



ANKARA ÜNİVERSİTESİ

Mühendislik Fakültesi

BLM3551 – Yapay Zeka

Proje Raporu

TweePy

Sosyal Medya Duygu/Düşünce Analizi



Akın Deniz 18290014

Berrak Küçük 18290038

Gülperi Tunçyürek 18290061

Özet

Sosyal medya, günümüzde bireyler ve toplumlar üzerinde büyük bir etkiye ve öneme sahip. Haber ve bilgi paylaşımı ile gündemin oluşması, insanların kendi fikirlerini paylaşarak büyük bir etkileşim oluşturmaları; farkında olmadan da olsa insanların gündemdeki konulara karşı bakış açısını oluşturur. Bu denli büyük bir etkileşimin yararları olduğu gibi negatif yönleri de vardır. Bazen bu etkileşimler nefret yorumlarına ve siber zorbalığa sebebiyet verebilir. Bu nedenle bu projede, bir dizi tweet içeren bir veri setini Makine Öğrenimi kullanarak belirli bir tweet ile ilişkili duyguların tespit edilmesinde kullandık. İnternetteki kullanıcılar tarafından atılmış tweet'lerin negatif veya pozitif olarak sınıflandırmasını yaptık.

İçerik

1. Giriş

1.1. Konsept

1.2. Hedefler

2. Problemin Tanımı

2.1. Görev Tanımı

2.2. Kullanılan Araçlar ve Metotlar

3. Değerlendirme ve Sonuçlar

3.1. Test Veri Seti Sonuçları ve Veri Görselleştirme

3.2. Tweepy Modülü ve Twitter API Sonuçları

4. Tartışma

5. Kaynakça

6. Ek Bölüm

6.1. Kaynak Kod

6.2. Video

1. Giriş

1.1 Konsept

Etkileşimin sürekli devam ettiği dünyada doğal dil metninin hacmi her geçen gün daha da artıyor. Geniş bir bilgi içeriğine sahip olunmasına rağmen, bir insan tarafından, özellikle bir zaman sınırlaması içinde bu metinlerinden içindeki bilgiyi keşfetmek ve yaymak oldukça zorlayıcı. Otomatikleştirilmiş doğal dil işleme bu işi, sınırlı miktarda metin için, tıpkı bir insan gibi etkili ve doğru bir şekilde yapmayı amaçlamaktadır.

1.2 Hedefler

Bu projedeki hedefimiz, Twitter'daki kullanıcıların atmış olduğu tweet'leri pozitif ya da negatif olacak şekilde tespit etmek. Eğer tweet'te cinsiyetçi veya ırkçı kelimeler, nefret söylemi ifadeleri yer alıyorsa tweet negatif olarak işaretlenecek. Bunun için yaklaşık 32.000 tweet'ten oluşan, önceden negatif/pozitif his olarak işaretlenmiş bir veri seti kullanarak makine öğrenimi tekniğini kullandık.

Sadece şirketler için pazarlama stratejisi oluşturmada veya politik kampanyalar için değil, gerek bir ürün konusunda kullanıcıların fikir ve yorum analizinde, gerek bir hizmetle ilgili tüketici memnuniyetini ölçmede kullanılabilir. Doğal dil işlemenin en yaygın kullanım alanları sosyal medya takibi, marka takibi, müşteri destek analizi, müşteri geri bildirim analizi ve pazar araştırması. Bu geniş skalaya sahip kullanımı sebebiyle bu konuda oldukça fazla çalışma yapılmış durumda. Farklı kullanım amaçlarına hitap eden birçok makale ve çalışmaya ulaşmak mümkün.

2. Problemin Tanımı

2.1. Görev Tanımı

Etiketlenmiş tweet'lerde, '1' etiketinin ırkçı/cinsiyetçi/nefret söylemi içeren bir ifade olduğunu, '0' etiketinin ise negatif söylemler içermeyen bir ifade olduğunu belirterek, amacımız bu eğitim örneğini kullanarak veri setimizi test etmek ve etiketlerini tahmin etmektir.

Burada id, verilen veri setindeki tweet'lerle ilişkilendirilen numaradır.

2.2. Kullanılan Araçlar ve Metotlar

Kodu Jupyter Notebook üzerinde çalıştırdık. Kod içerisinde Natural Language Toolkit ve Pandas, Numpy, Matplotlib gibi kütüphaneler, veri görselleştirme için Seaborn kütüphanesini kullandık.

```
In [15]: import re
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import string
import nltk
import warnings
import os

warnings.filterwarnings("ignore", category=DeprecationWarning)
```

```
In [18]: %matplotlib inline
train = pd.read_csv("train.csv", engine='python')
train.head()
```

```
Out[18]:
```

	id	label	tweet
0	1	0	@user when a father is dysfunctional and is s...
1	2	0	@user @user thanks for #lyft credit i can't us...
2	3	0	bihday your majesty
3	4	0	#model i love u take with u all the time in ...
4	5	0	factsguide: society now #motivation

Görselde Pandas kullanarak train.csv dosyasını okuduk ve yapacağımız değişikliklere karşı bir kopyasını oluşturduk. Sonraki adımda aynısını test.csv dosyası için yaptık. Böylece öğrenme ve test için kullanacağımız veri setlerini okumuş olduk. Bunları bir dosyada birleştirdik.

```
In [22]: test.tail()
```

```
Out[22]:
```

	id	tweet
17192	49155	thought factory: left-right polarisation! #tru...
17193	49156	feeling like a mermaid Å°ÅŸÅ°Å°~Å°~ #hairflip #nev...
17194	49157	#hillary #campaigned today in #ohio((omg)) &am...
17195	49158	happy, at work conference: right mindset leads...
17196	49159	my song "so glad" free download! #shoegaze ...

```
In [23]: test_original = test.copy()
```

```
In [24]: combine = train.append(test, ignore_index=True, sort=True)
combine.head()
```

```
Out[24]:
```

	id	label	tweet
0	1	0.0	@user when a father is dysfunctional and is s...
1	2	0.0	@user @user thanks for #lyft credit i can't us...
2	3	0.0	bihday your majesty
3	4	0.0	#model i love u take with u all the time in ...
4	5	0.0	factsguide: society now #motivation

Ardından makine öğrenmesi için işimize yaramayacak bazı karakterleri, tweet'lerden kaldırıyoruz. Bunun sebebi bu karakterlerin his analizinde negatif veya pozitif anlam ifade etmemesi.

```
In [26]: def remove_pattern(text, pattern):
         r = re.findall(pattern, text)

         for i in r:
             text = re.sub(i, "", text)

         return text
```

```
In [33]: combine['Tidy_Tweets'] = np.vectorize(remove_pattern)(combine['tweet'], "@[\w]*")
```

```
In [34]: combine.head()
```

Out[34]:

	id	label	tweet	Tidy_Tweets
0	1	0.0	@user when a father is dysfunctional and is s...	when a father is dysfunctional and is so sel...
1	2	0.0	@user @user thanks for #lyft credit i can't us...	thanks for #lyft credit i can't use cause th...
2	3	0.0	bihday your majesty	bihday your majesty
3	4	0.0	#model i love u take with u all the time in ...	#model i love u take with u all the time in ...
4	5	0.0	factsguide: society now #motivation	factsguide: society now #motivation

Kaldırdığımız karakterler “@user_name” ifadeleri, noktalama işaretleri, sayılar ve özel karakterler. Bunların yerini boşlukla değiştirdik. Yaptığımız başka bir düzenleme ise kısa kelimeleri kaldırmak oldu. Bunun sebebi İngilizce’de bulunan ve tek başına anlam ifade etmeyen kelimelerin analizde bir faydası olmaması. Bu temizleme işlemlerinden sonra veri setimizdeki tüm temizlenmiş tweet'leri belirttik. Girdiler ayrı terimler veya kelimelerdir ve belirteçler, bir metin dizesini belirteçlere bölme işlemidir. Burada cümlelerimizi belirtiyoruz çünkü sonraki adımda "NLTK" paketinden Stemming'i uygulayacağız. Stemming, veri setlerimizin dili İngilizce olduğu için uyguladığımız bir adım ve kısaca bir kelimeden son ekleri ("ing", "ly", "es", "s" vb.) ayırmanın kural tabanlı bir işlemi. Bu adımdan sonra birleştirme işlemi uyguladık.

```
In [39]: tokenized_tweet = combine['Tidy_Tweets'].apply(lambda x: x.split())
```

```
In [40]: tokenized_tweet.head()
```

```
Out[40]: 0    [when, a, father, is, dysfunctional, and, is, ...  
1    [thanks, for, #lyft, credit, i, can, t, use, c...  
2              [bihday, your, majesty]  
3    [#model, i, love, u, take, with, u, all, the, ...  
4              [factsguide, society, now, #motivation]  
Name: Tidy_Tweets, dtype: object
```

```
In [41]: from nltk import PorterStemmer  
ps = PorterStemmer()  
tokenized_tweet = tokenized_tweet.apply(lambda x: [ps.stem(i) for i in x])  
tokenized_tweet.head()
```

```
Out[41]: 0    [when, a, father, is, dysfunct, and, is, so, s...  
1    [thank, for, #lyft, credit, i, can, t, use, ca...  
2              [bihday, your, majesti]  
3    [#model, i, love, u, take, with, u, all, the, ...  
4              [factsguid, societi, now, #motiv]  
Name: Tidy_Tweets, dtype: object
```

```
In [42]: for i in range(len(tokenized_tweet)):  
    tokenized_tweet[i] = ' '.join(tokenized_tweet[i])  
  
combine['Tidy_Tweets'] = tokenized_tweet  
combine.head()
```

```
Out[42]:
```

	id	label	tweet	Tidy_Tweets
0	1	0.0	@user when a father is dysfunctional and is s...	when a father is dysfunct and is so selfish he...
1	2	0.0	@user @user thanks for #lyft credit i can't us...	thank for #lyft credit i can t use caus they d...
2	3	0.0	bihday your majesty	bihday your majesti
3	4	0.0	#model i love u take with u all the time in ...	#model i love u take with u all the time in ur
4	5	0.0	factsguide: society now #motivation	factsguid societi now #motiv

Uyguladığımız bu adımlar, metinsel veri içeren bir veri kümesini ön işlemden geçirmemiz izlenecek temel adımlardır.

3. Değerlendirme ve Sonuçlar

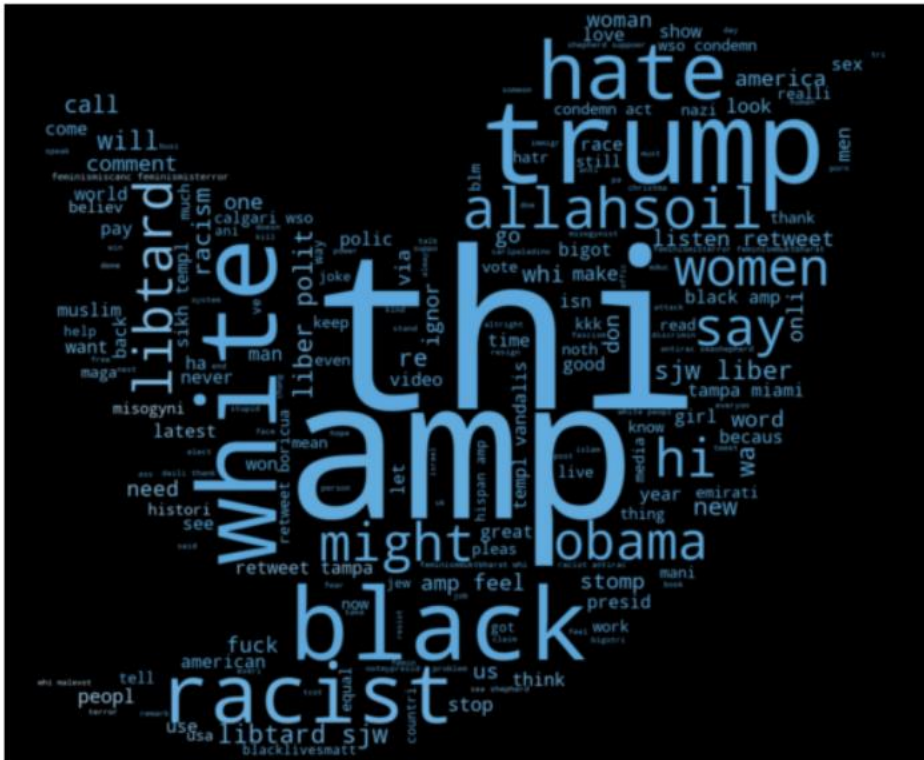
3.1. Test Veri Seti Sonuçları

Sonuçlarımızı veri görselleştirme ile birleştirdik. En popüler veri görselleştirme tekniklerinden biri WordCloud’du kullandık. WordCloud, en sık kullanılan kelimelerin büyük boyutta görüldüğü ve daha seyrek olan kelimelerin küçük boyutlarda görüldüğü bir görselleştirmedir. Python’da WordCloud oluşturmak için bir paketimiz var. Burada negatif işaretlenmiş tweet’lerde kullanılan kelimelerle bir kelime bulutu oluşturduk:

```
In [49]: all_words_negative = ' '.join(text for text in combine['Tidy_Tweets'][(combine['label']==1)])

In [50]: Mask = np.array(Image.open(requests.get('http://clipart-library.com/image_gallery2/Twitter-PNG-Image.png', stream=True).raw))
image_colors = ImageColorGenerator(Mask)
wc = WordCloud(background_color='black', height=1500, width=4000,mask=Mask).generate(all_words_negative)

In [51]: plt.figure(figsize=(10,20))
plt.imshow(wc.recolor(color_func=image_colors),interpolation="gaussian")
plt.axis('off')
plt.show()
```



Aynı yöntemi pozitif tweet'ler için de kullanalım:

```
In [43]: from wordcloud import WordCloud, ImageColorGenerator
         from PIL import Image
         import urllib
         import requests
```

```
In [44]: all_words_positive = ' '.join(text for text in combine['Tidy_Tweets'][combine['label']==0])
```

```
In [45]: Mask = np.array(Image.open(requests.get('http://clipart-library.com/image_gallery2/Twitter-PNG-Image.png', stream=True).raw))
image_colors = ImageColorGenerator(Mask)
wc = WordCloud(background_color='black', height=1500, width=4000, mask=Mask).generate(all_words_positive)
```

```
In [46]: plt.figure(figsize=(10,20))
plt.imshow(wc.recolor(color_func=image_colors),interpolation="hamming")
plt.axis('off')
plt.show()
```

```
In [46]: plt.figure(figsize=(10,20))
plt.imshow(wc.recolor(color_func=image_colors),interpolation="hamming")
plt.axis('off')
plt.show()
```



Kelimelerin çoğunun olumlu veya nötr olduğunu görebiliyoruz. Happy, smile ve love en sık görülenleri. Dolayısıyla, sık sık kullanılan kelimelerin çoğu olumlu duygularla tweet'lerle uyumludur.

Twitter’da yayınlanan sık sık kullanılan bir etiket de etiketidir (hashtag). Bu özellik yoğunlaşmak için kullanılır. Hashtag kullanımının tweet içermek üzere yazmak için veri içermek üzere içinde yer alan bir cümlede topladık.

```
In [52]: def Hashtags_Extract(x):
          hashtags=[]

          # Loop over the words in the tweet
          for i in x:
              ht = re.findall(r'#[\w+]',i)
              hashtags.append(ht)

          return hashtags
```

```
In [53]: ht_positive = Hashtags_Extract(combine['Tidy_Tweets'][combine['label']==0])
          ht_positive
```

```
Out[53]: [['run'],
          ['lyft', 'disapoint', 'getthank'],
          [],
          ['model'],
          ['motiv'],
          ['allshowandnogo'],
          [],
          ['school', 'exam', 'hate', 'imagin', 'actorslif', 'revolutionschool', 'girl'],
          ['allin', 'cav', 'champion', 'cleveland', 'clevelandcavali'],
          ['gr'],
          ['ireland', 'blog', 'silver', 'gold', 'forex'],
          ['orlando',
          'standwithorlando',
          'pulseshoot',
          'orlandoshoot',
          'biggerproblem',
          'selfish',
          'heabreak',
          'valu',
          'love']]
```

```
In [54]: ht_positive_unnest = sum(ht_positive,[])
```

```
In [55]: ht_positive_unnest
```

```
Out[55]: ['run',
          'lyft',
          'disapoint',
          'getthank',
          'model',
          'motiv',
          'allshowandnogo',
          'school',
          'exam',
          'hate',
          'imagin',
          'actorslif',
          'revolutionschool',
          'girl',
          'allin',
          'cav',
          'champion',
          'cleveland',
          'clevelandcavali',
          'gr']
```

Aynı şekilde negatif etiketlere de bakalım.

```
In [56]: ht_negative = Hashtags_Extract(combine['Tidy_Tweets'][combine['label']==1])
ht_negative
```

```
[[],
 ['race', 'ident', 'med'],
 ['altright', 'whitesupremaci'],
 ['linguist', 'race', 'power', 'raciolinguist'],
 ['brexit'],
 ['peopl', 'trump', 'republican'],
 ['michelleobama'],
 ['knick', 'golf'],
 ['jewishsupremacist'],
 ['libtard', 'sjw', 'liber', 'polit'],
 ['trash', 'hate'],
 [],
 [],
 ['stereotyp', 'prejudic', 'hope', 'hate', 'conflict'],
 ['pol', 'bluelivesmatt', 'draintheswamp', 'ferguson'],
 [],
 ['antisemit', 'hocoschool', 'columbiamd', 'hocomd'],
 ['nazi', 'hatr', 'bigotri'],
 ['libtard', 'sjw', 'liber', 'polit'],
 ['p', 'p', 'fyi', 'tcot'],
 ['tcot', 'p', 'p', 'fyi', 'tcot']]
```

```
In [57]: ht_negative_unnest = sum(ht_negative,[])
ht_negative_unnest
```

```
Out[57]: ['cnn',
 'michigan',
 'tcot',
 'australia',
 'opkillingbay',
 'seashepherd',
 'helpcovedolphin',
 'thecov',
 'helpcovedolphin',
 'neverump',
 'xenophobia',
```

Bu veriler için bar grafiği çizerek pozitif ve negatif hislere sahip en sık kullanılan etiketleri gösterdik. Örneğin en sık kullanılan pozitif etiket listemiz şu şekilde oldu.

```
In [58]: word_freq_positive = nltk.FreqDist(ht_positive_unnest)
word_freq_positive
```

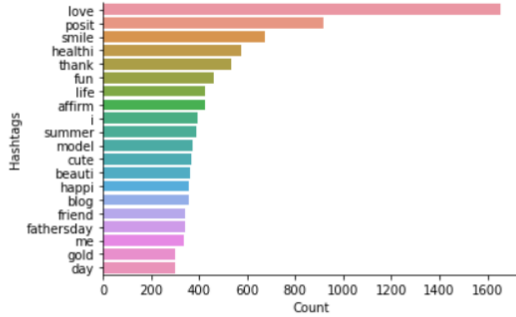
```
Out[58]: FreqDist({'love': 1654, 'posit': 917, 'smile': 676, 'healthi': 573, 'thank': 534, 'fun': 463, 'life': 425, 'affirm': 423, 'i': 394, 'summer': 390, ...})
```

```
In [59]: df_positive = pd.DataFrame({'Hashtags':list(word_freq_positive.keys()),'Count':list(word_freq_positive.values())})
df_positive.head(10)
```

Out[59]:

	Hashtags	Count
0	run	72
1	lyft	2
2	disapoint	1
3	getthank	2
4	model	375
5	motiv	202
6	allshowandnogo	1
7	school	30
8	exam	9
9	hate	27

```
In [60]: df_positive_plot = df_positive.nlargest(20,columns='Count')
sns.barplot(data=df_positive_plot,y='Hashtags',x='Count')
sns.despine()
```



```
In [61]: word_freq_negative = nltk.FreqDist(ht_negative_unnest)
word_freq_negative
```

```
Out[61]: FreqDist({'trump': 136, 'polit': 95, 'allahsoil': 92, 'liber': 81, 'libtard': 77, 'sjw': 75, 'retweet': 63, 'black': 46, 'miam i': 46, 'hate': 37, ...})
```

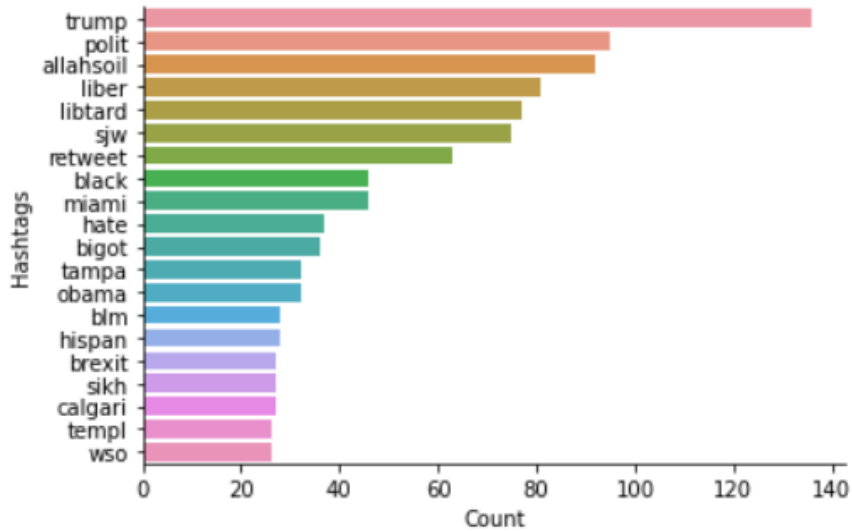
Negatif etiket listemiz ve bar grafiği ise şu şekilde:

```
In [62]: df_negative = pd.DataFrame({'Hashtags':list(word_freq_negative.keys()),'Count':list(word_freq_negative.values())})
df_negative.head(10)
```

Out[62]:

	Hashtags	Count
0	cnn	10
1	michigan	2
2	tcot	14
3	australia	6
4	opkillingbay	5
5	seashepherd	22
6	helpcovedolphin	3
7	thecov	4
8	neverump	8
9	xenophobia	12

```
In [63]: df_negative_plot = df_negative.nlargest(20,columns='Count')
sns.barplot(data=df_negative_plot,y='Hashtags',x='Count')
sns.despine()
```



3.2. Tweepy Modülü ve Sonuçları

Şimdi Natural Language Processing ve Sci-Kit Learn'ün geliştirilebilir Duygu Analizi yapılmasını bu ortamda Twitter API'ye bağlanarak gerçek zamanlı analiz yapılmasını sağlar bir modül geliştirilebilir.

Veri Seti: Veri seti “short” adında directory'nin içinde pozitif yorumlar için pozitif.txt, olumsuz yorumlar için negatif.txt dosyasında saklanmaktadır. Bir tweet data'sı kullanıp kullanmamamız çok önemli değil sadece Twitter'da olduğu gibi kısa cümleler/ifadeler içermesi yeterli olacaktır.

Şimdi bu veri setini gerçek algılamayı algılamaya duyarlı kod oluşturulabilir.

```
1 '''
2 Bu modül sentiment analizi yapmamızı sağlar.
3
4 '''
5
6 #!/usr/bin/env python
7 # coding: utf-8
8
9 # In[ ]:
10
11
12 #File: sentiment_mod.py
13
14 import nltk
15 import random
16 from nltk.classify.scikitlearn import SklearnClassifier
17 import pickle
18 from sklearn.naive_bayes import MultinomialNB, BernoulliNB
19 from sklearn.linear_model import LogisticRegression, SGDClassifier
20 from sklearn.svm import SVC, LinearSVC, NuSVC
21 from nltk.classify import ClassifierI
22 from statistics import mode
23 from nltk.tokenize import word_tokenize
24
```

Bu modül için Natural Language Tool Kit'in yanı sıra Makine Öğrenimi kütüphanesi olan Sklearn'i ve işimize yarayacak diğer modülleri de import ettik. Sonra nltk.classify toolkitinde bulunan ClassifierI classından miras alarak VoteClassifier adında bir class oluşturulur. Bu classın oluşturulma amacı Sci-Kit Learn paketinin barındırdığı farklı classifierları kullanmak ve nltk.classifydaki ClassifierI classı aracılığıyla kullanacağımız classifierların verdiği değerlerden mode olarak ve birden fazla classifierdan yararlanarak daha güvenilir bir sonuç oluşturmak. Aynı zamanda bu classta tanımlanan confidence fonksiyonu aracılığıyla da bir güvenilirlik değeri elde edilir.

Dataseti biçimlendirmek ve etiketlendirmek için öncelikle pozitif ve negatif için iki farklı değişken belirleyip sonrasında veri seti dosyaları açılır ve okumalar yapıldıktan sonra bu değişkenlere atama yapılacaktır.

Sonraki adımda Pickling kullandık. Pickling, bir programda üretilen değişkenleri kaydetmek ve program tekrar çalıştığında okuyabilmek için kullanılan modüldür. Birçok

classifier kullanılması ve zaten var olacak classifier'ların her biri ayrı ayrı işleme tabii tutulması sebebiyle, Pickling ile zamanı kısaltmayı amaçladık.

Daha sonra classifier'ları çağırıp training işlemi yaptık, ardından "accuracy function" ile test ederek doğruluk değerlerini yazdırdık. VoteClassifier sınıfını kullanarak bir ortalama doğruluk değeri elde ettik.

Elde ettiğimiz doğruluk değerleri şu şekilde oldu:

```
Original Naive Bayes Algo accuracy: 74.54819277108435
Most Informative Features
    engrossing = True          pos : neg = 21.6 : 1.0
    mediocre = True           neg : pos = 17.1 : 1.0
    generic = True             neg : pos = 16.4 : 1.0
    loud = True                neg : pos = 16.4 : 1.0
    boring = True              neg : pos = 14.5 : 1.0
    routine = True             neg : pos = 14.4 : 1.0
    inventive = True           pos : neg = 14.3 : 1.0
    flat = True                neg : pos = 14.3 : 1.0
    refreshing = True          pos : neg = 13.0 : 1.0
    wonderful = True           pos : neg = 13.0 : 1.0
    mindless = True           neg : pos = 11.7 : 1.0
    delicate = True            pos : neg = 11.6 : 1.0
    realistic = True           pos : neg = 11.6 : 1.0
    stupid = True              neg : pos = 11.0 : 1.0
    stale = True               neg : pos = 11.0 : 1.0
MNB_classifier Algo accuracy: 71.6867469879518
Bernoulli_NB Algo accuracy: 74.3975903614458
LogisticRegression Algo accuracy: 72.43975903614458
LinearSVC Algo accuracy: 69.12650602409639
Voted_classifier accuracy: 73.49397590361446
```

Son olarak yapılması gereken, modülü Twitter'a bağlayıp real-time plotting yaparak dosyalarımızda kullanmak üzere bir fonksiyon tanımlamaktır.

```
#sentimentmod modülümüz için live_graphing ve connection_twitter'da kullanacağımız fonksiyonumuzu tanımlarız
def sentiment(text):
    feats = find_features(text)
    return voted_classifier.classify(feats),voted_classifier.confidence(feats)
```

Twitter Api bağlantısı sağlamak için gerekli import işlemlerini yaptık. Bunun için kullandığımız modül Tweepy, Twitter bağlantısını sağlayacak olan pakettir. Aynı zamanda "tweerapistuff" adında bir Python dosyası oluşturulur ve bu dosya içerisinde Twitter Developer hesabı üzerinden API'ye bağlanmayı sağlayan Twitter'ın verdiği consumer key, consumer secret, access token, access secret değerleri saklanır.

Tweepy paketinin sağladığı StreamListener sınıfından (class) miras alarak bir sınıf oluşturduk. Bu sınıf içerisinde iki fonksiyon tanımladık. "on_data function"ı kullanarak güvenlik (confidence) değeri yeterli bulunan; 80 değeri üzerinde ya da eşit olanların sonucu bir txt dosyasına yazdırılır. "on_error function"ı errorlar için kullanırız.

Twitter'a bağlanmak için son olarak içinde hangi kelimenin geçmesi istenildiğiyle ilgili bir tanımlama yapılır. Kod dosyası çalıştırarak Twitter'da girilen kelimeyle atılmış tweet'lerin duyarlılığı ve güven değeri ile beraber tweet akışı başlatılır ve yürüterek sürekli akış oluşturan bir sonuç alınır.

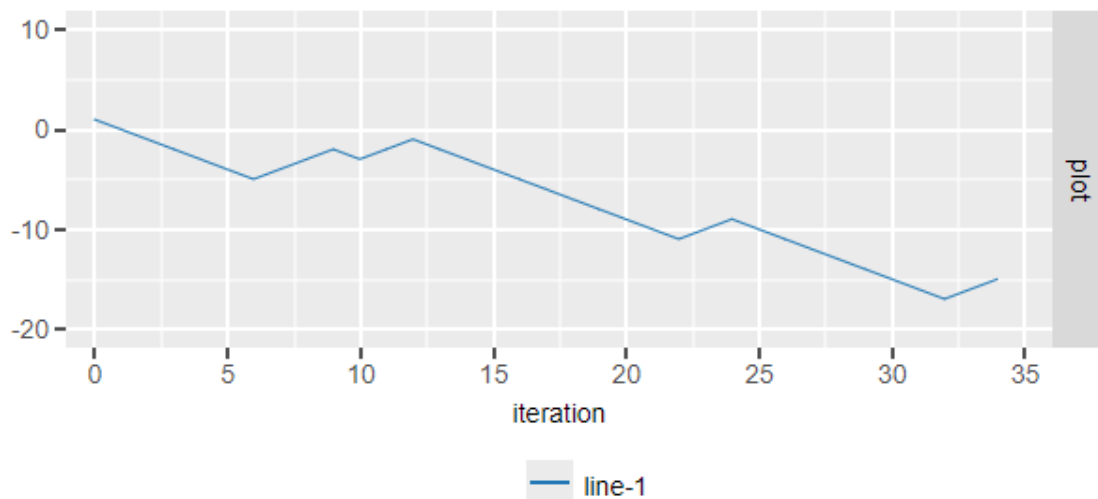
```
try:
    auth = OAuthHandler(ckey, csecret)
    auth.set_access_token(atoken, asecret)

    twitterStream = Stream(auth, listener())

    twitterStream.filter(track=['vaccine'], languages=["en"])
except err.TweepError as e:
    if not is_rate_limit_error(e):
        raise e
    handle_rate_limit_error()
```

Gla... pos 1.0
 RT @Wentzello: If the vaccine is as innocent as publicly portrayed, why shove it down peoples throats via a promo campaign? p
 os 0.6
 RT @CyrilRamaphosa: We have always said that an effective vaccine will be a game-changer. Vaccines offer to the peoples of th
 e world a mean... pos 1.0
 RT @ngyntlinh: genuine question, why are some of y'all more scared of the vaccine than you are of covid LOL pos 1.0
 RT @Stone_SkyNews: .@yarahawari tells @SkyNews why she thinks Israel is obliged to help provide the #COVID19 vaccine to the P
 alestinian Ter... neg 0.8
 RT @rupamurthy1: Thank you all for the warm wishes for my b'day. I really appreciate ur affection 🙏
 Got my SECOND & FINAL dose of COVID19... pos 1.0
 RT @JoeBiden: Getting America vaccinated will be one of the greatest operational challenges we've ever faced, but my administ
 ration will sp... pos 1.0
 RT @straits_times: Sinovac's Covid-19 vaccine efficacy less than 60% in Brazil trial <https://t.co/xZmGETMUHM> neg 1.0
 RT @DrEricDing: FANTASTIC NEWS—A potential vaccine for multiple sclerosis is now within sight on the horizon! And it's an mRN
 A vaccine by B... pos 0.8
 ASTRAZENECA PHARMA: SERUM INSTITUTE TO SELL 100M DOSES OF CO'S VACCINE TO INDIA GOVT FOR 200 RUPEES/DOSE; TO CHARGE MORE AFTE
 R THAT - ANI neg 0.8

Son olarak Twitter'a bağlanma dosyası da çalıştırılarak txt dosyasına gelen sonuçları 1 sn bekleme süresiyle birlikte sırayla animasyon şeklinde plotting yapılır. Örnekte kullanılan 'vaccine' kelimesiyle ilgili atılmış tweetlerde görüşlerin negatif ağırlıkta gittiğini görüyoruz.



4. Tartışma

Projemiz sonucunda negatif ve pozitif duyarlılığı olan tweet’lerde ortak ve sık kullanılan kelimeleri tespit ederek makine öğrenimi yöntemi ile yeni bir veri setindeki tweet’lerin duygu/düşünce analizini yaptık. Elde ettiğimiz verilerle en sık kullanılan pozitif/negatif his belirten kelimeler hangileridir, etiketlerin (hashtag) duygu analizinde ve his belirtmedeki rolü ve etkisi nedir görmüş olduk. Bu tarz bir çalışmanın spesifik konularda toplum araştırması yapmak için, gündemde yer alan bir olay hakkında toplumdaki insanların düşüncelerini görmek için kolayca kullanılabileceğini düşünüyoruz. Farklı bir modül daha kullanarak, belirli bir kelime araştırıldığında, bulunduğumuz zaman içerisinde (her saniye veya dakika gibi) o kelimeyi içeren tweet’ler atılıyorken, bu tweet’lerin ne yöne doğru eğilim gösterdiğini gözlemledik.

5. Kaynakça

- Barnes, N.G.: The Fortune 500 and social media: a longitudinal study of blogging, Twitter and Facebook usage by America’s largest companies (2010).
<https://virtualmarketingofficer.com/wp-content/uploads/2010/02/Fortune-500-and-Social-Media.pdf>
- Retrieved from Society for New Communications Research on 6 March 2011
- Bifet, A., Frank, E.: Sentiment knowledge discovery in Twitter streaming data. In: Pfahringer, B., Holmes, G., Hoffmann, A. (eds.) https://doi.org/10.1007/978-3-642-16184-1_1
- Bottles, K., Sherlock, T.: Who should manage your social media strategy. Physician Executive (2011) www.semanticscholar.org/paper/Who-should-manage-your-social-media-strategy-Bottles-Sherlock/5c18a3d04ff8306461cfb46cb201cddodco3135f?p2df
- Elgamal, M.: Sentiment analysis methodology of Twitter data with an application on Hajj season. Int. J. Eng. Res. Sci. (2016)
https://scholar.google.com/scholar_lookup?title=Sentiment%20analysis%20methodology%20of%20Twitter%20data%20with%20an%20application%20on%20Hajj%20season&author=M.%20Elgamal&journal=Int.%20J.%20Eng.%20Res.%20Sci.%20%28IJOER%29&volume=2&pages=82-87&publication_year=2016
- Gruz, A.: Netlytic: Software for Automated Text and Social Network Analysis (2016).
<http://Netlytic.org>
- Kietzmann, J.H., Hermkens, K., McCarthy, I.P., Silvestre, B.S.: Social media? Get serious! Understanding the functional building blocks of social media. Bus. Horiz. (2011)
<https://www.sciencedirect.com/science/article/abs/pii/S0007681311000061>
- Lovejoy, K., Waters, R.D., Saxton, G.D.: Engaging stakeholders through Twitter: how nonprofit organizations are getting more out of 140 characters or less. Public Relat. Rev. (2012) <https://www.sciencedirect.com/science/article/abs/pii/S0363811112000082>

- Malita, L.: Social media time management tools and tips. Procedia Comput. Sci. (2011) <https://www.sciencedirect.com/science/article/pii/S1877050910004989>
- Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-49247-5_7
- Uzelac, E.: Mastering social media. Research (2011) [Google Scholar](#)
- Zhong, N., Li, Y., Wu, S.T.: Effective pattern discovery for text mining. IEEE Trans. Knowl. Data Eng. (2010)
- Fundamentals of Artificial Intelligence pp 603-649 Natural Language Processing https://link.springer.com/chapter/10.1007/978-81-322-3972-7_19
- Natural language processing: an introduction Journal of the American Medical Informatics Association, Volume 18, Issue 5, September 2011, Pages 544–551 <https://academic.oup.com/jamia/article/18/5/544/829676>

6. Ek Bölüm

6.1. Kaynak Kod

6.2. Video Linki