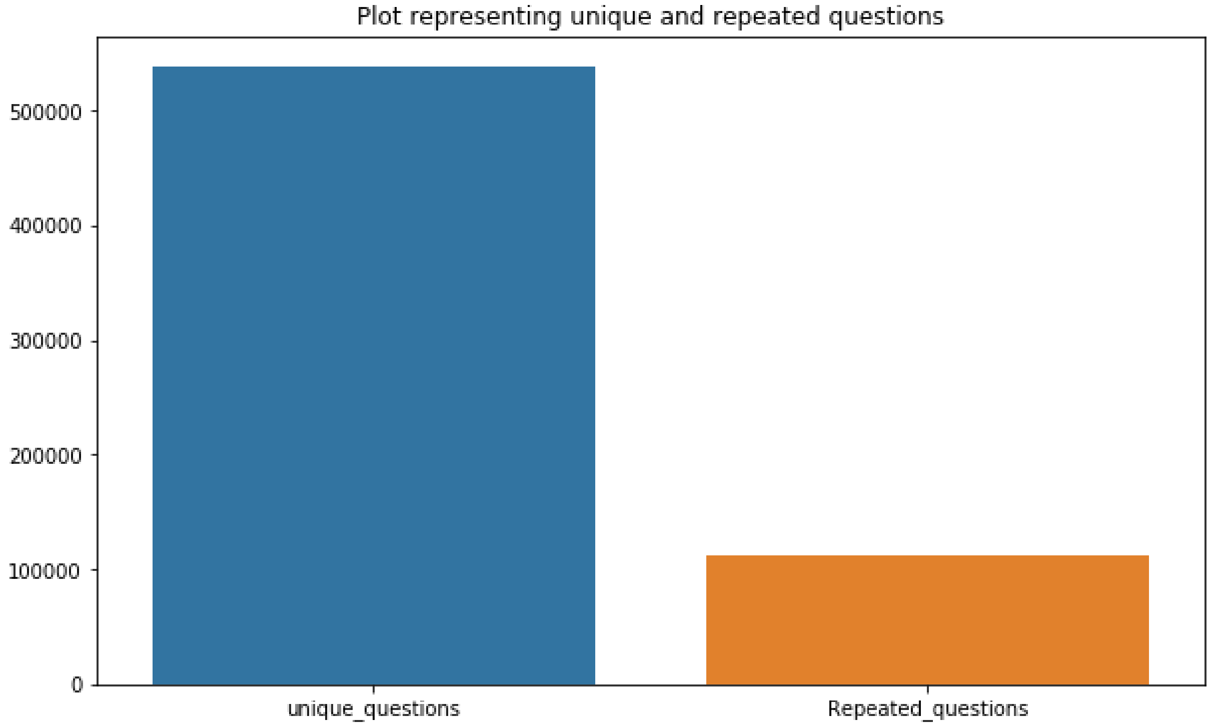
In [8]:
```
unique_questions = len(np.unique(qids))
qs_withmore_thanOneOcuurence = np.sum(qids.value_counts()>1)
```

In [9]:
```
print('Number of questions with more than one occurence:',unique_questions)
print('Max number of time a single question is repeated:{}\n'.format(max(qids.val
```

```
Number of questions with more than one occurence: 537933
Max number of time a single question is repeated:157
```

In [10]:

```
x = ['unique_questions','Repeated_questions']
y = [unique_questions,qs_withmore_thanOneOcuurence]
plt.figure(figsize=(10,6))
plt.title("Plot representing unique and repeated questions")
sns.barplot(x,y)
plt.show()
```



In [19]:

```
#checking for duplicate pairs
pair_duplicates = df[['qid1','qid2','is_duplicate']].groupby(['qid1','qid2']).cou
```

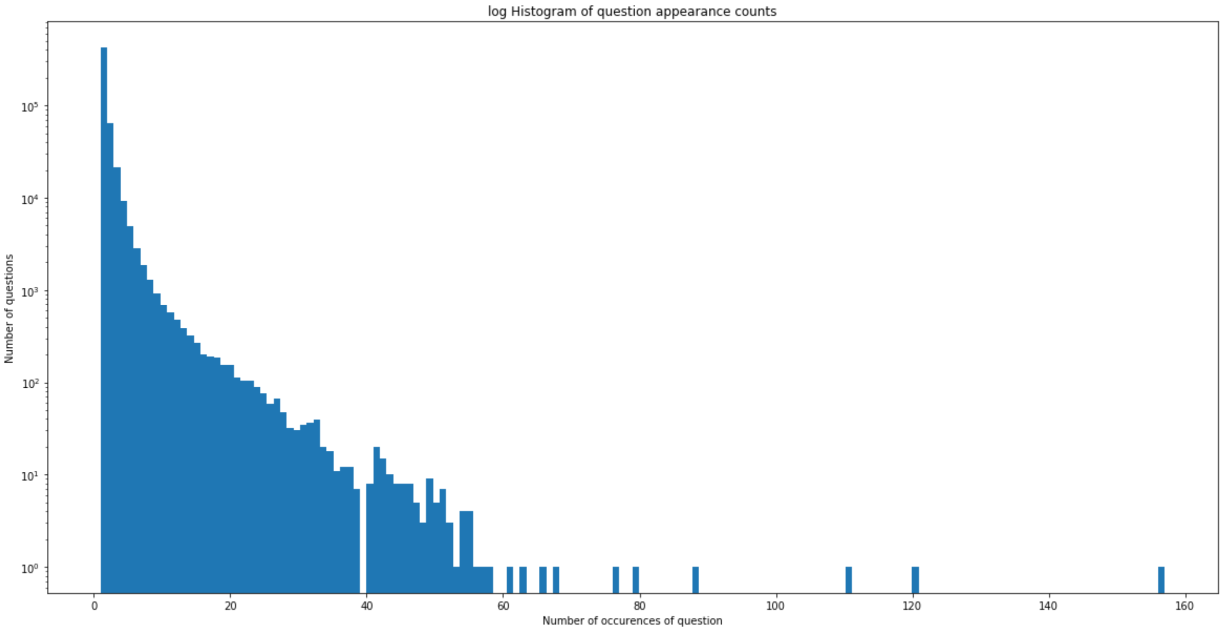
In [21]:

```
# this is to check if there are any duplicate questions
print(pair_duplicates.shape[0]-df.shape[0])
```

```
0
```

In [13]:
```
plt.figure(figsize=(20,10))
plt.hist(qids.value_counts(),bins=160)
plt.yscale('log',nonposy ='clip')
plt.title('log Histogram of question appearance counts')
plt.xlabel('Number of occurences of question')
plt.ylabel('Number of questions')
print('Maximum number of times a single question is repeated:',max(qids.value_cou
```

```
Maximum number of times a single question is repeated: 157
```



In [16]:
```
#cheking for rows having null values
nan_rows = df[df.isnull().any(1)]
print(nan_rows)
```

```
            id    qid1    qid2                         question1  \
105780  105780  174363  174364    How can I develop android app?
201841  201841  303951  174364  How can I create an Android app?
363362  363362  493340  493341                               NaN

                                               question2  is_duplicate
105780                                               NaN             0
201841                                               NaN             0
363362  My Chinese name is Haichao Yu. What English na...             0
```

In [17]:
```
df = df.fillna('')
nan_rows = df[df.isnull().any(1)]
print(nan_rows)
```

```
Empty DataFrame
Columns: [id, qid1, qid2, question1, question2, is_duplicate]
Index: []
```

Basic Feature extraction(before cleaning)

Constructing a few features like:

1)freq_id1 = frequency of qid1 2)freq_id2 = frequency of qid2 3)q1len = len of q1 4)q2len = len of q2 5)q1_n_words = Number of words in Question1 6)q2_n_words = Number of words in Question2 7)word_common = (Number of common unique words in Question1 and Question2) 8)word_total = (Total num of words in Question1 + Total Num of words in Question2) 9)word_share = (word_common)/(word_Total) 10)freq_q1+freq_q2 = sum total of frequency of qid1 and qid2 11)freq_q1-freq_q2 = absolute difference of frequency of qid1 and qid2

In [27]:
```
if os.path.isfile('df_fe_without_preprocessing_train.csv'):
    df = pd.read_csv('df_fe_without_preprocessing_train.csv',encoding='latin-1')
else:
    df['freq_qid1'] = df.groupby('qid1')['qid1'].transform('count')
    df['freq_qid2'] = df.groupby('qid2')['qid2'].transform('count')
    df['q1len']     = df['question1'].str.len()
    df['q2len']     = df['question2'].str.len()
    df['q1_n_words']= df['question1'].apply(lambda row: len(row.split(" ")))
    df['q2_n_words']= df['question2'].apply(lambda row: len(row.split(" ")))
```

In [29]:
```
def normalized_word_Common(row):
        w1 = set(map(lambda word:word.lower().strip(),row['question1'].split(" ")
        w2 = set(map(lambda word:word.lower().strip(),row['question2'].split(" ")
        return 1.0*len(w1&w2)

df['word_common'] = df.apply(normalized_word_Common,axis=1)
```

In [30]:
```
def normalized_word_total(row):
    w1 = set(map(lambda word:word.lower().strip(),row['question1'].split(" ")))
    w2 = set(map(lambda word:word.lower().strip(),row['question2'].split(" ")))
    return 1.0*(len(w1)+len(w2))
df['word_Total'] = df.apply(normalized_word_total,axis=1)
```

In [32]:
```
def normalized_word_share(row):
    w1 = set(map(lambda word:word.lower().strip(),row['question1'].split(" ")))
    w2 = set(map(lambda word:word.lower().strip(),row['question2'].split(" ")))
    return 1.0*len(w1&w2)/(len(w1)+len(w2))
df['Word_Share'] = df.apply(normalized_word_share,axis=1)
```

In [33]:
```
df['freq_q1+q2'] = df['freq_qid1'] + df['freq_qid2']
```

In [34]:
```
df['freq_q1-q2']  = abs(df['freq_qid1']-df['freq_qid2'])
```

In [35]:
```
df.to_csv('df_fe_without_preprocessing_train.csv',index=False)
```

In [36]:
```
df.head()
```

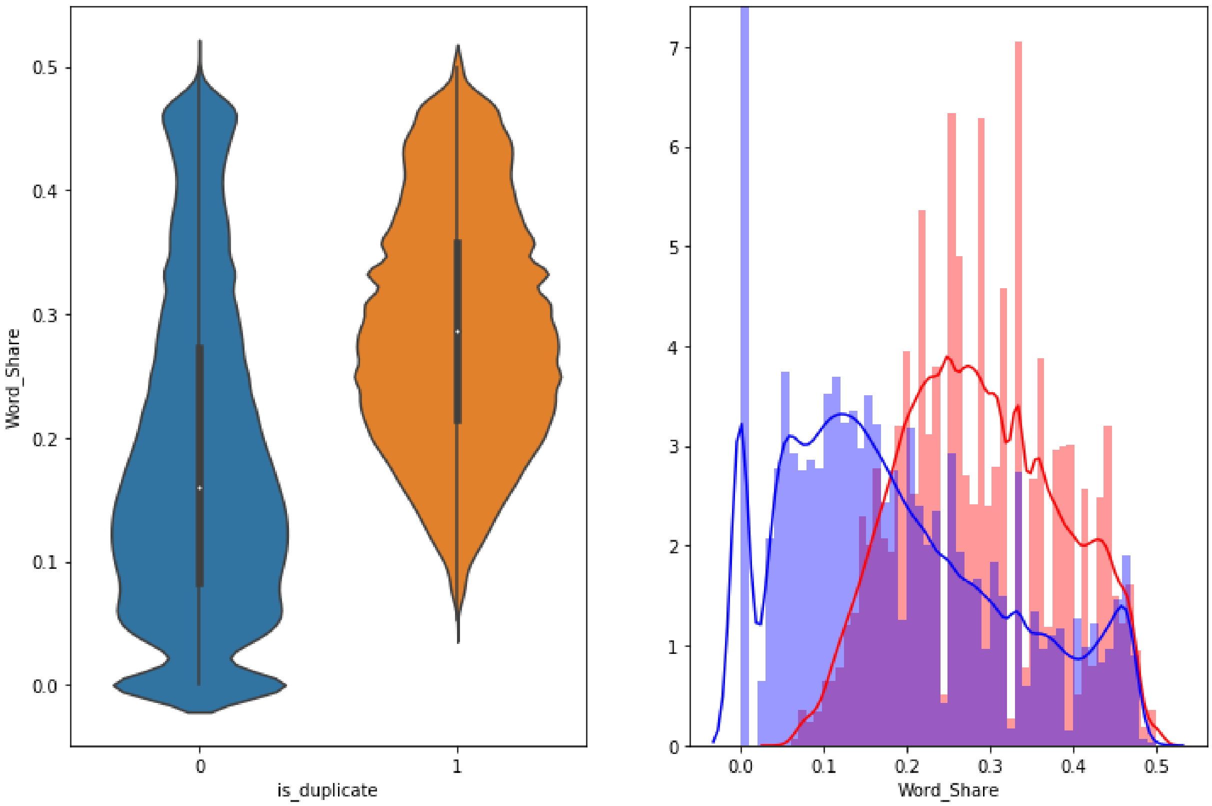
Out[36]:

	id	qid1	qid2	question1	question2	is_duplicate	freq_qid1	freq_qid2	q1len	q2len	q1_n_v
0	0	1	2	What is the step by step guide to invest in sh...	What is the step by step guide to invest in sh...	0	1	1	66	57	
1	1	3	4	What is the story of Kohinoor (Koh-i-Noor) Dia...	What would happen if the Indian government sto...	0	4	1	51	88	
2	2	5	6	How can I increase the speed of my internet co...	How can Internet speed be increased by hacking...	0	1	1	73	59	
3	3	7	8	Why am I mentally very lonely? How can I solve...	Find the remainder when [math]23^{24}[/math] i...	0	1	1	50	65	
4	4	9	10	Which one dissolve in water quikly sugar, salt...	Which fish would survive in salt water?	0	3	1	76	39	

In [38]:
```
plt.figure(figsize=(12,8))
plt.subplot(1,2,1)
sns.violinplot(x='is_duplicate',y='Word_Share',data=df[0:])

plt.subplot(1,2,2)
sns.distplot(df[df['is_duplicate']==1.0]['Word_Share'][0:],label = '1',color = 'r
sns.distplot(df[df['is_duplicate']==0.0]['Word_Share'][0:],label = '0',color = 'b
plt.show()
```

```
D:\Anaconda\envs\tensorflow\lib\site-packages\matplotlib\axes\_axes.py:6462: Us
erWarning: The 'normed' kwarg is deprecated, and has been replaced by the 'dens
ity' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
```

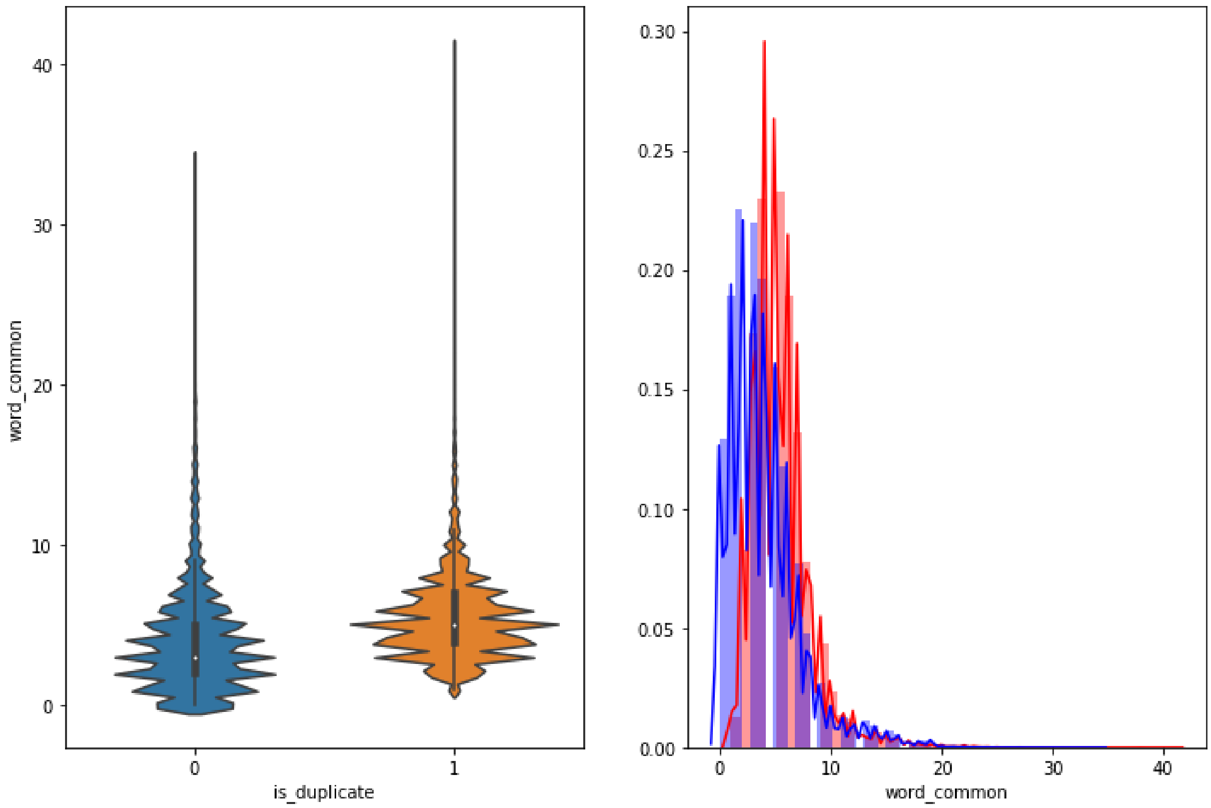


In [39]:

```
plt.figure(figsize=(12,8))
plt.subplot(1,2,1)
sns.violinplot(x='is_duplicate',y='word_common',data=df[0:])

plt.subplot(1,2,2)
sns.distplot(df[df['is_duplicate']==1.0]['word_common'][0:],label = '1',color = '
sns.distplot(df[df['is_duplicate']==0.0]['word_common'][0:],label = '0',color = '
plt.show()
```

```
D:\Anaconda\envs\tensorflow\lib\site-packages\matplotlib\axes\_axes.py:6462: Us
erWarning: The 'normed' kwarg is deprecated, and has been replaced by the 'dens
ity' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
```



In [ ]: