Binary_class_Exploratory_data_analysis

May 14, 2019

```
In [1]: !pip install vaderSentiment

Requirement already satisfied: vaderSentiment in /usr/local/lib/python3.6/dist-packages (3.2.1)
```

```
In [0]: import pandas as pd
```

```
In [0]: from sklearn.feature_extraction.text import CountVectorizer
        from sklearn.model_selection import train_test_split
        from sklearn.pipeline import make_pipeline
        from sklearn.linear_model import LogisticRegression
        from sklearn.model_selection import cross_val_score
        from sklearn.model_selection import GridSearchCV
        from sklearn.feature_extraction.text import TfidfVectorizer, TfidfTransformer
        import nltk
        import gensim
        from nltk.tokenize import sent_tokenize, word_tokenize
        from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer
        from scipy.sparse import hstack
        from sklearn.metrics import average_precision_score
        from sklearn.metrics import roc_auc_score
        from nltk import word_tokenize,sent_tokenize
        from gensim import corpora
        from sklearn.pipeline import Pipeline
        from sklearn.model_selection import cross_val_predict
        import matplotlib.pyplot as plt
```

```
In [0]: df_train = pd.read_csv("/content/labeled_data.csv")[["tweet","class"]]
        df_train.loc[df_train['class'] == 2, 'class'] = 1
        df_train=df_train.sample(frac=1)
```

```
In [0]: X_train = df_train[df_train['class']==0]
        X__train = df_train[df_train['class']==1][:len(X_train["class"])]
        df_train = X_train.append(X__train)
```

```
In [45]: nltk.download("stopwords")
         from nltk.stem.porter import *
```

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

```
In [0]: stopwords=stopwords = nltk.corpus.stopwords.words("english")

        other_exclusions = ["#ff", "ff", "rt","RT"]
        stopwords.extend(other_exclusions)

        stemmer = PorterStemmer()

        def preprocess(text_string):

            #Lowercase string
            text_string=text_string.lower()
            space_pattern = '\s+'
            giant_url_regex = ('http[s]?://(?:[a-zA-Z]|[0-9]|[$-_@.&+]|'
                '[!*\(\),]|(?:%[0-9a-fA-F][0-9a-fA-F]))+')
            mention_regex = '@[\w\-]+'
            hashtag_regex = '#[\w\-]+'
            parsed_text = re.sub(space_pattern, ' ', text_string)
            parsed_text = re.sub(giant_url_regex, 'URLHERE', parsed_text)
            parsed_text = re.sub(mention_regex, 'MENTIONHERE', parsed_text)
            parsed_text = re.sub(hashtag_regex, 'HASHTAGHERE', parsed_text)

            #Stem it
            tweet = " ".join(re.split("[^a-zA-Z]*", parsed_text)).strip()
            tokens = [stemmer.stem(t) for t in tweet.split()]
            return tokens


        def pos_tag_seq(tokens):
            tags = nltk.pos_tag(tokens)
            tag_list = [x[1] for x in tags]
            tag_str = " ".join(tag_list)
            return tag_str
```

```
In [0]: def join_sent(l):
          return " ".join(l)
```

```
In [48]: df_train.head()
```

```
Out[48]:                                                  tweet  class     counts  \
         16489  RT @Mitchellharri: Dont be a faggot, cover you...      0  (0, 1, 0)
         24091  ion speak on how flaw niggas and thirsty hoes ...      0  (0, 0, 2)
         15093  RT @EarlyLegend: I don't fuck with these hoes ...      0  (0, 1, 0)
         21578  The leftist/homosexual war on the #Catholic ch...      0  (0, 0, 1)
```

```
10083  I bet you think you a no-chill savage huh. Fuc...      0  (0, 1, 0)
```

	len	word_count	pos	neg	neu
16489	128	24	0.334	0.000	0.666
24091	100	18	0.000	0.091	0.909
15093	101	18	0.373	0.000	0.627
21578	134	20	0.000	0.170	0.830
10083	68	12	0.000	0.490	0.510

```
In [49]: s_train=df_train['tweet'].apply(preprocess)
```

```
/usr/lib/python3.6/re.py:212: FutureWarning: split() requires a non-empty pattern match.
  return _compile(pattern, flags).split(string, maxsplit)
```

```
In [0]: s_tr=s_train.apply(join_sent)
```

```
In [0]: vectorizer = TfidfVectorizer(
            preprocessor=None,
            lowercase=False,
            ngram_range=(1, 3),
            use_idf=True,
            smooth_idf=False,
            norm=None,
            stop_words=stopwords,
            decode_error='replace',
            max_features=10000,
            min_df=5,
            max_df=0.75)
```

```
In [0]: tfidf_tr = vectorizer.fit_transform(s_tr).toarray()
```

```
        vocab = {v:i for i, v in enumerate(vectorizer.get_feature_names())}
        idf_vals = vectorizer.idf_
        idf_dict = {i:idf_vals[i] for i in vocab.values()}
```

```
In [0]: from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer as VS
        sentiment_analyzer = VS()
```

```
In [0]: def get_sentiment(text):
          sentiment = sentiment_analyzer.polarity_scores(text)
          return sentiment

        #   return sentiment["neg"], sentiment["pos"], sentiment["neu"]
```

```
In [0]: df_train["sent"]=df_train["tweet"].apply(get_sentiment)
```

```
In [0]: foo_tr = lambda x: pd.Series([x["pos"],x["neg"],x["neu"]])
        rev_tr = df_train['sent'].apply(foo_tr)
```

```
In [0]: rev_tr.columns=["pos","neg","neu"]

In [58]: rev_tr.head()

Out[58]:          pos    neg    neu
         16489  0.334  0.000  0.666
         24091  0.000  0.091  0.909
         15093  0.373  0.000  0.627
         21578  0.000  0.170  0.830
         10083  0.000  0.490  0.510

In [0]: def return_cont(parsed_text):
          return(parsed_text.count('urlher'),parsed_text.count('mentionher'),parsed_text.count

In [0]: df_train["counts"]=s_tr.apply(return_cont)

In [0]: foo = lambda x: pd.Series([x[0],x[1],x[2]])
        mention_counts_tr = df_train['counts'].apply(foo)

In [62]: !pip install textstat
         from textstat.textstat import *

Requirement already satisfied: textstat in /usr/local/lib/python3.6/dist-packages (0.5.6)
Requirement already satisfied: repoze.lru in /usr/local/lib/python3.6/dist-packages (from texts
Requirement already satisfied: pyphen in /usr/local/lib/python3.6/dist-packages (from textstat)


In [0]: def get_other_features(text):
            space_pattern = '\s+'
            giant_url_regex = ('http[s]?://(?:[a-zA-Z]|[0-9]|[$-_@.&+]|'
                '[!*\(\),]|(?:%[0-9a-fA-F][0-9a-fA-F]))+')
            mention_regex = '@[\w\-]+'
            parsed_text = re.sub(space_pattern, ' ', text)
            parsed_text = re.sub(giant_url_regex, '', parsed_text)
            words = re.sub(mention_regex, '', parsed_text)

            syllables = textstat.syllable_count(words)
            num_chars = sum(len(w) for w in words)
            num_chars_total = len(text)
            num_terms = len(text.split())
            num_words = len(words.split())
            avg_syl = round(float((syllables+0.001))/float(num_words+0.001),4)
            num_unique_terms = len(set(words.split()))

            ###Modified FK grade, where avg words per sentence is just num words/1
            FKRA = round(float(0.39 * float(num_words)/1.0) + float(11.8 * avg_syl) - 15.59,1)
            ##Modified FRE score, where sentence fixed to 1
            FRE = round(206.835 - 1.015*(float(num_words)/1.0) - (84.6*float(avg_syl)),2)
```

```
            features = [FKRA, FRE,syllables, avg_syl, num_chars, num_terms, num_words,
                        num_unique_terms]
            return features
```

```
In [0]: other_feats_tr=df_train["tweet"].apply(get_other_features)
```

```
In [0]: other_features_names = ["FKRA", "FRE","num_syllables", "avg_syl_per_word", "num_chars"
```

```
In [0]: foo = lambda x: pd.Series(elem for elem in x)
        of_counts_tr = other_feats_tr.apply(foo)
```

```
In [0]: of_counts_tr.columns=other_features_names
```

```
In [68]: #Removing unnecessary columns
         df_train.drop([ "sent","counts"], axis=1)
```

Out[68]:

	tweet	class	len	\
16489	RT @Mitchellharri: Dont be a faggot, cover you...	0	128	
24091	ion speak on how flaw niggas and thirsty hoes ...	0	100	
15093	RT @EarlyLegend: I don't fuck with these hoes ...	0	101	
21578	The leftist/homosexual war on the #Catholic ch...	0	134	
10083	I bet you think you a no-chill savage huh. Fuc...	0	68	
3869	@L1LTR4P fucking losers wetbacks #SorryNotSorry	0	47	
7674	Almost got to see a white boy get beat around ...	0	77	
3310	@GrizzboAdams @wyattnuckels haha ight nig calm...	0	53	
8041	Bitch ass nigga, be hating on black women... U...	0	66	
7548	Al noooo ... Too late fuckin faggot ! Lmao	0	45	
22088	This kid looks like a retard when he tries hid...	0	73	
9329	Fucking gook	0	12	
19630	RT @markiiejay: You got niggas & i got bit...	0	50	
22798	White bus drivers are all white trash. #LosAng...	0	50	
24191	looking like a hillbilly and not matching is w...	0	99	
4396	@Pecan_B19 @ObeyMy_Realness dryer then them ba...	0	68	
18062	RT @WhitesOnly_1: #niggers! http://t.co/Hb3uJa...	0	50	
2832	@CheCha__ be gone my nigguh	0	27	
4572	@Roscoedash lmao, soft ass nigga trying to act...	0	122	
7040	@sizzurp__ @bootyacid @ILIKECATS74 @yoPapi_chu...	0	64	
4930	@TheKushZombie Aww y u so mad tho, a successfu...	0	81	
16595	RT @NYTMinusContext: kill whitey	0	32	
23708	cant trust niggas when it come to bitches	0	41	
12587	Lol RT @JayFucknHarris: Youuuuu got niggas but...	0	66	
2969	@Darienbaby_xoxo jk I'd fuck a dog before I fu...	0	63	
10631	I know righttt RT @dta_87: I hate fat loud bit...	0	50	
13722	Oregon losing IN OREGON ahhhahahaha faggots	0	43	
4714	@SlimGirl_Probz das calld usen yo brain retard...	0	90	
10601	I just wanna slit this mf throat	0	32	
4221	@NOT_UR_BRO @niccol3_xo lmao nigga shoulda sai...	0	111	

```
...                                                    ...    ...  ...
22030  This dumb Berk drives in circles through town ...      1  132
2554   @BarbieNixon a moment of silence for the bitch...      1   82
12638      Look at that THOT... pussy popping for usher.      1   45
1330   &#8220;@JalenRose: "I'm rich..." (Dave Chap...     1  108
3939                 @LittleNamms stick to the court hoe      1   35
21193        Stupid and ain't got hoes #ShmediumProblems      1   43
24212  mickeyblowsyourmind: who wants to be a skype g...      1  131
7649     All theses hoes on me , they so phony &#128534;      1   47
13943  Pussy nicca we ain't friends you ain't gotta @...      1   74
6746             @mza361 I love you and bitches love me.      1   39
5942   @erykadamitio_ @RiRi_Candee why nobody never i...      1   87
20876   Slack jawed yokel husband http://t.co/VE1PWFrz9t      1   48
7394                                  @zimm16 pussy!!!!!      1   18
18270  RT @_Creamm_: @Victoria_Finae dont mind my ret...      1   64
6450                             @kieffer_jason bye bitch      1   24
6212                           @jaimescudi_ fucking bitch      1   26
24450  some prude bitch unfollwed me show me ur ugly ...      1   50
13723    Oreo Ice Cream Sandwich http://t.co/RM0KsY99Bc"      1   47
21408  That nig was a G on that E60 I would have done...      1   61
22256  Translation for the slow he out saving hoes 1 ...      1  127
743    #Natitude? Okay. I'm only going to say it once...      1   78
13079  My mom told me never use the word cunt... ITS ...      1  140
20296  RT @vodkapapixo: "@Weed_Cloudz: "6 God" by Dra...      1   68
18506  RT @_xchaazelle: most of the bitches he fw not...      1  119
9414   Get your weak ass juco highlights off twitter hoe      1   49
14111  RT @AdamWeinstein: Working theory: This "selec...      1  140
258    "@The_Realist_Sam: @Madrid2_ yea, I'm on my iP...      1  129
6790   @obamac0re I don't like it peaceful that's bor...      1  139
15251  RT @FriendlyAssh0le: It's annoying as hell whe...      1  146
3067                                   @Dre_Day200 bitch      1   17

       word_count    pos    neg    neu
16489          24  0.334  0.000  0.666
24091          18  0.000  0.091  0.909
15093          18  0.373  0.000  0.627
21578          20  0.000  0.170  0.830
10083          12  0.000  0.490  0.510
3869            5  0.000  0.480  0.520
7674           17  0.152  0.242  0.606
3310            7  0.515  0.000  0.485
8041           12  0.000  0.643  0.357
7548            9  0.267  0.322  0.411
22088          14  0.138  0.309  0.552
9329            2  0.000  0.000  1.000
19630           9  0.000  0.512  0.488
22798           8  0.000  0.000  1.000
24191          15  0.333  0.169  0.498
```

4396	11	0.311	0.193	0.497
18062	4	0.000	0.000	1.000
2832	5	0.000	0.000	1.000
4572	22	0.133	0.289	0.578
7040	7	0.346	0.339	0.315
4930	15	0.417	0.320	0.263
16595	4	0.000	0.610	0.390
23708	8	0.149	0.484	0.367
12587	11	0.413	0.115	0.472
2969	12	0.119	0.440	0.440
10631	10	0.000	0.559	0.441
13722	6	0.000	0.630	0.370
4714	18	0.000	0.347	0.653
10601	7	0.000	0.000	1.000
4221	18	0.245	0.161	0.594
...	...	...	...	...
22030	18	0.131	0.149	0.721
2554	14	0.000	0.241	0.759
12638	8	0.000	0.000	1.000
1330	14	0.000	0.226	0.774
3939	6	0.000	0.000	1.000
21193	6	0.000	0.405	0.595
24212	20	0.000	0.162	0.838
7649	10	0.000	0.000	1.000
13943	13	0.000	0.206	0.794
6746	8	0.515	0.239	0.245
5942	13	0.128	0.492	0.380
20876	5	0.000	0.000	1.000
7394	2	0.000	0.000	1.000
18270	8	0.300	0.000	0.700
6450	3	0.000	0.655	0.345
6212	3	0.000	0.672	0.328
24450	10	0.000	0.470	0.530
13723	5	0.000	0.000	1.000
21408	15	0.000	0.000	1.000
22256	18	0.000	0.000	1.000
743	14	0.128	0.000	0.872
13079	28	0.058	0.756	0.186
20296	11	0.338	0.000	0.662
18506	21	0.000	0.180	0.820
9414	9	0.000	0.478	0.522
14111	23	0.050	0.139	0.811
258	20	0.000	0.226	0.774
6790	27	0.000	0.407	0.593
15251	24	0.170	0.313	0.517
3067	2	0.000	0.792	0.208

[2430 rows x 7 columns]

```
In [0]: import numpy as np
        x_train=np.concatenate([pd.DataFrame(tfidf_tr),rev_tr,mention_counts_tr, of_counts_tr]

In [70]: x_train[:10]

Out[70]: array([[ 0.,  0.,  0., ..., 24., 24., 23.],
                [ 0.,  0.,  0., ..., 18., 18., 17.],
                [ 0.,  0.,  0., ..., 18., 18., 12.],
                ...,
                [ 0.,  0.,  0., ...,  7.,  5.,  5.],
                [ 0.,  0.,  0., ..., 12., 12., 12.],
                [ 0.,  0.,  0., ...,  9.,  9.,  9.]])

In [0]: df_train['len'] = df_train['tweet'].astype(str).apply(len)
        df_train['word_count'] = df_train['tweet'].apply(lambda x: len(str(x).split()))

In [0]: df_train["pos"]=rev_tr["pos"]

In [0]: df_train["neg"]=rev_tr["neg"]
        df_train["neu"]=rev_tr["neu"]

In [74]: df_train.head()

Out[74]:                                                   tweet  class     counts  \
         16489  RT @Mitchellharri: Dont be a faggot, cover you...      0  (0, 1, 0)
         24091  ion speak on how flaw niggas and thirsty hoes ...      0  (0, 0, 2)
         15093  RT @EarlyLegend: I don't fuck with these hoes ...      0  (0, 1, 0)
         21578  The leftist/homosexual war on the #Catholic ch...      0  (0, 0, 1)
         10083  I bet you think you a no-chill savage huh. Fuc...      0  (0, 1, 0)

                len  word_count    pos    neg    neu  \
         16489  128          24  0.334  0.000  0.666
         24091  100          18  0.000  0.091  0.909
         15093  101          18  0.373  0.000  0.627
         21578  134          20  0.000  0.170  0.830
         10083   68          12  0.000  0.490  0.510

                                                     sent
         16489  {'neg': 0.0, 'neu': 0.666, 'pos': 0.334, 'comp...
         24091  {'neg': 0.091, 'neu': 0.909, 'pos': 0.0, 'comp...
         15093  {'neg': 0.0, 'neu': 0.627, 'pos': 0.373, 'comp...
         21578  {'neg': 0.17, 'neu': 0.83, 'pos': 0.0, 'compou...
         10083  {'neg': 0.49, 'neu': 0.51, 'pos': 0.0, 'compou...

In [0]: df_train=df_train.drop(["sent"], axis=1)

In [76]: df_train.drop(["counts"], axis=1)
         # pd.concat([df_train,pd.DataFrame(x_train)], axis=1)
```

Out[76]:

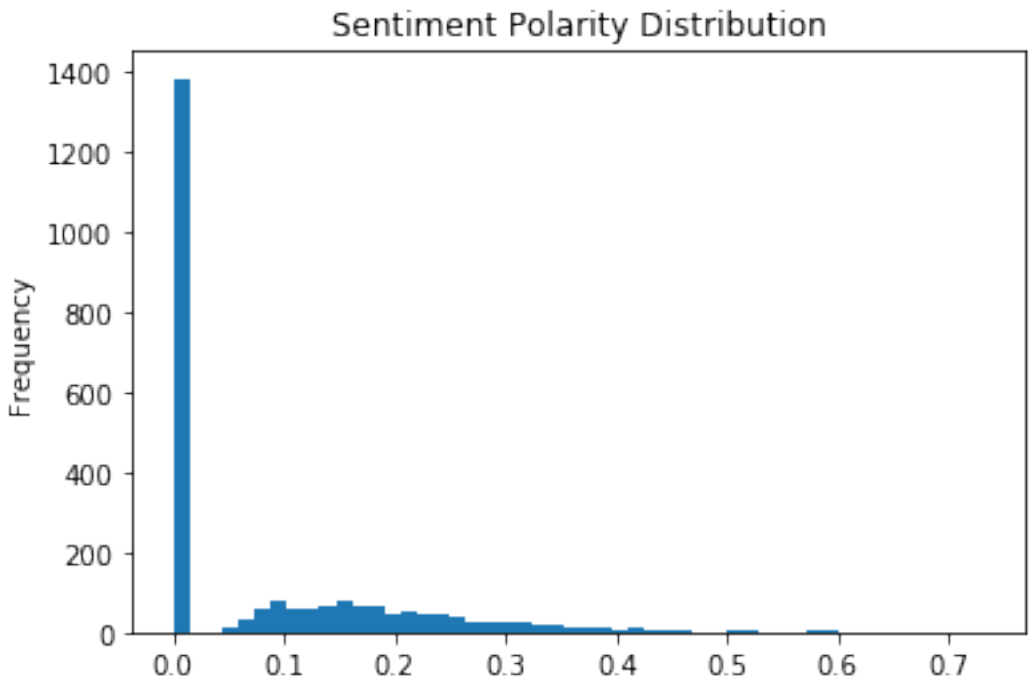
	tweet	class	len
16489	RT @Mitchellharri: Dont be a faggot, cover you...	0	128
24091	ion speak on how flaw niggas and thirsty hoes ...	0	100
15093	RT @EarlyLegend: I don't fuck with these hoes ...	0	101
21578	The leftist/homosexual war on the #Catholic ch...	0	134
10083	I bet you think you a no-chill savage huh. Fuc...	0	68
3869	@L1LTR4P fucking losers wetbacks #SorryNotSorry	0	47
7674	Almost got to see a white boy get beat around ...	0	77
3310	@GrizzboAdams @wyattnuckels haha ight nig calm...	0	53
8041	Bitch ass nigga, be hating on black women... U...	0	66
7548	Al noooo ... Too late fuckin faggot ! Lmao	0	45
22088	This kid looks like a retard when he tries hid...	0	73
9329	Fucking gook	0	12
19630	RT @markiiejay: You got niggas & i got bit...	0	50
22798	White bus drivers are all white trash. #LosAng...	0	50
24191	looking like a hillbilly and not matching is w...	0	99
4396	@Pecan_B19 @ObeyMy_Realness dryer then them ba...	0	68
18062	RT @WhitesOnly_1: #niggers! http://t.co/Hb3uJa...	0	50
2832	@CheCha__ be gone my nigguh	0	27
4572	@Roscoedash lmao, soft ass nigga trying to act...	0	122
7040	@sizzurp__ @bootyacid @ILIKECATS74 @yoPapi_chu...	0	64
4930	@TheKushZombie Aww y u so mad tho, a successfu...	0	81
16595	RT @NYTMinusContext: kill whitey	0	32
23708	cant trust niggas when it come to bitches	0	41
12587	Lol RT @JayFucknHarris: Youuuuu got niggas but...	0	66
2969	@Darienbaby_xoxo jk I'd fuck a dog before I fu...	0	63
10631	I know righttt RT @dta_87: I hate fat loud bit...	0	50
13722	Oregon losing IN OREGON ahhhahahaha faggots	0	43
4714	@SlimGirl_Probz das calld usen yo brain retard...	0	90
10601	I just wanna slit this mf throat	0	32
4221	@NOT_UR_BRO @niccol3_xo lmao nigga shoulda sai...	0	111
...	...	...	...
22030	This dumb Berk drives in circles through town ...	1	132
2554	@BarbieNixon a moment of silence for the bitch...	1	82
12638	Look at that THOT... pussy popping for usher.	1	45
1330	“@JalenRose: "I'm rich..." (Dave Chap...	1	108
3939	@LittleNamms stick to the court hoe	1	35
21193	Stupid and ain't got hoes #ShmediumProblems	1	43
24212	mickeyblowsyourmind: who wants to be a skype g...	1	131
7649	All theses hoes on me , they so phony 😖	1	47
13943	Pussy nicca we ain't friends you ain't gotta @...	1	74
6746	@mza361 I love you and bitches love me.	1	39
5942	@erykadamitio_ @RiRi_Candee why nobody never i...	1	87
20876	Slack jawed yokel husband http://t.co/VE1PWFrz9t	1	48
7394	@zimm16 pussy!!!!!	1	18
18270	RT @_Creamm_: @Victoria_Finae dont mind my ret...	1	64
6450	@kieffer_jason bye bitch	1	24
6212	@jaimescudi_ fucking bitch	1	26

```
24450  some prude bitch unfollwed me show me ur ugly ...      1   50
13723    Oreo Ice Cream Sandwich http://t.co/RM0KsY99Bc"      1   47
21408  That nig was a G on that E60 I would have done...      1   61
22256  Translation for the slow he out saving hoes 1 ...      1  127
743    #Natitude? Okay. I'm only going to say it once...      1   78
13079  My mom told me never use the word cunt... ITS ...      1  140
20296  RT @vodkapapixo: "@Weed_Cloudz: "6 God" by Dra...      1   68
18506  RT @_xchaazelle: most of the bitches he fw not...      1  119
9414   Get your weak ass juco highlights off twitter hoe      1   49
14111  RT @AdamWeinstein: Working theory: This "selec...      1  140
258    "@The_Realist_Sam: @Madrid2_ yea, I'm on my iP...      1  129
6790   @obamac0re I don't like it peaceful that's bor...      1  139
15251  RT @FriendlyAssh0le: It's annoying as hell whe...      1  146
3067                                    @Dre_Day200 bitch      1   17

       word_count    pos    neg    neu
16489          24  0.334  0.000  0.666
24091          18  0.000  0.091  0.909
15093          18  0.373  0.000  0.627
21578          20  0.000  0.170  0.830
10083          12  0.000  0.490  0.510
3869            5  0.000  0.480  0.520
7674           17  0.152  0.242  0.606
3310            7  0.515  0.000  0.485
8041           12  0.000  0.643  0.357
7548            9  0.267  0.322  0.411
22088          14  0.138  0.309  0.552
9329            2  0.000  0.000  1.000
19630           9  0.000  0.512  0.488
22798           8  0.000  0.000  1.000
24191          15  0.333  0.169  0.498
4396           11  0.311  0.193  0.497
18062           4  0.000  0.000  1.000
2832            5  0.000  0.000  1.000
4572           22  0.133  0.289  0.578
7040            7  0.346  0.339  0.315
4930           15  0.417  0.320  0.263
16595           4  0.000  0.610  0.390
23708           8  0.149  0.484  0.367
12587          11  0.413  0.115  0.472
2969           12  0.119  0.440  0.440
10631          10  0.000  0.559  0.441
13722           6  0.000  0.630  0.370
4714           18  0.000  0.347  0.653
10601           7  0.000  0.000  1.000
4221           18  0.245  0.161  0.594
...           ...    ...    ...    ...
22030          18  0.131  0.149  0.721
```

```
2554           14  0.000  0.241  0.759
12638           8  0.000  0.000  1.000
1330           14  0.000  0.226  0.774
3939            6  0.000  0.000  1.000
21193           6  0.000  0.405  0.595
24212          20  0.000  0.162  0.838
7649           10  0.000  0.000  1.000
13943          13  0.000  0.206  0.794
6746            8  0.515  0.239  0.245
5942           13  0.128  0.492  0.380
20876           5  0.000  0.000  1.000
7394            2  0.000  0.000  1.000
18270           8  0.300  0.000  0.700
6450            3  0.000  0.655  0.345
6212            3  0.000  0.672  0.328
24450          10  0.000  0.470  0.530
13723           5  0.000  0.000  1.000
21408          15  0.000  0.000  1.000
22256          18  0.000  0.000  1.000
743            14  0.128  0.000  0.872
13079          28  0.058  0.756  0.186
20296          11  0.338  0.000  0.662
18506          21  0.000  0.180  0.820
9414            9  0.000  0.478  0.522
14111          23  0.050  0.139  0.811
258            20  0.000  0.226  0.774
6790           27  0.000  0.407  0.593
15251          24  0.170  0.313  0.517
3067            2  0.000  0.792  0.208

[2430 rows x 7 columns]
```

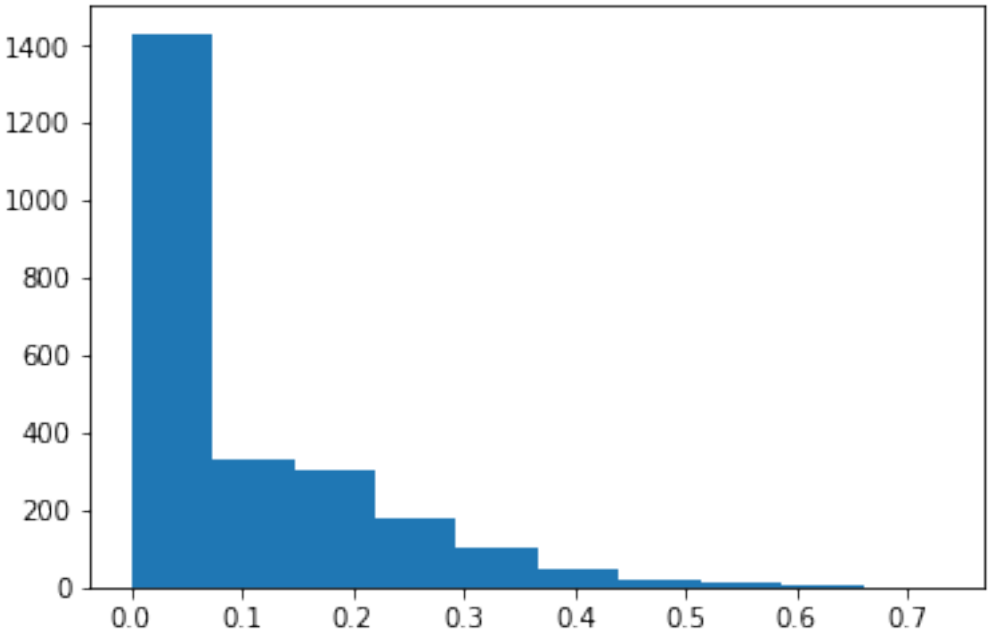
In [77]:
```
df_train['pos'].plot(
    kind='hist',
    bins=50,
#     xTitle='polarity',
#     linecolor='black',
#     yTitle='count',
    title='Sentiment Polarity Distribution')
```

Out[77]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f3dd06f31d0>
```

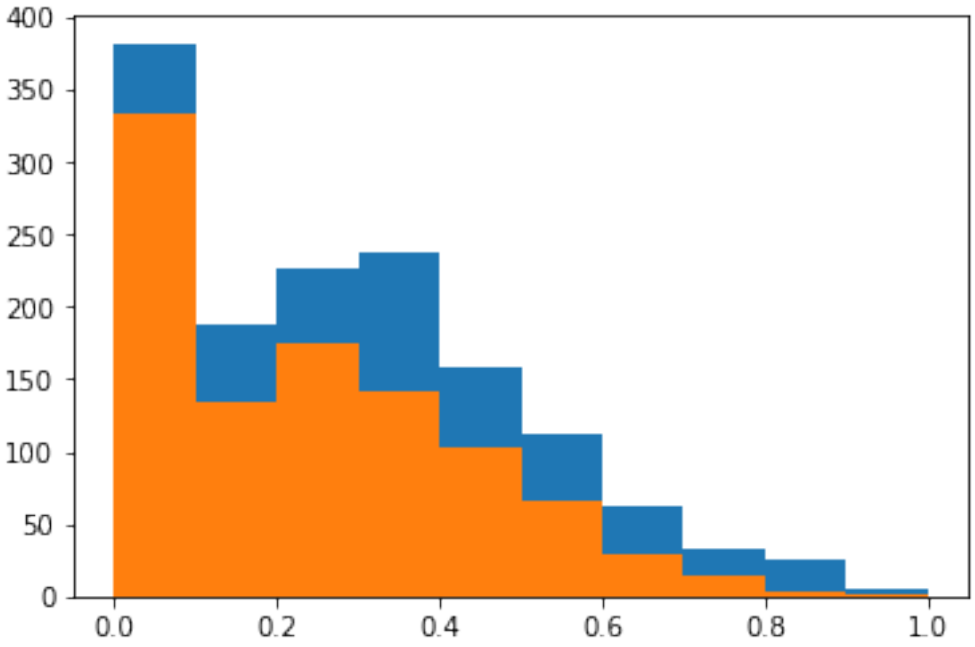



```
In [78]: import matplotlib.pyplot as plt
         plt.hist(df_train["pos"])
```

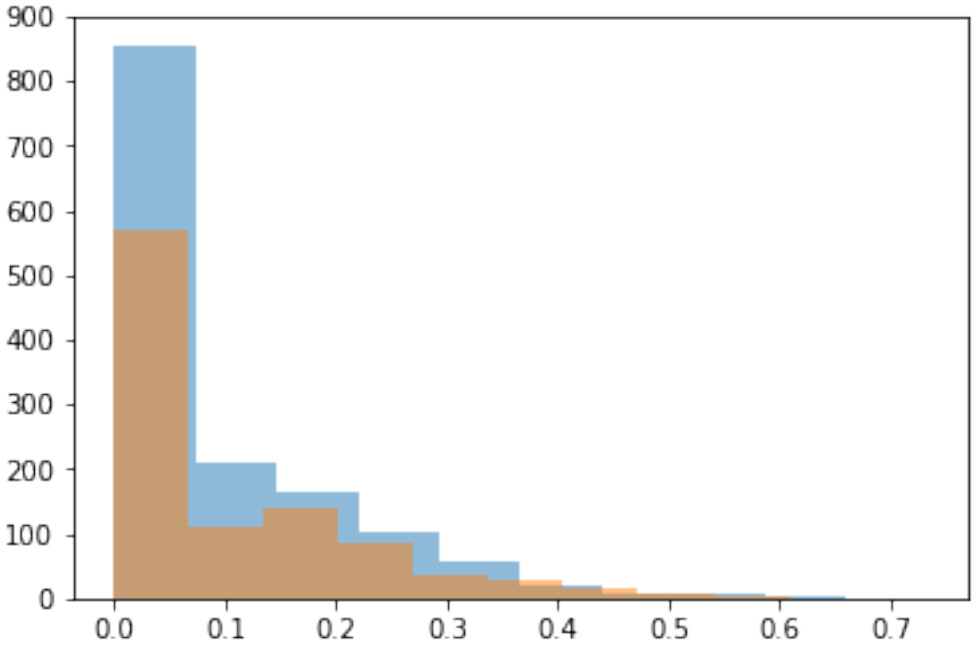
```
Out[78]: (array([1431.,  332.,  306.,  178.,  101.,   44.,   20.,   12.,    4.,
                    2.]),
          array([0.    , 0.0733, 0.1466, 0.2199, 0.2932, 0.3665, 0.4398, 0.5131,
                 0.5864, 0.6597, 0.733 ]),
          <a list of 10 Patch objects>)
```



In [79]:
```
import matplotlib.pyplot as plt

plt.hist(df_train.loc[df_train['class'] == 0]["neg"])
plt.hist(df_train.loc[df_train['class'] == 1]["neg"])
plt.show()
```

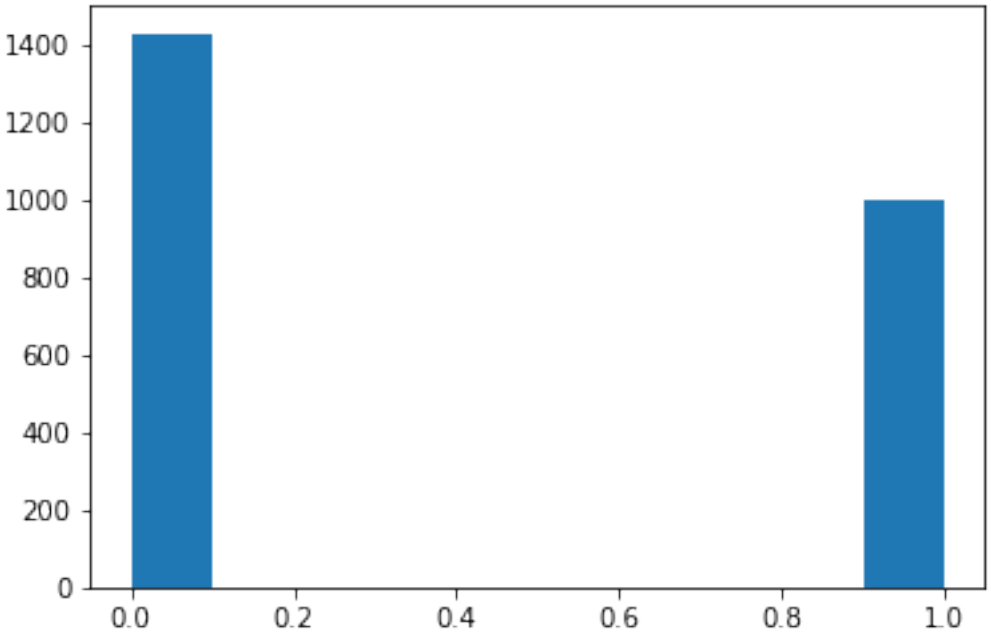


```
In [80]: plt.hist(df_train.loc[df_train['class'] == 0]["pos"], alpha=0.5)
         plt.hist(df_train.loc[df_train['class'] == 1]["pos"],alpha=0.5)
         plt.show()
```

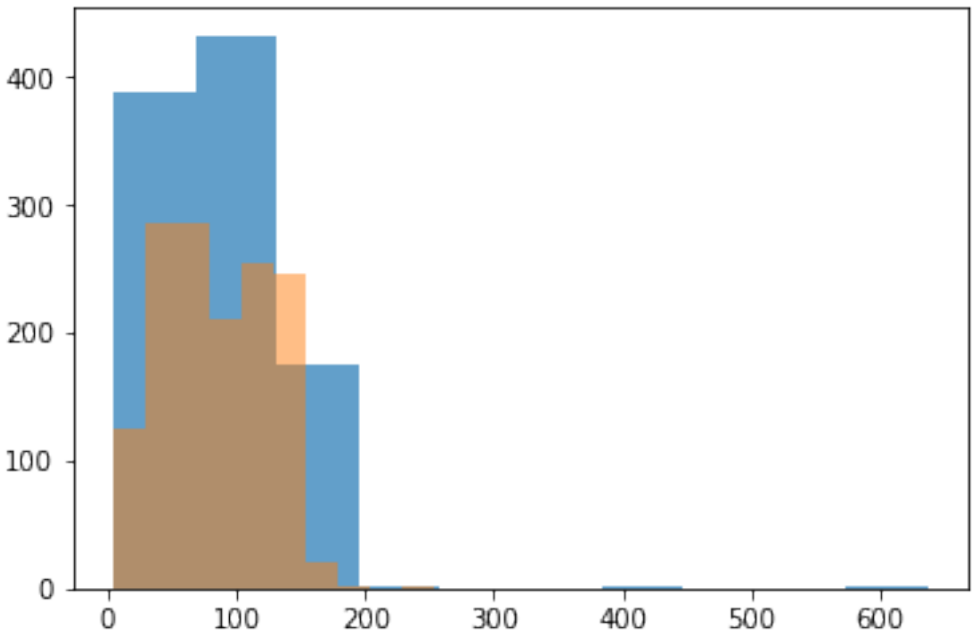



```
In [81]: plt.hist(df_train['class'])
```

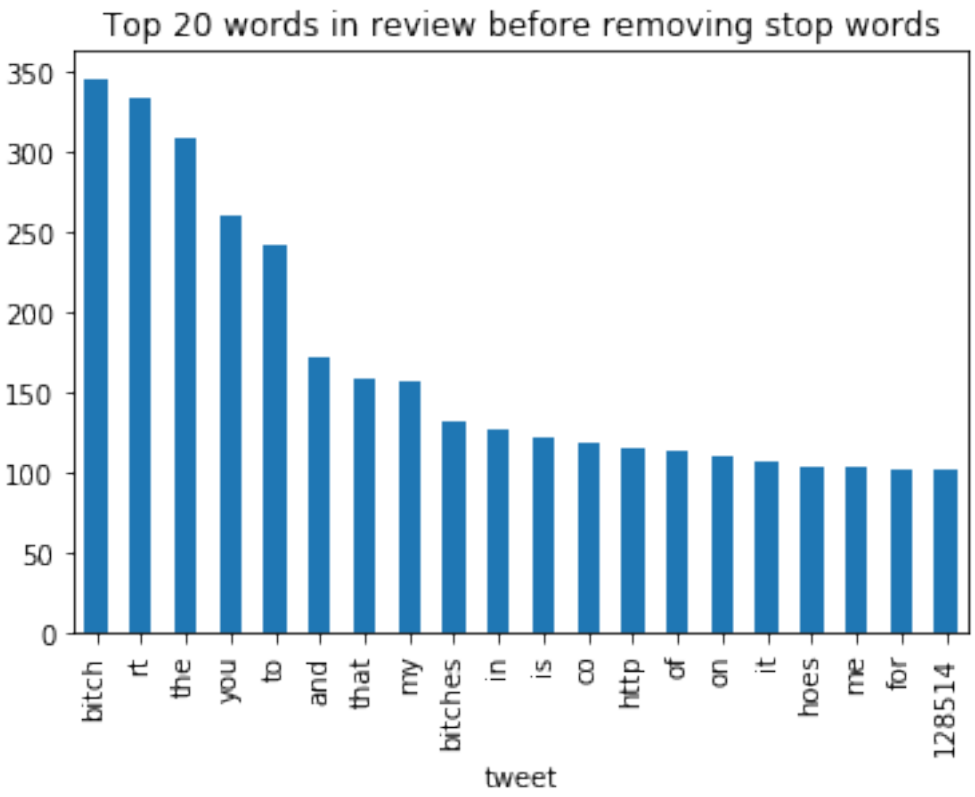
```
Out[81]: (array([1430.,    0.,    0.,    0.,    0.,    0.,    0.,    0.,    0.,
                  1000.]),
          array([0. , 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1. ]),
          <a list of 10 Patch objects>)
```



In [83]:
```
plt.hist(df_train.loc[df_train['class'] == 1]["len"],alpha=0.7)
plt.hist(df_train.loc[df_train['class'] == 0]["len"],alpha=0.5)
plt.show()
```



```
In [0]: def get_top_n_words(corpus, n=None):
            vec = CountVectorizer().fit(corpus)
            bag_of_words = vec.transform(corpus)
            sum_words = bag_of_words.sum(axis=0)
            words_freq = [(word, sum_words[0, idx]) for word, idx in vec.vocabulary_.items()]
            words_freq =sorted(words_freq, key = lambda x: x[1], reverse=True)
            return words_freq[:n]

In [85]: temp_df=df_train.loc[df_train['class'] == 1]["tweet"]
         common_words = get_top_n_words(temp_df, 20)
         for word, freq in common_words:
             print(word, freq)
         df1 = pd.DataFrame(common_words, columns = ['tweet' , 'count'])
         df1.groupby('tweet').sum()['count'].sort_values(ascending=False).plot(
             kind='bar', title='Top 20 words in review before removing stop words')

bitch 346
rt 333
the 309
you 260
to 242
and 172
that 159
my 156
bitches 132
in 126
is 122
co 119
http 115
of 113
on 110
it 106
hoes 104
me 103
for 102
128514 101


Out[85]: <matplotlib.axes._subplots.AxesSubplot at 0x7f3dccfcadd8>
```

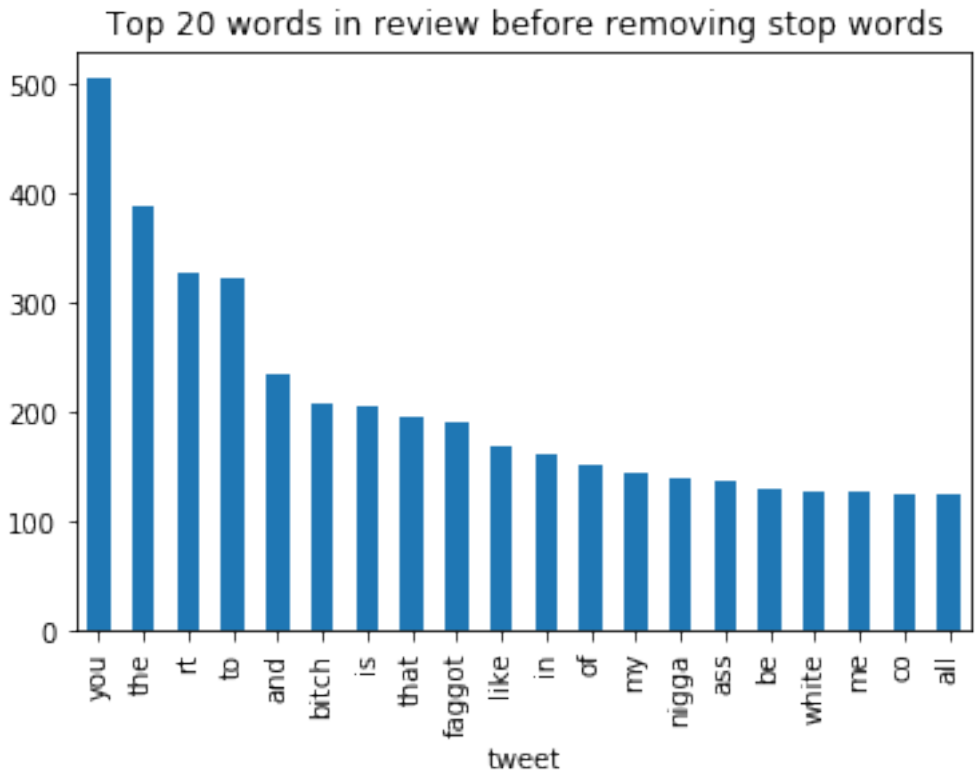



```
In [86]: temp_df=df_train.loc[df_train['class'] == 0]["tweet"]
         common_words = get_top_n_words(temp_df, 20)
         for word, freq in common_words:
             print(word, freq)
         df1 = pd.DataFrame(common_words, columns = ['tweet' , 'count'])
         df1.groupby('tweet').sum()['count'].sort_values(ascending=False).plot(
             kind='bar', title='Top 20 words in review before removing stop words')
```

```
you 505
the 388
rt 328
to 323
and 234
bitch 209
is 205
that 196
faggot 191
like 169
in 163
of 153
my 145
nigga 140
```

```
ass 138
be 131
white 127
me 127
all 126
co 126
```

Out[86]: <matplotlib.axes._subplots.AxesSubplot at 0x7f3dd14deb00>

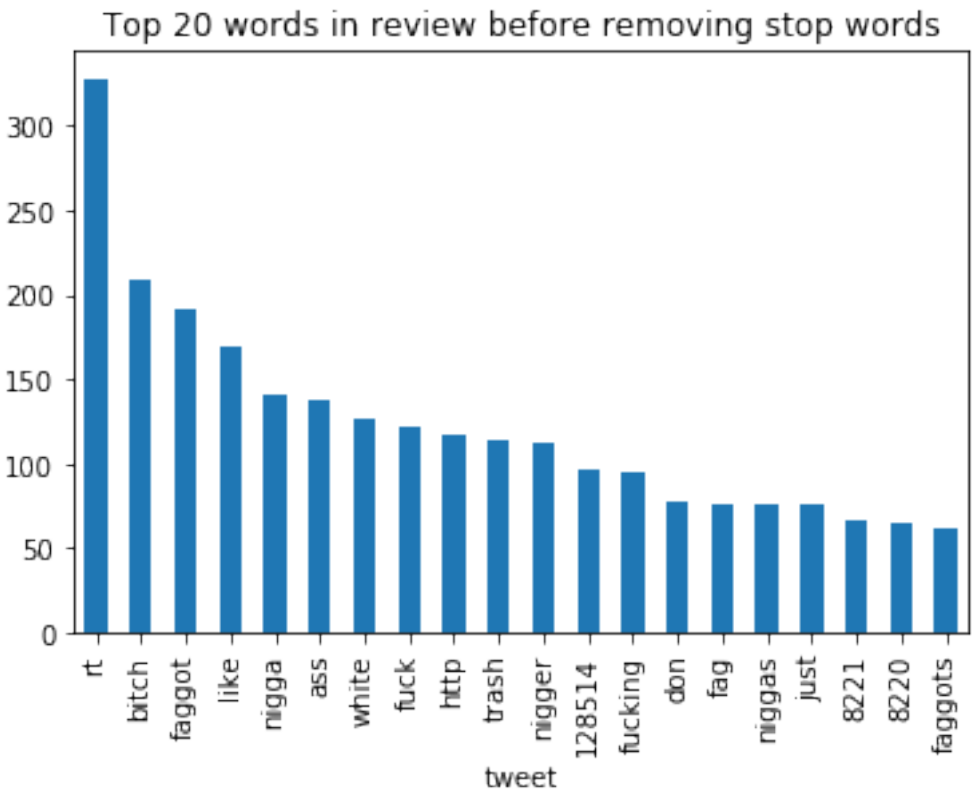



```
In [0]: def get_top_n_words(corpus, n=None):
            vec = CountVectorizer(stop_words = 'english').fit(corpus)
            bag_of_words = vec.transform(corpus)
            sum_words = bag_of_words.sum(axis=0)
            words_freq = [(word, sum_words[0, idx]) for word, idx in vec.vocabulary_.items()]
            words_freq =sorted(words_freq, key = lambda x: x[1], reverse=True)
            return words_freq[:n]

In [88]: temp_df=df_train.loc[df_train['class'] == 0]["tweet"]
         common_words = get_top_n_words(temp_df, 20)
         for word, freq in common_words:
             print(word, freq)
```

```
        df1 = pd.DataFrame(common_words, columns = ['tweet' , 'count'])
        df1.groupby('tweet').sum()['count'].sort_values(ascending=False).plot(
            kind='bar', title='Top 20 words in review before removing stop words')
```

```
rt 328
bitch 209
faggot 191
like 169
nigga 140
ass 138
white 127
fuck 121
http 117
trash 113
nigger 112
128514 96
fucking 95
don 77
niggas 76
fag 76
just 75
8221 66
8220 64
faggots 62
```

```
Out[88]: <matplotlib.axes._subplots.AxesSubplot at 0x7f3dcfc88a90>
```

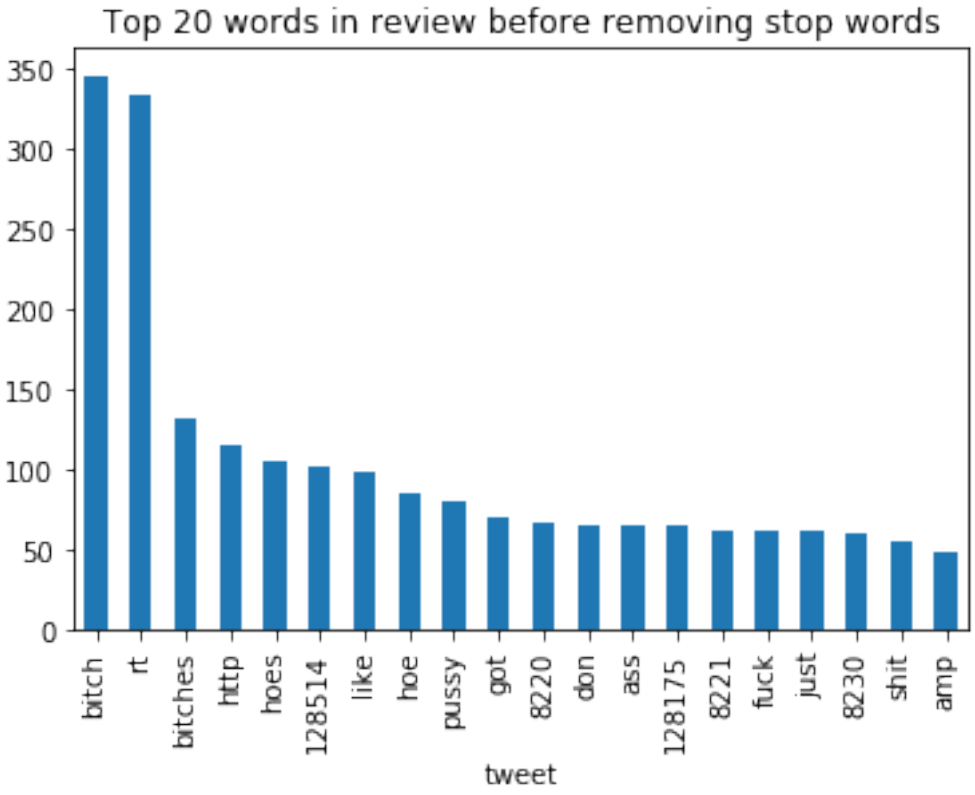



```
In [89]: temp_df=df_train.loc[df_train['class'] == 1]["tweet"]
         common_words = get_top_n_words(temp_df, 20)
         for word, freq in common_words:
             print(word, freq)
         df1 = pd.DataFrame(common_words, columns = ['tweet' , 'count'])
         df1.groupby('tweet').sum()['count'].sort_values(ascending=False).plot(
             kind='bar', title='Top 20 words in review before removing stop words')
```

```
bitch 346
rt 333
bitches 132
http 115
hoes 104
128514 101
like 98
hoe 84
pussy 79
got 70
8220 66
ass 64
don 64
128175 64
```

```
just 62
fuck 62
8221 62
8230 60
shit 55
amp 48
```

```
Out[89]: <matplotlib.axes._subplots.AxesSubplot at 0x7f3dd094dc18>
```



0.1 Bigrams

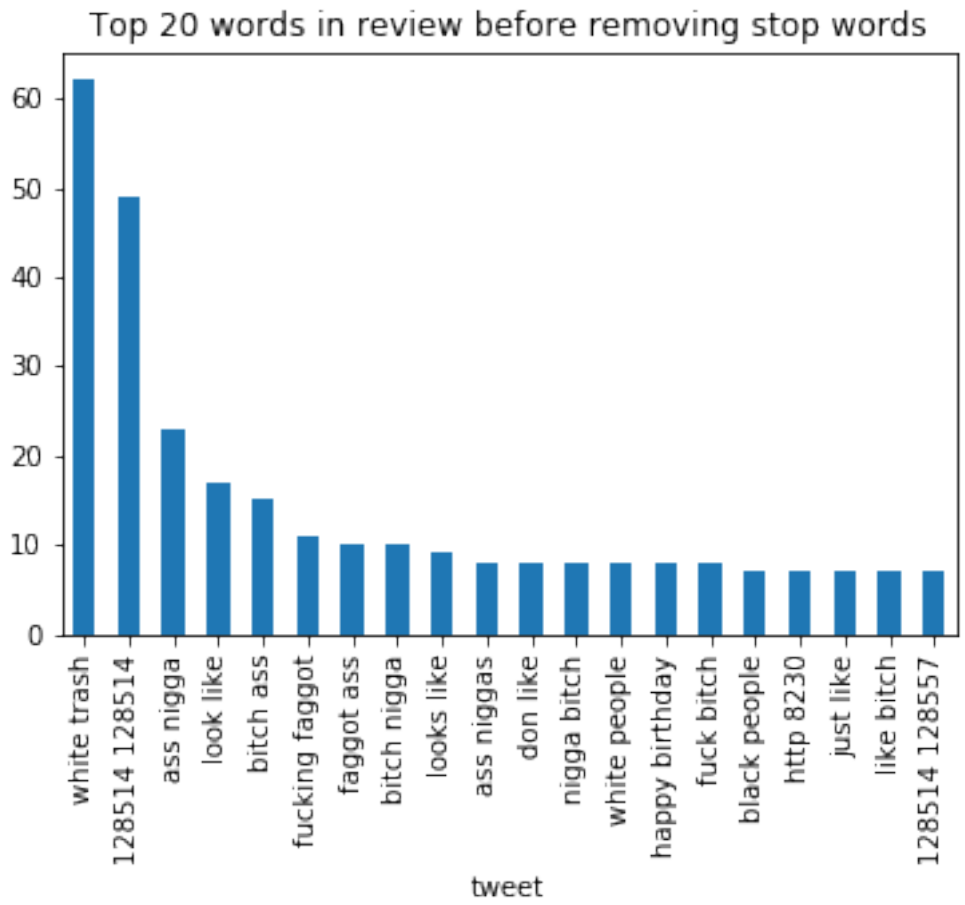
```
In [0]: def get_top_n_words(corpus, n=None):
            vec = CountVectorizer(ngram_range=(2, 2), stop_words = 'english').fit(corpus)
            bag_of_words = vec.transform(corpus)
            sum_words = bag_of_words.sum(axis=0)
            words_freq = [(word, sum_words[0, idx]) for word, idx in vec.vocabulary_.items()]
            words_freq =sorted(words_freq, key = lambda x: x[1], reverse=True)
            return words_freq[:n]
```

```
In [91]: temp_df=df_train.loc[df_train['class'] == 0]["tweet"]
         common_words = get_top_n_words(temp_df, 20)
```

```
        for word, freq in common_words:
            print(word, freq)
        df1 = pd.DataFrame(common_words, columns = ['tweet' , 'count'])
        df1.groupby('tweet').sum()['count'].sort_values(ascending=False).plot(
            kind='bar', title='Top 20 words in review before removing stop words')
```

```
white trash 62
128514 128514 49
ass nigga 23
look like 17
bitch ass 15
fucking faggot 11
faggot ass 10
bitch nigga 10
looks like 9
white people 8
nigga bitch 8
ass niggas 8
fuck bitch 8
don like 8
happy birthday 8
like bitch 7
128514 128557 7
just like 7
black people 7
http 8230 7
```

```
Out[91]: <matplotlib.axes._subplots.AxesSubplot at 0x7f3dcfd22518>
```

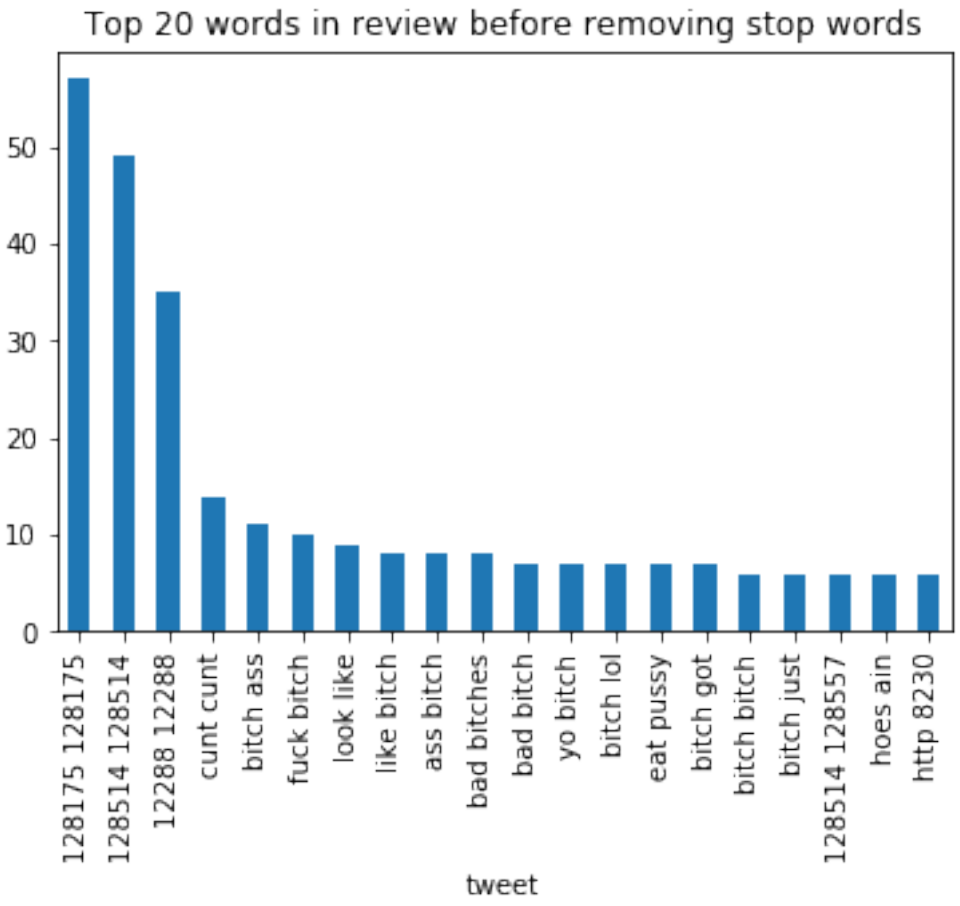



```
In [92]: temp_df=df_train.loc[df_train['class'] == 1]["tweet"]
         common_words = get_top_n_words(temp_df, 20)
         for word, freq in common_words:
             print(word, freq)
         df1 = pd.DataFrame(common_words, columns = ['tweet' , 'count'])
         df1.groupby('tweet').sum()['count'].sort_values(ascending=False).plot(
             kind='bar', title='Top 20 words in review before removing stop words')
```

```
128175 128175 57
128514 128514 49
12288 12288 35
cunt cunt 14
bitch ass 11
fuck bitch 10
look like 9
ass bitch 8
like bitch 8
bad bitches 8
yo bitch 7
```

```
bitch lol 7
eat pussy 7
bad bitch 7
bitch got 7
bitch just 6
hoes ain 6
bitch bitch 6
128514 128557 6
http 8230 6
```

Out[92]: <matplotlib.axes._subplots.AxesSubplot at 0x7f3dcfcc84a8>



1 Trigrams

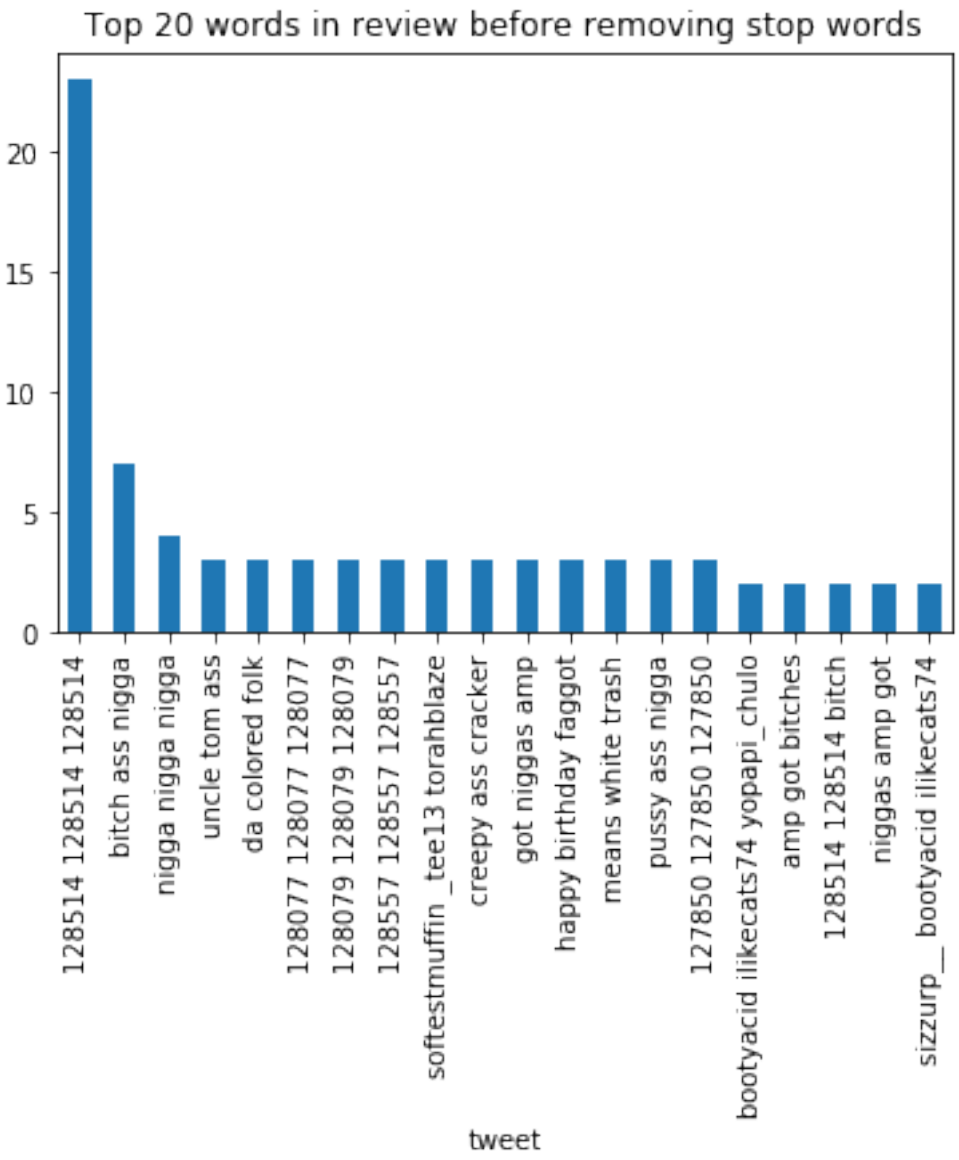
```
In [0]: def get_top_n_words(corpus, n=None):
            vec = CountVectorizer(ngram_range=(3, 3), stop_words = 'english').fit(corpus)
            bag_of_words = vec.transform(corpus)
```

```
    sum_words = bag_of_words.sum(axis=0)
    words_freq = [(word, sum_words[0, idx]) for word, idx in vec.vocabulary_.items()]
    words_freq =sorted(words_freq, key = lambda x: x[1], reverse=True)
    return words_freq[:n]
```

```
In [94]: temp_df=df_train.loc[df_train['class'] == 0]["tweet"]
         common_words = get_top_n_words(temp_df, 20)
         for word, freq in common_words:
             print(word, freq)
         df1 = pd.DataFrame(common_words, columns = ['tweet' , 'count'])
         df1.groupby('tweet').sum()['count'].sort_values(ascending=False).plot(
             kind='bar', title='Top 20 words in review before removing stop words')
```

```
128514 128514 128514 23
bitch ass nigga 7
nigga nigga nigga 4
got niggas amp 3
uncle tom ass 3
creepy ass cracker 3
pussy ass nigga 3
da colored folk 3
softestmuffin _tee13 torahblaze 3
happy birthday faggot 3
means white trash 3
127850 127850 127850 3
128079 128079 128079 3
128077 128077 128077 3
128557 128557 128557 3
niggas amp got 2
amp got bitches 2
sizzurp__ bootyacid ilikecats74 2
bootyacid ilikecats74 yopapi_chulo 2
128514 128514 bitch 2
```

```
Out[94]: <matplotlib.axes._subplots.AxesSubplot at 0x7f3dcfccb4a8>
```

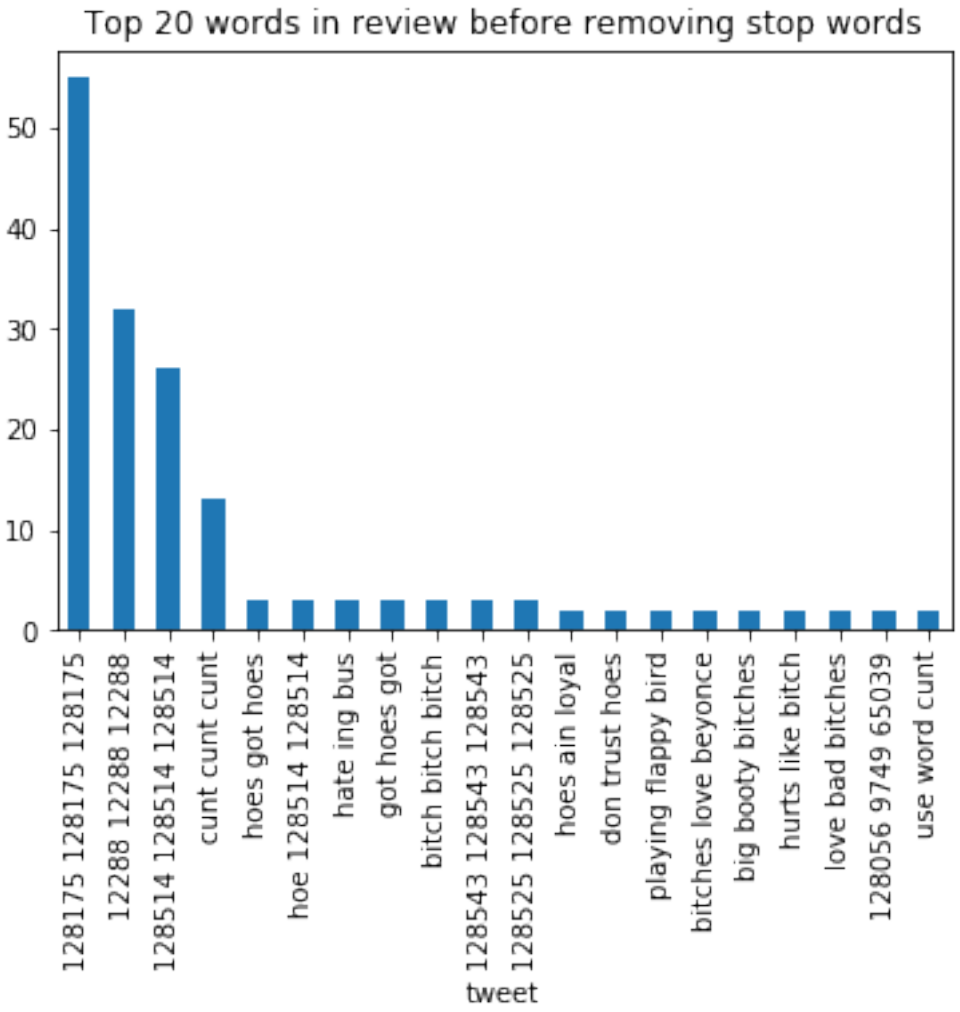



```
In [95]: temp_df=df_train.loc[df_train['class'] == 1]["tweet"]
         common_words = get_top_n_words(temp_df, 20)
         for word, freq in common_words:
             print(word, freq)
         df1 = pd.DataFrame(common_words, columns = ['tweet' , 'count'])
         df1.groupby('tweet').sum()['count'].sort_values(ascending=False).plot(
             kind='bar', title='Top 20 words in review before removing stop words')
```

```
128175 128175 128175 55
12288 12288 12288 32
128514 128514 128514 26
cunt cunt cunt 13
```

```
128525 128525 128525 3
hate ing bus 3
128543 128543 128543 3
bitch bitch bitch 3
got hoes got 3
hoes got hoes 3
hoe 128514 128514 3
bitches love beyonce 2
don trust hoes 2
hoes ain loyal 2
128056 9749 65039 2
hurts like bitch 2
big booty bitches 2
use word cunt 2
love bad bitches 2
playing flappy bird 2
```

Out[95]: <matplotlib.axes._subplots.AxesSubplot at 0x7f3dcd0d4898>



2 POS tags

In [96]:
```
!pip install TextBlob
nltk.download('punkt')
nltk.download('averaged_perceptron_tagger')
```

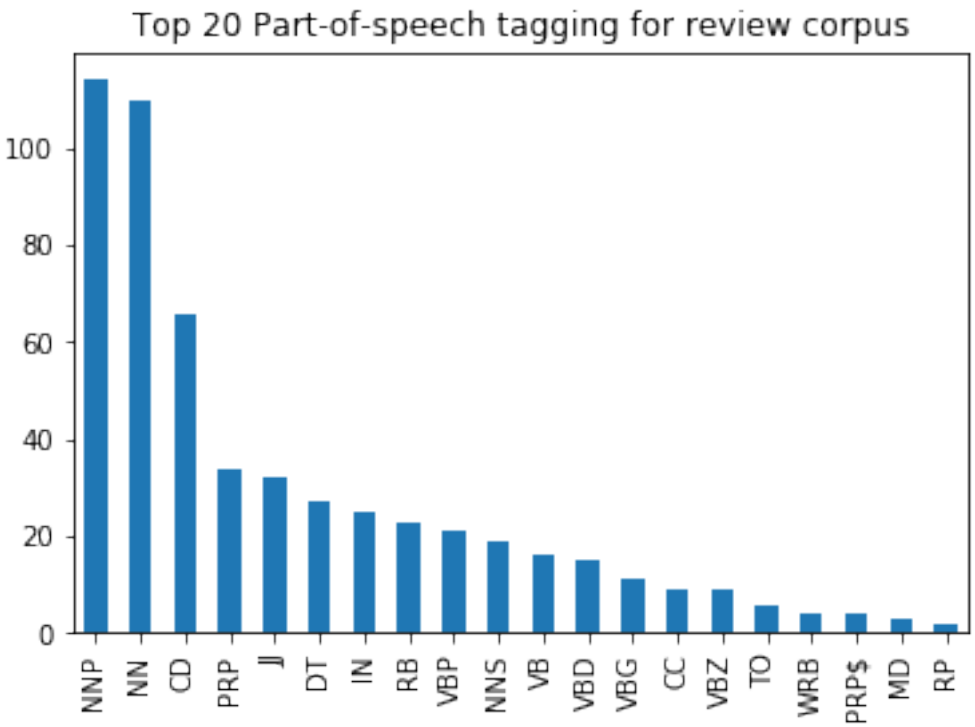
```
Requirement already satisfied: TextBlob in /usr/local/lib/python3.6/dist-packages (0.15.3)
Requirement already satisfied: nltk>=3.1 in /usr/local/lib/python3.6/dist-packages (from TextBl
Requirement already satisfied: six in /usr/local/lib/python3.6/dist-packages (from nltk>=3.1->T
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package averaged_perceptron_tagger to
[nltk_data]     /root/nltk_data...
[nltk_data]   Package averaged_perceptron_tagger is already up-to-
[nltk_data]       date!
```

Out[96]:
```
True
```

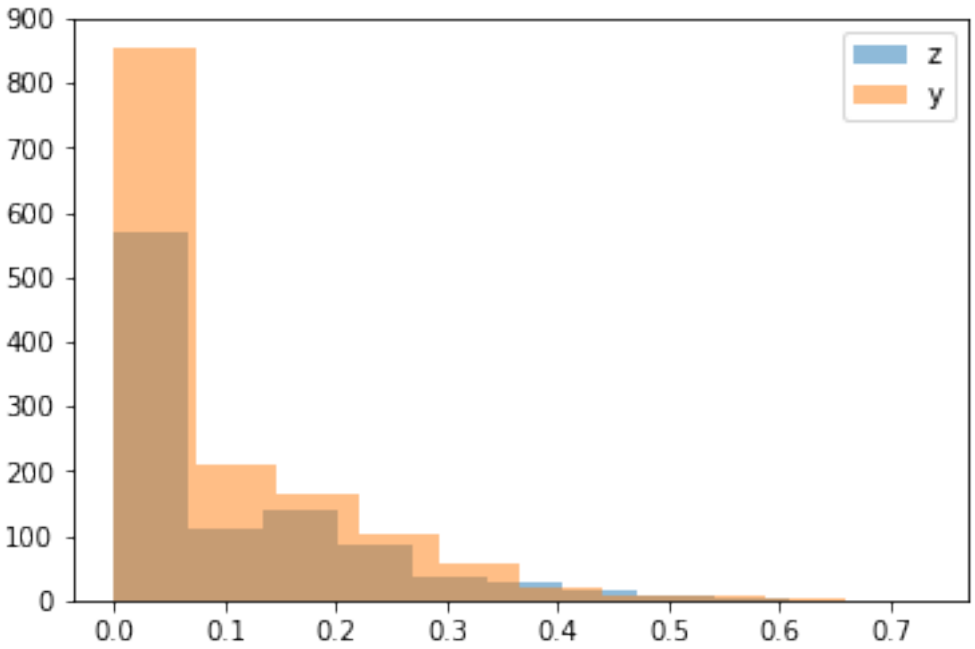
In [0]:
```
import textblob
from textblob import TextBlob
```

In [98]:
```
blob = TextBlob(str(df_train['tweet']))
pos_df = pd.DataFrame(blob.tags, columns = ['word' , 'pos'])
pos_df = pos_df.pos.value_counts()[:20]
pos_df.plot(
    kind='bar',
    title='Top 20 Part-of-speech tagging for review corpus')
```

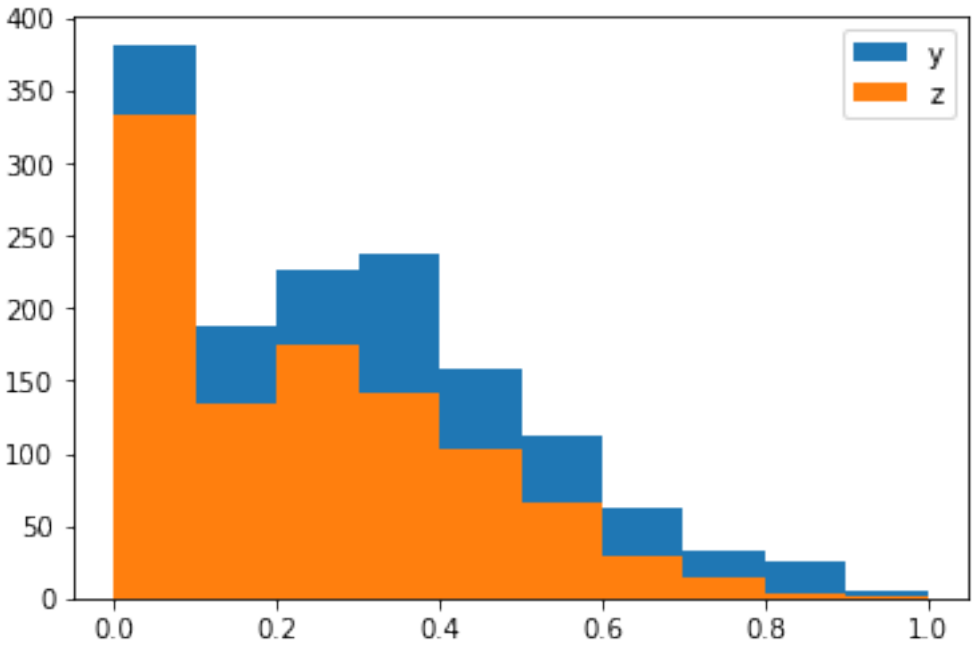
Out[98]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f3dcaeb5a90>
```



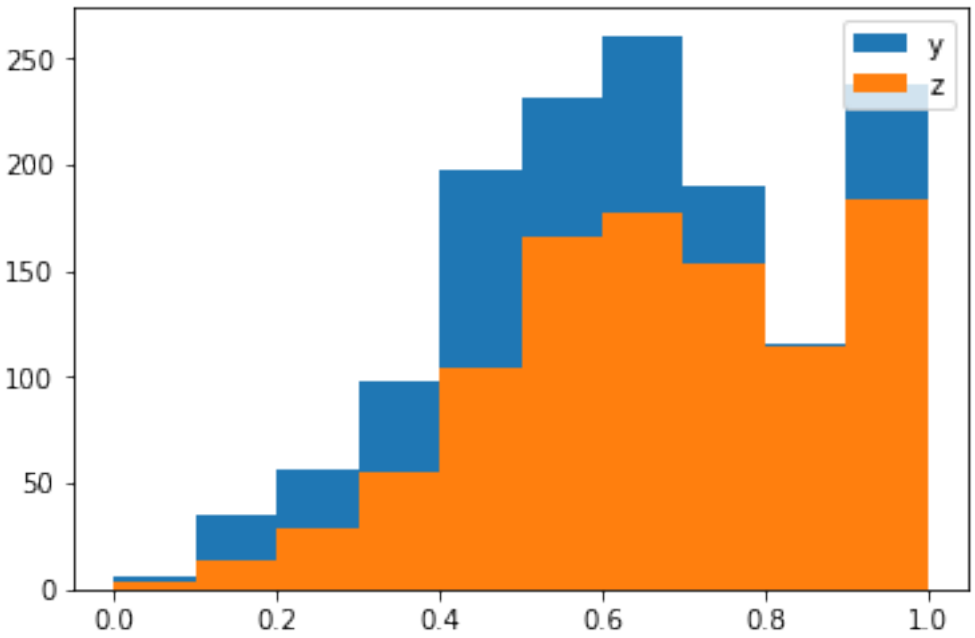
In [100]:
```
from matplotlib import pyplot

y = df_train.loc[df_train['class'] == 0, 'pos']
z = df_train.loc[df_train['class'] == 1, 'pos']

pyplot.hist(z,  label='z', alpha=0.5)
pyplot.hist(y,  label='y', alpha=0.5)
pyplot.legend(loc='upper right')
pyplot.show()
```



In [102]:
```
from matplotlib import pyplot

y = df_train.loc[df_train['class'] == 0, 'neg']
z = df_train.loc[df_train['class'] == 1, 'neg']


pyplot.hist(y,  label='y')
pyplot.hist(z,  label='z')
pyplot.legend(loc='upper right')
pyplot.show()
```

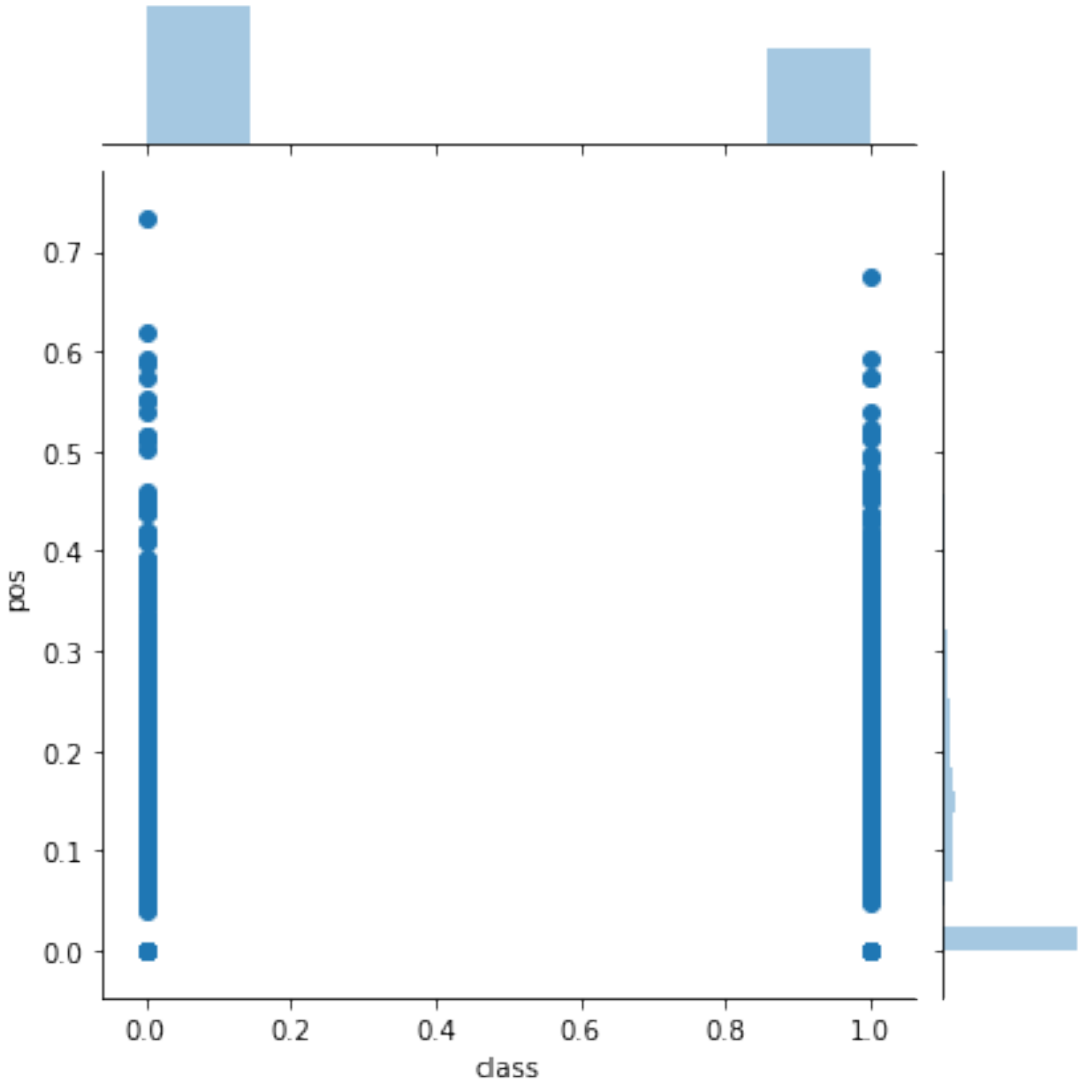


In [103]:
```
from matplotlib import pyplot

y = df_train.loc[df_train['class'] == 0, 'neu']
z = df_train.loc[df_train['class'] == 1, 'neu']


pyplot.hist(y,  label='y')
pyplot.hist(z,  label='z')
pyplot.legend(loc='upper right')
pyplot.show()
```



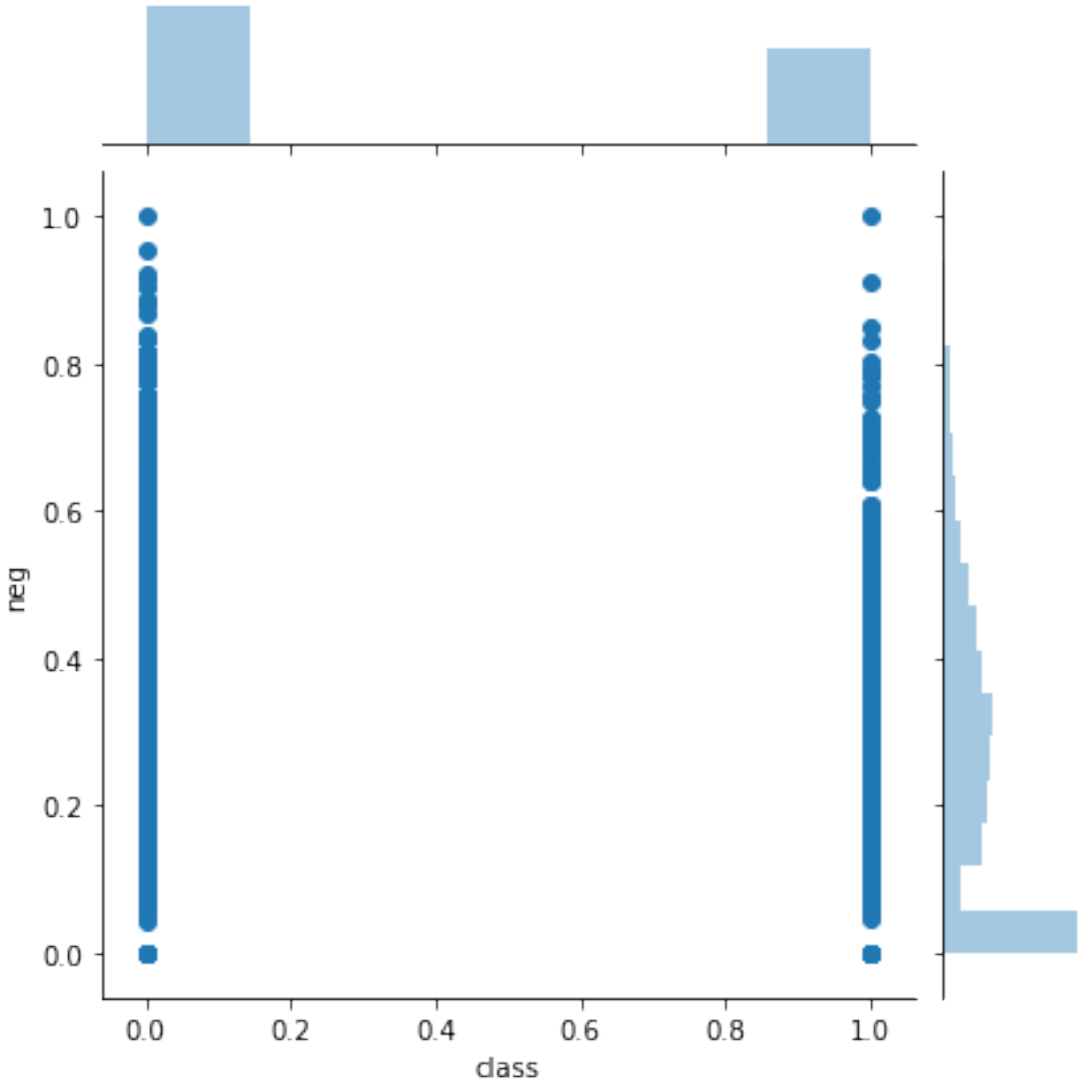

```
In [104]: import numpy as np
          import pandas as pd
          import seaborn as sns
          import matplotlib.pyplot as plt

          df=df_train[["class","pos"]]
          sns.jointplot(x="class", y="pos", data=df)
```

```
Out[104]: <seaborn.axisgrid.JointGrid at 0x7f3dcabf8ef0>
```



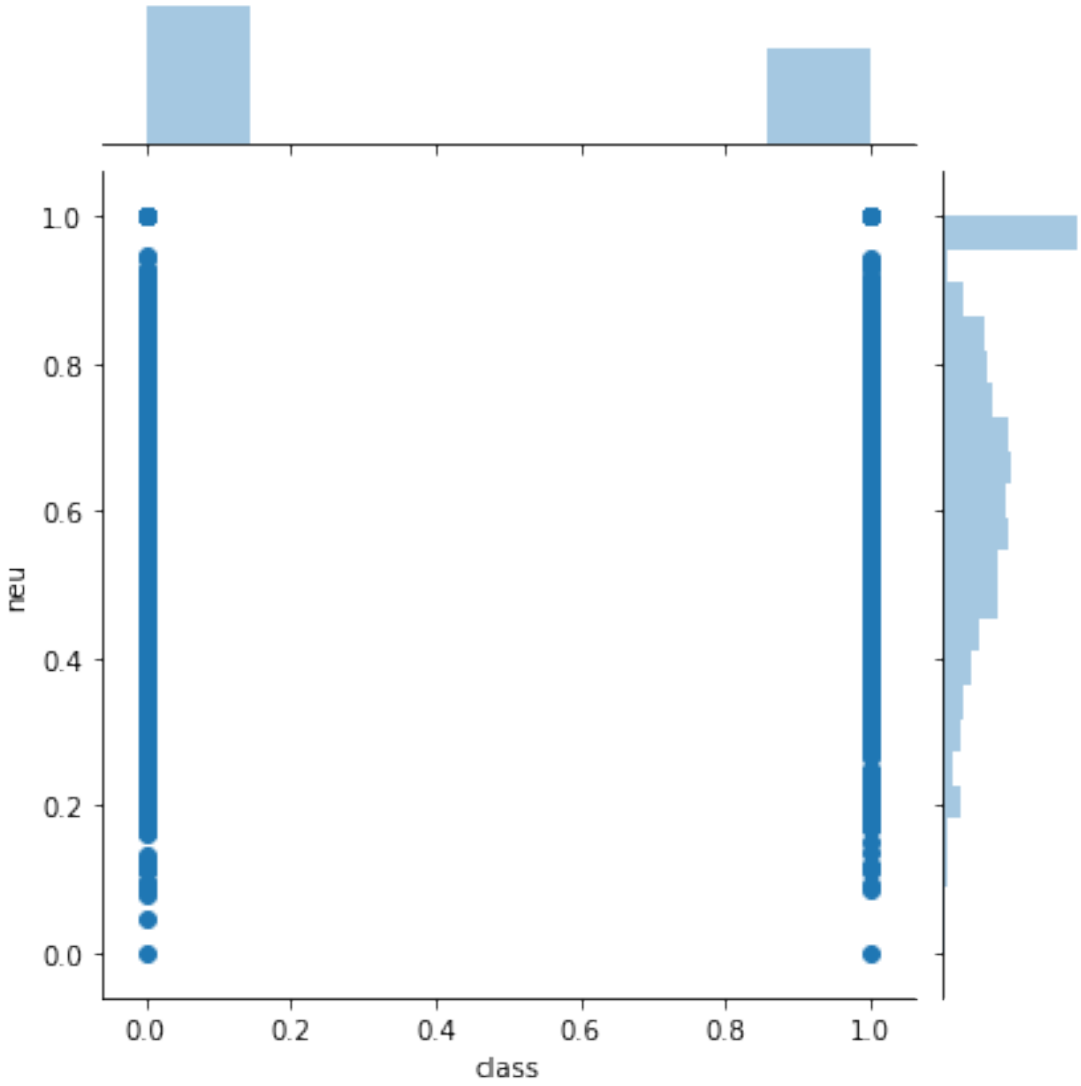

```
In [105]: df=df_train[["class","neg"]]
          sns.jointplot(x="class", y="neg", data=df)
```

```
Out[105]: <seaborn.axisgrid.JointGrid at 0x7f3dc9fc3198>
```




```
In [106]: df=df_train[["class","neu"]]
          sns.jointplot(x="class", y="neu", data=df)
```

```
Out[106]: <seaborn.axisgrid.JointGrid at 0x7f3dc9e3aa20>
```


1.0
0.8
0.6
0.4
0.2
0.0
neu
0.0
0.2
0.4
0.6
0.8
1.0
class