



T.C.

**SİVAS CUMHURİYET ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**DUYGU ANALİZİ YÖNTEMİYLE ALIŞVERİŞ
SİTELERİNDEKİ YORUMLARIN
PUANLANDIRILMASI**

Ece Nur DOĞAN

LİSANS BİTİRME PROJESİ

**AĞUSTOS-2019
SİVAS**

TEZ BİLDİRİMİ

Bu tezdeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edildiğini ve tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

İmza

Ece Nur DOĞAN

Tarih: 24.08.2019

ÖZET

DUYGU ANALİZİ YÖNTEMİYLE ALIŞVERİŞ SİTELERİNDEKİ YORUMLARIN PUANLANDIRILMASI

Öğrencinin Adı SOYADI

Ece Nur DOĞAN

Danışman:

Dr.Öğr.Üyesi Fırat İSMAİLOĞLU

Jüri:

Dr.Öğr.Üyesi Fırat İSMAİLOĞLU

Anahtar Kelimeler: Doğal Dil İşleme, Metin Madenciliği (text mining), Duygu Analizi (sentiment analysis), Fikir Madenciliği (opinion mining), Bag of Words, Regresyon Analizi

İÇİNDEKİLER

ÖZET	iv
İÇİNDEKİLER	v
KISALTMALAR VE ŞEKİL DİZİNİ.....	viii
1. GİRİŞ	1
1.1. Problemin Tanımı.....	1
1.2. Projenin Amacı	2
1.3. Projede Kullanılacak Programlama Dili.....	2
2. KAYNAK ARAŞTIRMASI	3
3. MATERYAL VE YÖNTEM	4
3.1. Veri Madenciliği	4
3.1.1. Proje İçin Veri Setinin Oluşturulması	5
3.1.1.1. Veri Analizi Kütüphaneleri	6
3.1.1.1.1. Pandas	6
3.1.1.1.1.1. ExcelWriter	7
3.1.1.1.2. Numpy	7
3.1.1.2. Projede Veri Analiz Kütüphanesi Kullanılarak Yapılan İşlemler...	8
3.1.2. Doğal Dil İşleme (NLP)	9
3.1.2.1. Doğal Dil İşleme Kütüphanesi	9
3.1.2.1.1. NLTK	9
3.1.2.2. Metin Madenciliği (Text Mining)	10
3.1.2.2.1. Tokenization	10
3.1.2.2.2. Stop Words.....	11
3.1.2.2.3. Stemming	11
3.1.2.2.4. Bag Of Words	11
3.1.2.3. Zemberek	12
3.1.2.3.1. Metin Madenciliği ve Zemberek' in Projede Kullanılması....	13
3.1.3. Regresyon Analizi	17
3.1.3.1. Regresyon Analizinin Projede Kullanılması	17

4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA	21
5. SONUÇLAR VE ÖNERİLER.....	22
5.1. Sonuçlar	22
5.2. Öneriler	22
KAYNAKLAR	23

KISALTMALAR ve ŞEKİL DİZİNİ

Kısaltmalar

- NLP (Natural Language Process)

Şekil Dizini

Şekil 2.1. Amazon Comprehend ile İlgili Yorumun Duyarlılık Analizi

Şekil 2.2. Amazon Comprehend ile Yorumlardaki Varlıkların Tanımlanması

Şekil 2.3. Amazon Comprehend ile Yorumlardaki Anahtar Sözcüklerin

Listelenmesi

Şekil 3.1. Veri Madenciliği Aşamaları

Şekil 3.2. Kitapyurdu Site Görüntüsü

Şekil 3.3. Yorumların ve Yorum Bilgilerinin Excel Dosyasına Taşınması

Şekil 3.4. Pandas Kütüphanesinin Kurulumu

Şekil 3.5. ExcelWriter Kütüphanesinin Eklenmesi

Şekil 3.6. Excel İşlemlerinin yapılması

Şekil 3.7. Pandas İle İlgili İşlemleri

Şekil 3.8. Veri Setinin Derleyiciye Aktarılması

Şekil 3.9. Belirli Satır Ve Sütundaki Yorumun Alınması

Şekil 3.10. NLTK Downloader penceresinin açılması

Şekil 3.11. NLTK Downloader İle İstenilen NLTK Paketlerinin Kurulması

Şekil 3.12. Stemming

Şekil 3.13. Bag Of Words

Şekil 3.14. Zemberek Çalışma Prensipleri

Şekil 3.15. Zemberek Cümle İşlemleri

Şekil 3.16. Zemberek Ekran Çıktısı

Şekil 3.17. Belirli Sheetteki Yorumların Zemberek Yardımıyla Düzenlenerek

Bag Of Words Düzenine Getirilmesi

Şekil 3.18. Belirli Sheetteki Yorumların Zemberek Yardımıyla Düzenlenerek

Bag Of Words Düzenine Getirilmesi

Şekil 3.19. Kelimenin Yorumda Olup Olmadığının Kontrol Edilmesi

Şekil 3.20. Kelimelerin Olup Olmadığının Kontrol Edildiği Excel Dosyası

Şekil 3.21. Tüm Kelimelerin List Yapısında Sıralanmış Hali

Şekil 3.22. Veri Setindeki Son Yorumun Kelimelere Ayrılması

Şekil 3.23. Veri Setindeki Son Yorumun Kelimelerinin Kökünün Bulunması

Şekil 3.24. Regresyon için kullanılan Kütüphaneler

Şekil 3.25. Kelime Matrisi Oluşturma

Şekil 3.26. Kelime Matrisinin Oluşturulması

Şekil 3.27. Regresyon Analizi

Şekil 3.28. Oluşturulan Kelime Matrisinin Ekran Çıktısı

Şekil 3.29. Yorum Puanı Tahmin Etme

Şekil 3.30. Sonradan Eklenen Yorumun Kelimelere Ayrılması

1) GİRİŞ

Günümüzde gelişen teknoloji, artan bilgi akışı, yoğun üretim ve tüketim, nüfus artışı nedeniyle internet zorunlu bir araç haline gelmiştir. Özellikle değişen yaşam şartları, çalışma hayatı gibi sebepler nedeniyle zamanı verimli kullanmak önemli bir meseledir.

Zamanın değerinin farkında olan tüketicilerin, ihtiyaçlarına cevap veren firmalar tüketicilere internette alışveriş imkânı sağlamıştır. Online alışveriş, zaman tasarrufunun yanında kolay ödeme seçenekleri, tek bir “tık” ile binlerce ürüne istenilen her saatte ulaşılabilmesi gibi olanakları da beraberinde getirmiştir. İnsanların enerjiden ve paradan tasarruf etmesini sağlayan bu yöntemi kullananların sayısı her geçen gün artmaktadır.

Oluşturulan alışveriş sitelerine eklenen yorum ve puanlama alanları sayesinde ürünü daha önce kullanan kişilerin, ürün hakkında bilgiler paylaşması ve gelecekteki alıcılara rehber olması amaçlanmıştır. Verilen her puan, ürünün genel puanını etkilemesi açısından önemlidir. Ürün puanı ne kadar yüksek olursa tüketicilerin o ürünü alma ihtimali de o ölçüde artar. Fakat bazı durumlarda kişilerin ürünlere yaptıkları yorumlarla puanlar birbiriyle çelişebilir. Bu yüzden bu puanlamalar yanıltıcı olabilir.

Projede, kullanıcıların yapmış olduğu yorumlarla verilen puanlar arasında ilişki kurulması, yorumların gereksiz kelimelerden arındırılması, puanların düzenlenerek ürünün genel puanın değiştirilmesi hedeflenmiştir.

1.1. Problemin Tanımı

Bir müşteri internette ürün alırken çoğunlukla o ürünün genel puanına (overall rating) bakar; çünkü tüm yorumların okunması mümkün değildir. Fakat ürünün genel puanı ürünün genel değerini yansıtmayabilir.

Ürünü deneyen müşterilerin yapmış olduğu yorumlar, verdikleri puanla uyuşmuyor olabilir ve ürünün genel puanı kişilerin verdiği puanlardan etkilenir.

Yorumlarda da ürünle ilgili olmayan kelimelere yer verilmiş olabilir; kişilerin kargo şirketi ya da müşteri ilişkileriyle alakalı sorunları veya övgüleri buna örnek verilebilir.

Türkçe çok zengin ve karmaşık bir yapıya sahip olması nedeniyle kullanılan bazı kelimeler hem olumsuz hem de olumlu anlam içeriyor olabilir. Ayrıca kişilerin ürün hakkında karışık düşünceleri vardır. Ürünün bazı yönlerini beğenmiş bazı yönlerini beğenmemiş olabilir. Sonuç olarak ürünün genel puanı pek güvenilir değildir.

1.2. Projenin Amacı

Projede, seçilen siteden alınan kullanıcı yorumları ve kullanıcının bireysel puanı arasındaki ilişkinin saptanması ile ürünün genel puanının daha doğru bir şekilde belirlenmesini amaçlanmaktadır.

Hedeflenen şeyin yapılabilmesi için yorumların sadeleştirilmesi gereksiz ve yanlış kelimelerden arındırılmış olması, yorumlar ile puanlar arasındaki sentiment analysis ve text mining işlemlerinin yapılabilmesi ve puanlamaya aktarılması gerekmektedir. Bu sayede ürünün gerçek değeri daha iyi bir şekilde saptanmış olacaktır.

1.3. Projede Kullanılacak Programlama Dili

Taşınabilir, interaktif, okunabilirliği yüksek ve gerekli olan NLP ve veri analiz kütüphanelerine sahip olduğu için Python programlama dili kullanıldı. Derleyici olarak ise Spyder kullanıldı.

2) KAYNAK ARAŞTIRMASI

Proje ile ilgili çeşitli çalışmalara rastlanmaktadır. Bu çalışmalar arasında en dikkat çekici olanı ise Amozon'un, Amazon Comprehend adlı uygulamasıdır.

Bu uygulamada, makine öğrenmesi tekniklerini kullanılarak; anahtar sözcük çıkarma, duyarlılık analizi, varlık tanıma, konu modelleme ve dil algılama yapılmaktadır.

Analiz edilmek istenen yorum, API Explorer'daki uygun alana yazılır. Bu sayede yorumun olumlu, olumsuz ya da karışık bir duyarlılık için analizi yapılmış olur. Metin alanına 1000 karaktere kadar metin girilebilir. Şekil 2.1. de görüldüğü üzere çıkan sonuçların tamamı olumludur.

The screenshot displays the AWS API Explorer interface for the Amazon Comprehend service. The left sidebar shows the navigation menu with 'API explorer' selected. The main area is titled 'API explorer' and contains a text input field with a sample paragraph: 'This was such a beautiful book. I wasn't even planning any travel when I came across this and just started flipping through the pages. I really like the cover and all the large glossy photographs in this book. John Smith did a wonderful job with the photography. I've found a perfect home for this on my coffee table. I'm planning a trip to Paris and Barcelona soon and I know this will come in handy. In the meantime, it's perfect for assisting this armchair traveler!'. Below the text input, the 'Language' dropdown is set to 'Auto-Detect', and the 'Detected language: English' is shown. The 'Analyze' button is visible. On the right, the results for the 'Language' and 'Sentiment' APIs are displayed. The 'Language' result shows 'en' for English with a confidence of 0.99. The 'Sentiment' result shows 'Positive' with a confidence of 0.99+, 'Neutral' with 0.0, 'Mixed' with 0.0, and 'Negative' with 0.0. The 'Positive' result is highlighted with a red box.

Language code	Language	Confidence
en	English	0.99

Sentiment	Confidence
Positive	0.99+
Neutral	0.0
Mixed	0.0
Negative	0.0

Şekil 2.1. Amazon Comprehend ile İlgili Yorumun Duyarlılık Analizi

The screenshot shows the Amazon Comprehend API explorer interface. The main text input area contains a paragraph about a book. The 'Entity' section on the right displays a table of detected entities, with a red box highlighting the first three rows: John Smith (Person), Paris (Location), and Barcelona (Location). The 'Key phrases' section on the right displays a table of key phrases with their counts and confidence scores.

Entity	Category	Count	Confidence
John Smith	Person	1	0.99+
Paris	Location	1	0.99+
Barcelona	Location	1	0.97

Key phrase	Count	Confidence
such a beautiful book	1	0.97
any travel	1	0.87
the pages	1	0.9
the cover	1	0.98
all the large glossy photographs	1	0.97
this book	1	0.99+
John Smith	1	0.99+
a wonderful job	1	0.99+

Şekil 2.2. Amazon Comprehend ile Yorumlardaki Varlıkların Tanımlanması

Şekil 2.2. de görüldüğü gibi yorum metninin içerisinde üç adet varlık olduğu tespit edilmiştir. Bir tanesinin kişi, diğer ikisinin konum olduğu verilmektedir

The screenshot shows the Amazon Comprehend API explorer interface. The main text input area contains the same paragraph as in the previous image. The 'Key phrases' section on the right displays a table of key phrases with their counts and confidence scores, with a red box highlighting the first ten rows. The 'Language' section on the right displays the detected language as English.

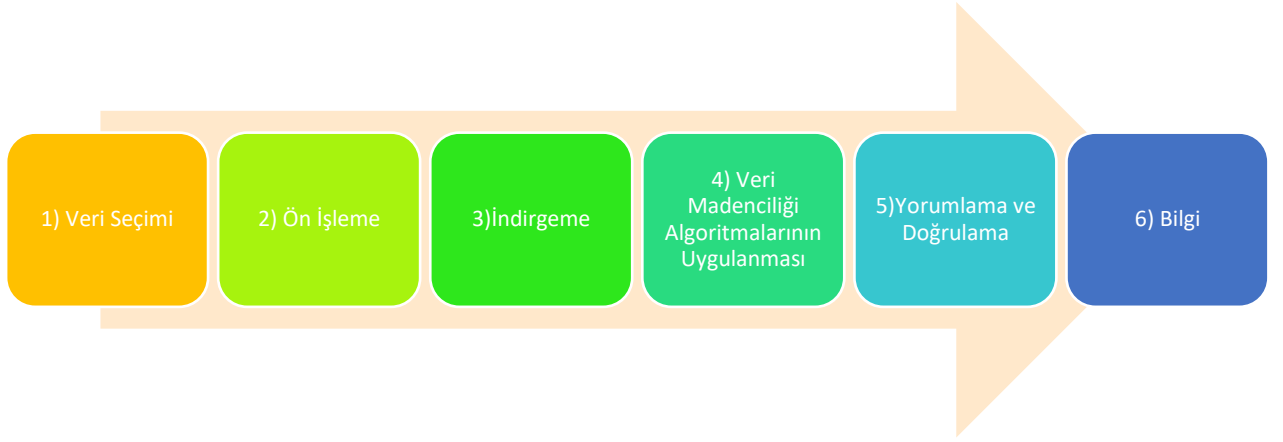
Key phrase	Count	Confidence
such a beautiful book	1	0.97
any travel	1	0.87
the pages	1	0.9
the cover	1	0.98
all the large glossy photographs	1	0.97
this book	1	0.99+
John Smith	1	0.99+
a wonderful job	1	0.99+
the photography	1	0.99+
a perfect home	1	0.95
my coffee table	1	0.99+

Şekil 2.3. Amazon Comprehend ile Yorumlardaki Anahtar Sözcüklerin Listelenmesi

3) MATERYAL VE YÖNTEM

3.1. Veri Madenciliği

Veri madenciliği, büyük veri yığınları arasından otomatik veya yarı otomatik yöntemler ile anlamlı, faydalı bilgiler elde etme işidir.



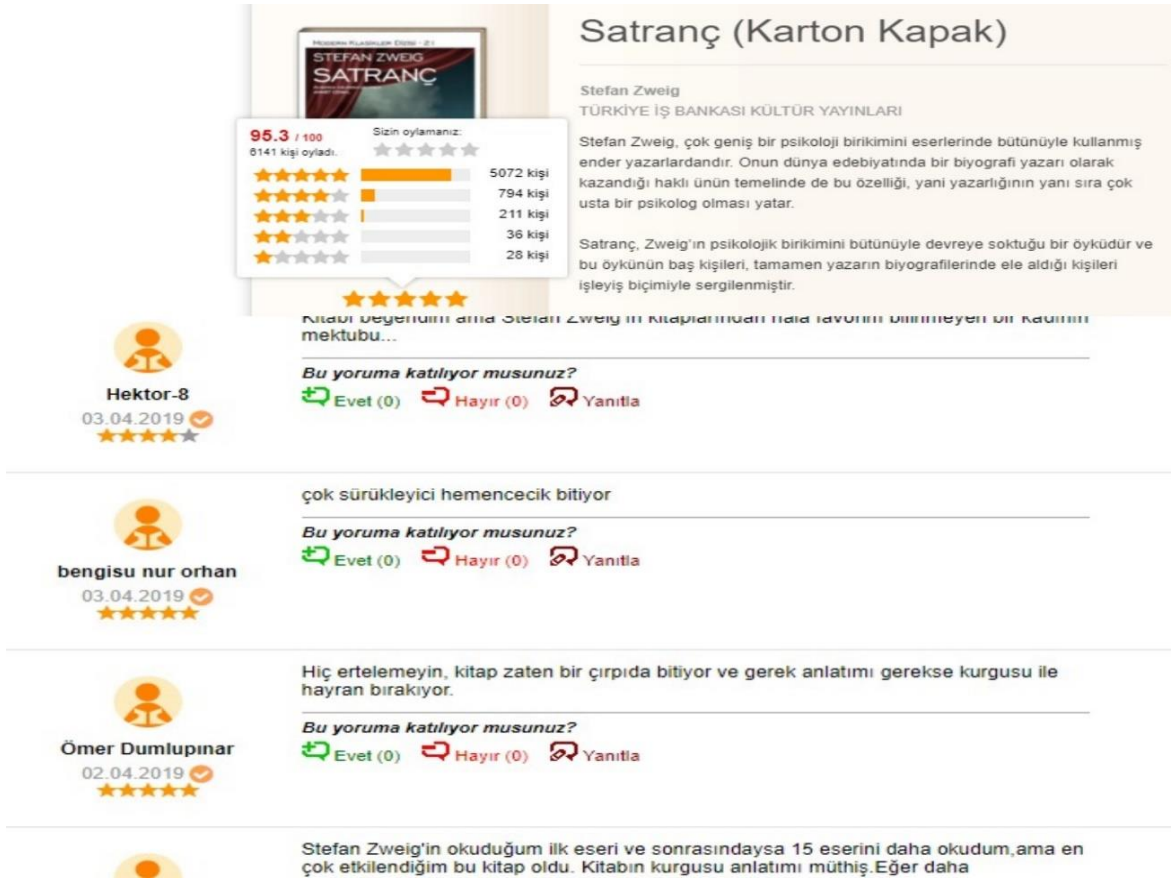
Şekil 3.1. Veri Madenciliği Aşamaları

Şekil 3.1 de görüldüğü gibi veri madenciliği yapılmaya başlanmadan önce veri bazı işlemlerden geçmelidir.

- 1) İlk olarak veri seçiminde, veri iyi analiz edilmiş, problem ile uyuyor olmalıdır.
- 2) Veri elimize geldiğinde, işlem yapmak için uygun olmayabilir. Veri içinde gürültü, eksik bilgi, tutarsızlık vs. gibi sorunlar olabilir. Bunların giderilmesi bu aşamada yapılır.
- 3) İşe yaramayan gereksiz verilerin, veri setinden çıkarılması gerekmektedir. Bu sayede performansın olumsuz etkilenmesinin önüne geçilmiş olunur.
- 4) Çalışmanın amacına göre veri madenciliği algoritmalarının bir veya bir kaç uygulanır.
- 5) Ulaşılan sonuçların doğrulukları kontrol edilir. Eğer varsa daha önceden yapılmış olan çalışmalar incelenir.

3.1.1. Proje İçin Veri Setinin Oluşturulması

Proje için kullanılacak veri setinin kaynağı olan alışveriş sitesi, Kitapyurdu olarak belirlenmiştir.



Şekil 3.2. Kitapyurdu Site Görüntüsü

yorumlar [Uyumluluk Modu] - Excel												
Dosya Giriş Ekle Sayfa Düzeni Formüller Veri Gözden Geçir Görünüm Eklentiler LOAD TEST Team Ne yapmak istediğinizi söyleyin... Oturum Aç Paylaş												
Yapıştır Pano Yazı Tipi Hizalama Sayı Stiller Hücreler Düzenleme												
M1												
	A	B	C	D	E	F	G	H	I	J	K	L
	S.NU	YORUM YAPAN	YORUM	YORUM PUANI	YORUMUN TARİHİ	YORUM EVET	YORUM HAYIR	ÜRÜNÜN GENEL PUANI				
1	1	Büsra Kürkcüoğlu	çok çok çok iyi bir kitap. kısacık ve konusu basit gibi görünen ancak derin bir kitap.	4	26.11.2018	0	0	94.2				
2	2	sociologisttt	Elinizden bırakmadan okunabilecek bir kitap zaten uzun da değil. mükemmel bir kitap	5	24.07.2017	0	1	94.2				
3	3	fserkann	Çok harika bir kitap. Çok beğendim Okumakla kalmadım, akıcılığı ve sürükleyiciliği nedeniyle yaşadım sanki kitabı. Stefan Zweig'in okuduğum ilk kitabıydı. Keşke daha önce okusaydım dedim. Sırada yazanın diğer kitapları var...	5	29.08.2016	2	0	94.2				
4	4	burcu5869	Tek nefeste okudum harika bir kitap	5	19.01.2018	2	0	94.2				
5	5	Aslı Ceren	kesinlikle tavsiye ettiğim bir kitap. Yazarın öyle bir anlatımı var ki sanki olaylar sizin başınızdan geçiyor.	5	27.01.2018	0	0	94.2				
6	6	Melike Döm	Kitabın kalınlığı ile etkileyiciliği ters orantılı mutlaka okumalısınız	5	1.04.2017	1	1	94.2				
7												

Şekil 3.3. Yorumların ve Yorum Bilgilerinin Excel Dosyasına Taşınması

Şekil 3.2.de görünen yorumlar, Şekil 3.3.daki gibi Excel dosyasına taşınmıştır. Dosyadaki her bir sheet bir kitabı, her sheet içindeki her bir satır da bir yorumu ifade etmektedir. Veri seti içinde toplam 30 kitap, her bir kitap içinde yaklaşık 50 yorum bulunmaktadır.

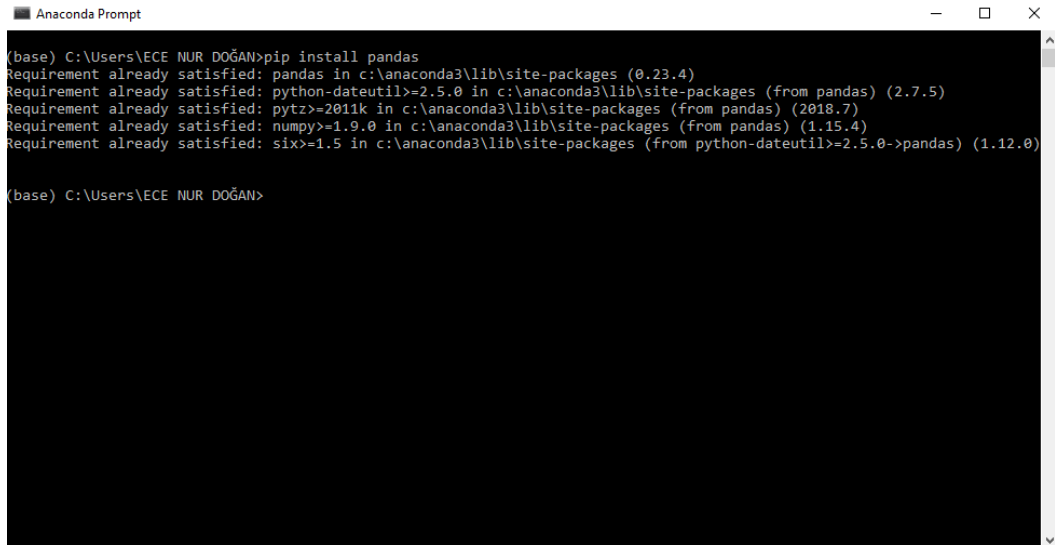
3.1.1.1. Veri Analizi Kütüphaneleri

3.1.1.1.1. Pandas

Pandas, Python programlama dilinin sağladığı kütüphanelerden bir tanesidir. Veri analizi, veri yapıları gibi işlemler için araçlar sağlar. Kullanımı kolaydır.

Pandas, genellikle xlsx, csv ve text dosyalarını açmak, satır sütün işlemleri yapmak için kullanılır. Bu kütüphaneyi, projede veri setini derleyiciye atarak, satır sütün işlemleri yapmak için kullanacağız.

Kullanılan derleyicide pandas kütüphanesinin kurulu olup olmadığına bakmak için komut istemine pip install pandas yazılır.



```
Anaconda Prompt
(base) C:\Users\ECE NUR DOĞAN>pip install pandas
Requirement already satisfied: pandas in c:\anaconda3\lib\site-packages (0.23.4)
Requirement already satisfied: python-dateutil>=2.5.0 in c:\anaconda3\lib\site-packages (from pandas) (2.7.5)
Requirement already satisfied: pytz>=2011k in c:\anaconda3\lib\site-packages (from pandas) (2018.7)
Requirement already satisfied: numpy>=1.9.0 in c:\anaconda3\lib\site-packages (from pandas) (1.15.4)
Requirement already satisfied: six>=1.5 in c:\anaconda3\lib\site-packages (from python-dateutil>=2.5.0->pandas) (1.12.0)

(base) C:\Users\ECE NUR DOĞAN>
```

Şekil 3.4. Pandas Kütüphanesinin Kurulumu

3.1.1.1.1.1. ExcelWriter

Pandas kütüphanesi ile birlikte çalışan kütüphane sayesinde excel dosyası oluşturma, yeni sheet ekleme, satır ve sütunlara veriler ekleme gibi işlemler yapılabilir.

```
1 import pandas as pd
2 import numpy as np
3 from pandas import ExcelWriter
```

Şekil 3.5. ExcelWriter Kütüphanesinin Eklenmesi

```
yorumPuani= df.iloc[:,3] #yorumların olduğu kolon
finalDF = myDF=DataFrame(yorumKelimeDict, columns=list(yorumKelimeDict.keys()))
finalDF.insert(finalDF.shape[1],"Yorum Puani", yorumPuani)
#yorum_kelime_sayisi.xlsx isimli yeni bir excel dosyasının oluşturulması
file = ExcelWriter('yorum_kelime_sayisi.xlsx')
finalDF.to_excel(file,'Sheet1',index=False)
#Oluşturulan dosyanın kaydedilmesi
file.save()
```

Şekil 3.6.Excel İşlemlerinin yapılması

3.1.1.1.2. Numpy

Bilimsel hesaplama ve dönüşüm işlemlerinde kullanılır. Matematiksel işlemleri hızlı yaptığı için tercih edilmektedir. Projede bu kütüphanenin kullanılma amacı, dizi ve matrislerle işlem yapılacak olmasıdır.

Numpy kütüphanesi de aynı pandas gibi komut istemine pip install numpy yazılmasıyla kullanılabilir hale gelir.

3.1.2. Projede Veri Analiz Kütüphanesi Kullanılarak Yapılan İşlemler

```
1 import pandas as pd
2 #
3 #Exceldeki dosyayı belli bir sheetindeki tüm verileri okuma
4 df=pd.read_excel('yorumlar.xlsx',sheetname='SATRANÇ Stefan Zweig 28004')
5 #
6 # İlgili satır ve sütundaki değeri alma
7 kelime=df.loc[35][2]
8 #
9 # Tüm harfleri küçük harf haline getirme
10 kelimeler=kelime.lower()
11 #
```

Şekil 3.7. Pandas ile ilgili işlemleri

df - DataFrame

Index	S.NU	YORUM YAPAN	YORUM	YORUM PUANI	/ORUMU
0	1	Büsra Kürkcüoğlu	çok çok çok iyi bir kitap...	4	2018-11-00:00:00
1	2	sociologisttt	Elinizden bırakmadan o...	5	2017-07-00:00:00
2	3	fserkann	Çok harika bir kitap. Ç...	5	2016-08-00:00:00
3	4	burcu5869	Tek nefeste okudum harik...	5	2018-01-00:00:00
4	5	Aslı Ceren	kesinlikle tavsiye etti...	5	2018-01-00:00:00
5	6	Melike Döm	Kitabın kalınlığı il...	5	2017-04-00:00:00
6	7	caneda_18	Tarihsel bağlamda Naz...	5	2016-09-00:00:00
7	8	Sevda Demirbaş	yi kurgulanmış ...	5	2017-01-00:00:00
8	9	suleyman0399	Kısa olsa da güzel bir es...	5	2016-09-00:00:00
9	10	seri28	Satranca bakış açımı ...	5	2018-07-00:00:00
10	11	vexilloidd	Tek seferde okunan harik...	5	2016-02-00:00:00
11	12	shawshak1903	kesinlikle okunması ger...	5	2017-02-00:00:00
12	13	Esin Senbik	kesinlikle kusursuz, ke...	5	2018-08-00:00:00
13	14	memnosoftt	Çok ama çok	5	2018-01-00:00:00

Format Resize Background color Column min/max Save and Close Close

Şekil 3.8. Veri Setinin Derleyiciye Aktarılması

Text editor - kelime

Çok etkileyici...Tüm satrançseverlerin okuması gerekir...

Save and Close Close

Şekil 3.9. Belirli Satır Ve Sütundaki Yorumun Alınması

3.1.2. Doğal Dil İşleme (NLP)

Herkesin bildiği üzere, doğal dil insanların iletişim kurması için var olan en temel özelliklerden biridir. Doğduğumuz andan itibaren duyduğumuz dili öğrenmek her ne kadar kolay gibi görünse de aslında yeni bir dilin öğrenilmesi kolay bir süreç değildir. Her dilin kendine has dil bilgisi kuralları, telaffuzu ve alfabesi vardır.

Doğal Dil İşleme, doğal dillerin kurallı yapısının çözümlenerek anlaşılması veya yeniden üretilmesi amacını taşır. Yapay zekâ ve dil biliminin ortak kullanımları ile oluşturulmuştur. Bilgisayar ortamında, seslerin yazıya geçmesi, dil tanımlaması, metinlerin işlenmesi gibi çok geniş çalışma alanları vardır.

Doğal dil işleme, sabit algoritmalar içermez ve verilen bir problemin anlaşılıp, çözümün bulunup en uygun yazılımın geliştirilmesi gerekir. Bu aşamada doğal dil işleme teknikleri ile yazım hatalarının bulunması ve bu hataların en yüksek başarı ile düzeltilmesi amaçlanmıştır. Buna göre;

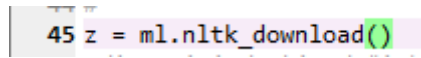
- Elimizdeki veri seti ile metinlerdeki kelimelerin yazım hatalarının denetlenmesi,
- Yanlış yazılan kelimeler için sözcük önerilmesi,
- Bulunan yazım hatalarının düzenlenmesi üzerinde çalışılmıştır.

3.1.2.1. Doğal Dil İşleme Kütüphanesi

3.1.2.1.1. NLTK

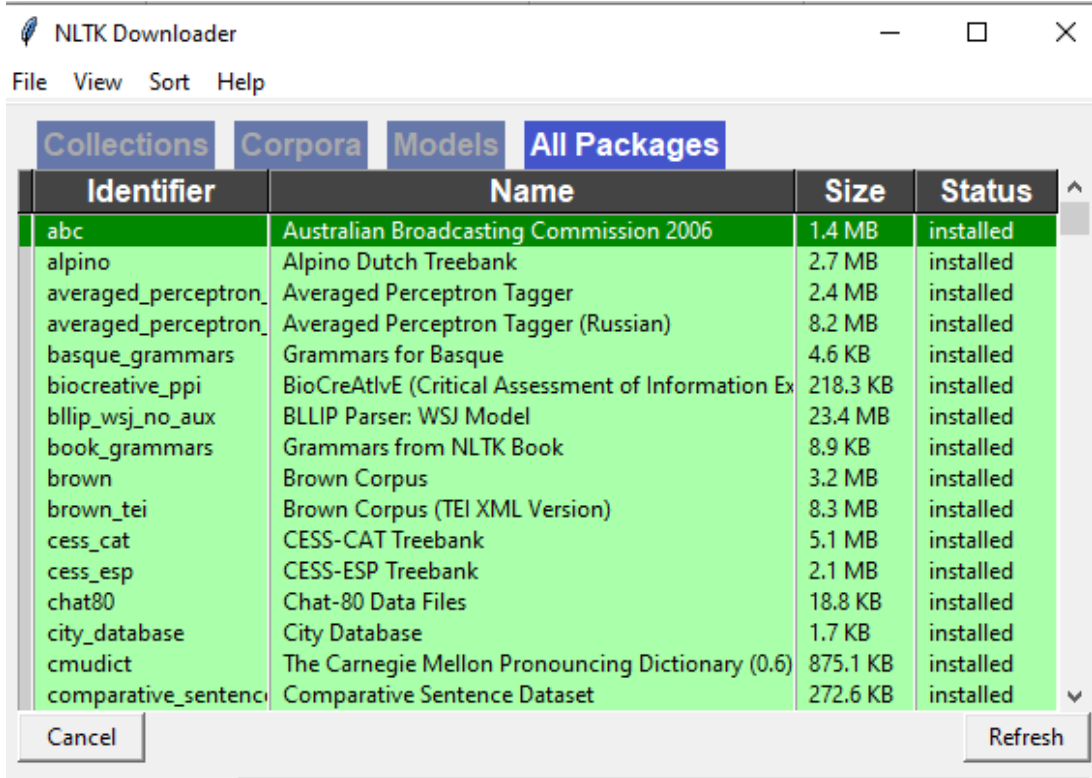
Metin içerikli dokümanlar üzerinde dil işleme yapmayı kolaylaştıran bir kütüphane olan NLTK (natural language toolkit), özellikle yapay öğrenim algoritmaları, işlemeden dil metinleri üzerinde ön işleme yapmak için faydalıdır. Bu algoritmalar, çoğu dokümanı "kelime çuvalı" (bag-of-words) olarak temsil eder ve bu tür işlemler NLTK ile çok hızlı şekilde yapılabilir.

Kurmak için, command açılarak pip install nltk yazılmalıdır. Daha sonra nltk derleyicide import edilmelidir.



```
45 z = ml.nltk_download()
```

Şekil 3.10. NLTK Downloader penceresinin açılması



Şekil 3.11. NLTK Downloader ile İstenilen NLTK Paketlerinin Kurulması

Şekil 3.10.de görüldüğü şekilde bağlantı yazılarak .py dosyası run edildiğinde, Şekil 3.11. NLTK downloader penceresi açılmaktadır. Bu sayede eksik olan NLTK paketleri yüklenebilmektedir.

3.1.2.2. Metin Madenciliği (Text Mining)

Metni veri kaynağı olarak kabul eden veri madenciliği çalışmasıdır. Metin üzerinden yapılandırılmış veri elde etmeyi amaçlar. Metin madenciliği ile metinlerin sınıflandırılması (classification), kümelenmesi (clustering), metinlerden konu çıkarılması (entity extraction), sınıf taneciklerinin üretilmesi (production of granular taxonomy), duygusal analiz (sentimental analysis), metin özetleme (document summarization), ve varlık ilişki modellemesi (entity relationship modelling) gibi çalışmalar yapılabilir.

3.1.2.2.1. Tokenization

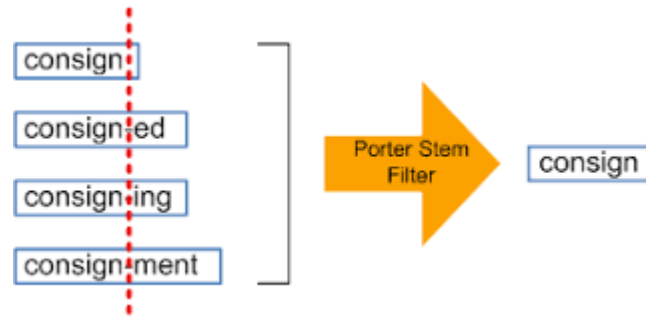
Metnin boşluk, noktalama işaretleri vb. istenen özelliklere göre parçalara ayrılmasıdır. Aynı zamanda bu işlem yapılırken, ayrılan parçalar bir array yapısında tutulur.

3.1.2.2.2. Stop Words

Metinde geçen gereksiz kelimelerin veri setinden çıkarılmasıdır.

3.1.2.2.3. Stemming

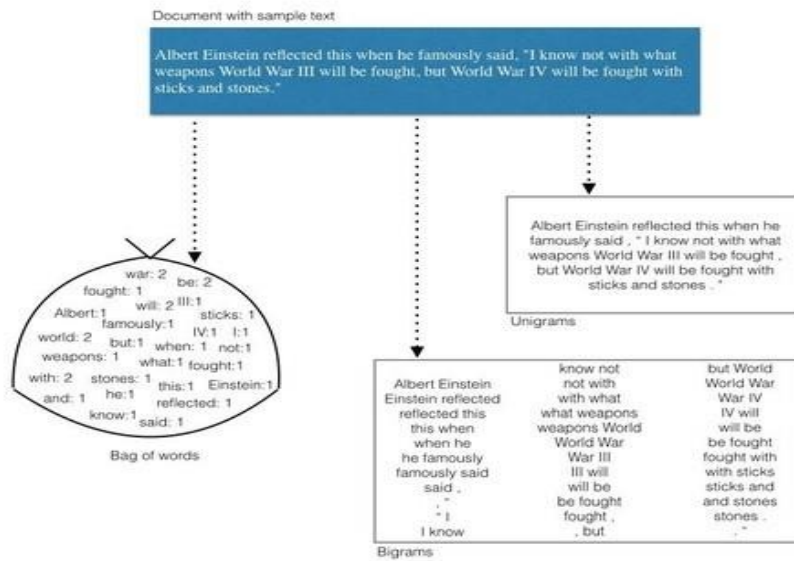
Metinde geçen aynı köke sahip kelimelerin eklerden arındırılarak aynı kelime olarak kabul edilmesidir.



Şekil 3.12. Stemming

3.1.2.2.4. Bag Of Words

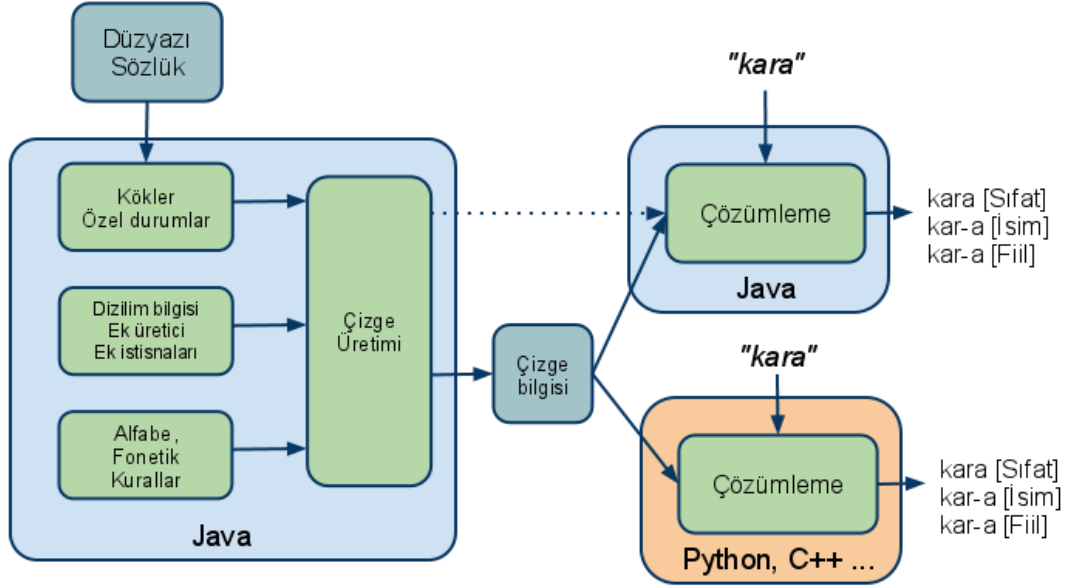
Veri setinde aynı kelimenin kaç kere geçtiği, kelimenin olup olmadığı kontrol edilir ve bu veriler bir arada tutulur.



Şekil 3.13. Bag Of Words

3.1.2.2. Zemberek

Zemberek, Türkçe için oluşturulmuş doğal dil işleme aracıdır. İlk başlarda Java için üretilmiş olsa da sonradan Python'a uyarlanmıştır.



3.14. Zemberek Çalışma Prensibi

Zemberek kullanılarak; cümlede geçen köklerin bulunması, gereksiz eklerin çıkarılması, verilen kelime ile ilgili kelime önerme ve kelime heceleme gibi işlemler yapılabilir.

```
0
7 corpus = "merhaba, dünya!"
8
9
10 ## cümleyip gereksiz eklerden boşluklardan ayırır temizler sunar
11 b = ml.ZemberekTool(zemberek_api).cumleyi_parcalara_ayir(corpus)
12 print(b)
13
14 ## cümlede geçen kökleri bulur
15 c = ml.ZemberekTool(zemberek_api).metinde_gecen_kokleri_bul(corpus)
16 print(c)
17
```

3.15. Zemberek Cümle İşlemleri

IPython 7.2.0 -- An enhanced Interactive Python.

```
In [1]: runfile('C:/Users/ECE NUR DOĞAN/.spyder-py3/bitirme/
zemberek_ornek.py')
['merhaba', 'dünyaaa']
['merhaba', 'dünyaa']
```

3.16. Zemberek Ekran Çıktısı

Zemberek kullanılabilmesi için öncelikle, JPyype modülünün kurulması gerekmektedir. Bu işlem de command açılarak pip install JPyype1 yazılarak sağlanır. Daha sonra Zemberek' in tüm verilerinin son sürümünün indirip ilgili klasöre yerleştirilmesi gerekmektedir.

3.1.2.3.1. Metin Madenciliği ve Zemberek' in Projede Kullanılması

```
1 import pandas as pd
2 import zemberek_python
3 import nltk
4 from pandas import DataFrame
5 from pandas import ExcelWriter
6 from zemberek_python import main_libs as ml
7
8 #
9 #Exceldeki dosyayı belli bir sheetindeki tüm verileri okuma
10 df=pd.read_excel('yorumlar.xlsx',sheetname='SATRANÇ Stefan Zweig 28004')
11 #
12 #Zemberek ile ilgili dosya uygun klasöre yerleştirildi.
13 zemberek_api = ml.zemberek_api(libjvmpath="C:\\Program Files\\Java\\jre1.8.0_152\\bin\\server",
14                                zemberekJarpath="./zemberek_python/zemberek-tum-2.0.jar").zemberek()
15
16
17 yorumSayisi=df.shape[0]
18 yorumKelimeDict = {} #bu matrix olusturmaya calistigimiz dictionary
19
20 yorum = df.loc[0][2]
21 yorum = yorum.lower() # Tüm harfleri küçük harf haline getirme
22 yorumKelimeList = ml.ZemberekTool(zemberek_api).cumleyi_parcalara_ayir(yorum)
23 yorumKelimeListKok= ml.ZemberekTool(zemberek_api).metinde_gecen_kokleri_bul(yorumKelimeList)
24
25 for kelime in yorumKelimeListKok:
26     yorumKelimeDict.update({kelime:[1]})
27
```

3.17. Belirli Sheetteki Yorumların Zemberek Yardımıyla Düzenlenerek Bag Of Words Düzenine Getirilmesi

```

28 for i in range(1,yorumSayisi): #her bir yorum için
29     for kel in yorumKelimeDict:
30         gorulmeListesi = yorumKelimeDict[kel]
31         gorulmeListesi.append(0)
32         yorumKelimeDict[kel] = gorulmeListesi
33
34     yorum = df.loc[i][2]
35     yorum = yorum.lower() # Tüm harfleri küçük harf haline getirme
36     yorumKelimeList = ml.ZemberekTool(zemberek_api).cumleyi_parcalara_ayir(yorum)
37     yorumKelimeListKok= ml.ZemberekTool(zemberek_api).metinde_gecen_kokleri_bul(yorumKelimeList)
38     #kelimeSayisi = len(yorumKelimeListKok)
39
40     for kelime in yorumKelimeListKok: #her bir kelime için
41         suAnaDekGorulmusKelimeler = list(yorumKelimeDict.keys())
42         if kelime in suAnaDekGorulmusKelimeler: #eger kelime daha once gorulmusse
43             gorulmeListesi = yorumKelimeDict[kelime]
44             gorulmeListesi[-1]= gorulmeListesi[-1] +1 #kelimenin gorulme sayisini 1 artir
45             yorumKelimeDict[kelime] = gorulmeListesi
46         else: #eger kelime daha once gorulmemisse
47             yeniGorulmeListesi = [0]*(i+1)
48             yeniGorulmeListesi[-1] = 1
49             yorumKelimeDict.update({kelime:yeniGorulmeListesi})
50
51 yorumPuani= df.iloc[:,3] #yorumların olduğu kolon
52 finalDF = myDF=DataFrame(yorumKelimeDict, columns=list(yorumKelimeDict.keys()))
53 finalDF.insert(finalDF.shape[1],"Yorum Puani", yorumPuani)
54 #yorum_kelime_sayisi.xlsx isimli yeni bir excel dosyasının oluşturulması
55 file = ExcelWriter('yorum_kelime_sayisi.xlsx')
56 finalDF.to_excel(file,'Sheet1',index=False)
57 #Oluşturulan dosyanın kaydedilmesi
58 file.save()

```

3.18. Belirli Sheetteki Yorumların Zemberek Yardımıyla Düzenlenerek Bag Of Words Düzenine Getirilmesi

Yukarıdaki kodlar kullanılarak; ilgili sheette bulunan kitap hakkında yapılmış olan yorumlar, Zemberek kullanılarak kelimelere parçalanmıştır. Kelimenin diğer yorumlarda olup olmadığı kontrol edilmiştir. Eğer yoksa yeni kelime, kelime satırının sonuna eklenmiş, varsa ilgili yorum satırı ve kelimenin kesiştiği alanın değeri 1 olarak değiştirilmiştir. Her yorum bir satırla ifade edilmiş, aynı zamanda kod, otomatik olarak Excel dosyası oluşturup tüm kelimelerin düzenli bir şekilde aktarımı yapılmıştır. Excel dosyası şu şekildedir;

finalIDF - DataFrame

Index	i	bir	kitap	kısacık	kon
0	1	1	1	1	1
1	0	1	1	0	0
2	0	1	4	0	0
3	0	1	1	0	0
4	0	1	1	0	0
5	0	0	1	0	0
6	0	1	1	0	0
7	0	0	1	0	0
8	0	1	0	0	0
9	0	0	1	0	0
10	0	1	1	0	0
11	0	1	1	0	0
12	0	0	1	0	0
13	0	0	0	0	0

3.19. Kelimenin Yorumda Olup Olmadığının Kontrol Edilmesi

L1

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	
1	i	bir	kitap	kısacık	konus	basit	görüne	ancak	der	el	bırakma	kunabilece	zate	uz	değil	mükemmel	harik	beğ	okumak	kalr
2	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
3	0	1	1	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0	0	0
4	0	1	4	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
5	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
6	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
13	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	1	2	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
20	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
22	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
23	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Sheet1

3.20. Kelimelerin Olup Olmadığının Kontrol Edildiği Excel Dosyası

suAnaDekGorulmusKelimeler - List (321 elements)

Index	Type	Size	Value
265	str	1	dair
266	str	1	anlatım
267	str	1	net
268	str	1	olarak
269	str	1	görebiliyor
270	str	1	savaş
271	str	1	yarattık
272	str	1	tahribat
273	str	1	dünya
274	str	1	er
275	str	1	ayrılmas

3.21. Tüm Kelimelerin List Yapısında Sıralanmış Hali

yorumKelimeList - List (6 elements)

Index	Type	Size	Value
0	str	1	bir
1	str	1	çirpıda
2	str	1	okuyacağınız
3	str	1	muhteşem
4	str	1	zweig
5	str	1	eseri

3.22. Veri Setindeki Son Yorumun Kelimelere Ayrılması

yorumKelimeListKok - List (6 elements)

Index	Type	Size	Value
0	str	1	bir
1	str	1	çırpı
2	str	1	okuyacak
3	str	1	muhteşe
4	str	1	zweigi
5	str	1	eser

3.23. Veri Setindeki Son Yorumun Kelimelerinin Kökünün Bulunması

3.1.3. Regresyon Analizi

İki değişken arasındaki ilişkiyi ölçmek için kullanılan analiz yöntemidir. Amaç iki değişken arasındaki bağlantıya en yakın doğrusal sonucu elde etmektir.

3.1.3.1. Regresyon Analizinin Projede Kullanılması

```
1 import pandas as pd
2 import numpy as np
3 import nltk
4 from pandas import DataFrame
5 from pandas import ExcelWriter
6 import zemberek_python
7 from zemberek_python import main_libs as ml
8 from sklearn.linear_model import LinearRegression
```

3.24. Regresyon için kullanılan Kütüphaneler


```

17
18 yorumSayisi=df.shape[0]
19 yorumKelimeDict = {} #bu matrix olusturmaya calistigimiz dictionary
20
21 yorum = df.loc[0][2]
22 yorum = yorum.lower() # Tüm harfleri küçük harf haline getirme
23 yorumKelimeList = ml.ZemberekTool(zemberek_api).cumleyi_parcalara_ayir(yorum)
24 #yorumKelimeListKok= ml.ZemberekTool(zemberek_api).metinde_gecen_kokleri_bul(yorumKelimeList)
25
26 for kelime in yorumKelimeList:
27     yorumKelimeDict.update({kelime:[1]})
28
29 for i in range(1,yorumSayisi): #her bir yorum için
30     #herbir yorumda
31     for kel in yorumKelimeDict:
32         gorulmeListesi = yorumKelimeDict[kel]
33         gorulmeListesi.append(0) #listenin sonuna 0 ekle
34         yorumKelimeDict[kel] = gorulmeListesi
35
36 yorum = df.loc[i][2]
37 yorum = yorum.lower() # Tüm harfleri küçük harf haline getirme
38 yorumKelimeList = ml.ZemberekTool(zemberek_api).cumleyi_parcalara_ayir(yorum)
39 #yorumKelimeListKok= ml.ZemberekTool(zemberek_api).metinde_gecen_kokleri_bul(yorumKelimeList)
40 #kelimeSayisi = len(yorumKelimeListKok)
41
42 for kelime in yorumKelimeList: #her bir kelime için
43     suAnaDekGorulmusKelimeler = list(yorumKelimeDict.keys())
44     if kelime in suAnaDekGorulmusKelimeler: #eger kelime daha once gorulmusse
45         gorulmeListesi = yorumKelimeDict[kelime]
46         gorulmeListesi[-1]= gorulmeListesi[-1] +1 #kelimenin gorulme sayisini 1 artir
47         yorumKelimeDict[kelime] = gorulmeListesi
48     else: #eger kelime daha once gorulmemisse
49         yeniGorulmeListesi = [0]*(i+1)
50         yeniGorulmeListesi[-1] = 1
51         yorumKelimeDict.update({kelime:yeniGorulmeListesi})

```

3.25. Kelime Matrisi Oluşturma

```

53 yorumPuanı= df.iloc[:,3] #yorumların olduğu colon
54 finalDF =DataFrame(yorumKelimeDict, columns=list(yorumKelimeDict.keys()))
55 #finalDF.insert(finalDF.shape[1],"Yorum Puanı", yorumPuanı)
56 #file = ExcelWriter('sth2.xlsx')
57 #finalDF.to_excel(file, 'Sheet1', index=False)
58 #file.save()
59

```

3.26. Kelime Matrisinin Oluşturulması

```

60 ###Regresyon bolumu#####
61 finalDFnp=np.array(finalDF)
62 yorumPuanınp=np.array(yorumPuanı)
63 model = LinearRegression().fit(finalDFnp, yorumPuanınp)
64 #####
65
66 #####yorumu al#####
67 kisiYorumu= input('Kitap hakkındaki yorumlarınızı yazınız: \n') #amacımız bu yorumu binary vektor hal
68 kisiYorumu = kisiYorumu.lower()
69 kisiYorumKelimeList = ml.ZemberekTool(zemberek_api).cumleyi_parcalara_ayir(kisiYorumu)
70 #yorum kelime list'teki her bir kelimenin daha once gorulup gorulmedigine bakalım
71 suAnaDekGorulmusKelimeler = list(yorumKelimeDict.keys())
72 suAnaDekGorulmusToplamKelimeSayisi=len(suAnaDekGorulmusKelimeler)
73 kisiYorumuVektor=[0]*suAnaDekGorulmusToplamKelimeSayisi
74
75
76 for kelime in kisiYorumKelimeList: #her bir kelime için
77     if kelime in suAnaDekGorulmusKelimeler: #eger kelime daha once gorulmusse
78         kacinciKelime=suAnaDekGorulmusKelimeler.index(kelime)
79         kisiYorumuVektor[kacinciKelime]=1
80
81 kisiYorumuVektor = np.array(kisiYorumuVektor)
82 ##### Tahmin#####
83 tahminiSkor =np.dot(kisiYorumuVektor,model.coef_)+ model.intercept_
84
85 print('Yazdığınız yoruma göre kitaba vereceğiniz tahmini puan:', round(tahminiSkor,2))
86

```

3.27. Regresyon Analizi

yorumKelimeDict - Dictionary (450 elements)

Key	Type	Size	Value
1980	list	50	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...]
1984	list	50	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...]
abartılacak	list	50	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...]
abartıldığı	list	50	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...]
abartılmış	list	50	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...]
abartılı	list	50	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...]
aklıma	list	50	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...]
aklınız	list	50	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...]
akıcı	list	50	[1, 0, 0, 0, 0, 0, 0, 0, 0, 1, ...]
akışın	list	50	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...]
alakası	list	50	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...]

Save and Close Close

3.28. Oluşturulan Kelime Matrisinin Ekran Çıktısı

```
IPython console
Console 1/A x

Kitap hakkındaki yorumlarınızı yazınız:
Kitabı okumadan önce yazıldığı zamana ve kitaba ilişkin bilgi veren
önsözü ile yazarın kısa bir özgeçmişini okumayı tavsiye ediyorum.
Kitabı anlayabilmem de oldukça faydalı oldu. Kitap genel olarak akıcı
bir anlatıma sahip, ancak çevirinden mi kaynaklı bilemiyorum zaman
zaman betimlemeleri anlamakta zorlandım.
Yazdığınız yoruma göre kitaba vereceğiniz tahmini puan: 4.5

In [2]:
```

3.29. Yorum Puanı Tahmin Etme

Kod çalıştırıldığında, çıktı olarak regresyon analizi yapılan kitap hakkında bir yorum yazılması istenir. Yazılan yoruma, daha önceden yazılmış yorumlardan elde edilen regresyon sonucundan yararlanarak bir puan elde edilir. Kısacası yapılan yoruma göre verilmesi gereken puan bu şekilde olmalıdır.

kisiYorumKelimeList - List (36 elements)

Index	Type	Size	Value
0	str	1	kitabı
1	str	1	okumadan
2	str	1	önce
3	str	1	yazıldığı
4	str	1	zamana
5	str	1	kitaba
6	str	1	ilişkin
7	str	1	bilgi
8	str	1	veren
9	str	1	önsözü
10	str	1	yazarın

Save and Close Close

3.30. Sonradan Eklenen Yorumun Kelimelere Ayrılması

4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA

Gelişen teknoloji sayesinde internetten alışveriş yapanların sayısı her geçen gün artmaktadır. Alışveriş yapanların aldıkları ürünlere yorumları ve verdikleri puanlarda aynı ölçüde artmaktadır. Yorumların değerlendirilerek ürünlerin puanlandırılması konusunda yapılan farklı çalışmalar bulunmaktadır. Ürünün genel puanının artması hem ürünü satın alacak olan müşteriler için hem de satıcılar için büyük önem taşımaktadır.

Bitirme projesi kapsamında gerçekleştirmiş olduğum bu çalışmada, veri analiz teknikleri kullanarak; veri setinin oluşturulması, veri setinin derleyiciye taşınması, kütüphaneler kullanılarak yeni Excel dosyası oluşturulması, gerekli satır sütun işlemlerinin yapılması gibi aşamalar gerçekleştirildi.

Doğal dil işlemenin bir alt dalı olan, metin madenciliği kullanılarak; Bag of Words çalışması yapıp kelimler yeni bir Excel dosyasına aktarılmıştır.

Türkçe dili kullanılarak doğal dil işlemeyi mümkün kılan zemberek sayesinde; yorumların kelimelerine parçalanması, kelimelerin köklerinin bulunması gibi işlemler yapılabilmektedir.

Sayılan tüm işlemlerden elde edilen sonuçlar birleştirilerek yapılan regresyon analizi sayesinde; kelime matrislerinin oluşturulması ve yapılan yeni bir yorum için tahmini bir puan üretilmesi gerçekleştirilmiştir.

5. SONUÇLAR VE ÖNERİLER

5.1 Sonuçlar

Yapılan çalışmalar sonucunda, kullanılan veri sertine uygun olan tahmini yorum puanları alındığı görülmüştür.

5.2 Öneriler

Projenin daha kullanışlı olabilmesi için, GUI (Graphical User Interface) oluşturulabilir. Bu sayede kullanıcıların programı daha rahat kullanabilmesi sağlanmış olur.

KAYNAKLAR

- [1] “Metindeki duyarlılığı analiz etme”, 2019. [Online]. Available: <https://aws.amazon.com/tr/getting-started/tutorials/analyze-sentiment-comprehend/#> [Accessed: 26-Aug-2019].
- [2] M. Gocenoglu, “Veri Madenciliği Aşamaları”,2014. [Online]. Available: <http://mgocenoglu.blogspot.com/2014/06/veri-madenciligi-asamalar.html> [Accessed: 26-Aug-2019].
- [3] Burak Bayramli,” Dil Isleme, Python” 2016. [Online]. Available: <http://sayilarvekuramlar.blogspot.com/2016/04/dil-isleme-python-nltk.html> [Accessed: 26-Aug-2019].
- [4] “Zemberek-3 Durum” 2011. [Online]. Available: <http://zembereknlp.blogspot.com/2011/04/zemberek-3-durum.html> [Accessed: 26-Aug-2019].
- [5] Berkay Ozturk,” Zemberek Python ” 2018. [Online]. Available: <https://github.com/ozturkberkay/Zemberek-Python-Examples> [Accessed: 26-Aug-2019].

