



BİTİRME PROJESİ

Haftalık Rapor – 25.03.2022

25 MART 2022

KIRIKKALE ÜNİVERSİTESİ – BİLGİSAYAR MÜHENDİSLİĞİ İÖ

AHMET MUNGAN – 160255081

İÇİNDEKİLER

ÖZET.....	2
YEMEKSEPETİ VERİ TOPLAMA İŞLEMLERİ	3
Firmadan Alınan İzin	3
Program Kodu İle Veri Çekme	3
REFERANS VE KAYNAKÇA	8
EKLER.....	9

ÖZET

Türkiye'nin öncü ve ileri gelen yemek siparişi platformu Yemeksepeti'nin verilere erişim ve kullanım konusunda gerekli izinler alınmıştır. Bu izinlerin kapsamı, firmanın belirttiği ölçüde kullanılıp veri seti oluşturma işlemlerini program kodu vasıtasıyla gerçekleştirmeye çalışılmıştır. Henüz veri seti oluşturma kısımları eksik kalmış olsa da bu rapor özelinde ilk ve en önemli adımlar atılmıştır.

YEMEKSEPETİ VERİ TOPLAMA İŞLEMLERİ

Firmadan Alınan İzin

Veri toplamak için geçen hafta öğrenilen BeautifulSoup kütüphanesi ile Yemeksepeti verileri toplanamıştır. Bad request gibi hatalar ile karşılaşılır. Fakat Yemeksepeti bilgi teknolojileri grubu ile mailleşme sonucu veri toplamak için özel izinler alınmış ve bu izne dair teknik detaylar paylaşılmıştır.

Merhaba.

Sevgili Ahmet, isteğini değerlendirdik. İsteği üzerine veritabanı paylaşımı maalesef kurumsal gerekçeler sebebi ile paylaşılamaz. Fakat senin sadece bayi yorumlarına ihtiyacına Yemeksepeti Ailesi olarak sessiz kalmadık. Sana verilecek özel Header Name ile Türkiye ve Kıbrıs'taki bayilere, kullanıcılar tarafından yapılmış son 5 aylık bütün yorumlara erişim imkanı sunuyoruz. Bu projeyi eğitimin kapsamında sana en faydalı olacak şekilde tamamlayıp, rapor olarak bizimle paylaşman tarafınız. Bir sonraki mail dizisinde Header Name ve bazı özel bilgileri bulabilirsiniz. Yemeksepeti'ne katkılarından ötürü sonsuz teşekkürler!

Yemeksepeti IT Team

Mail 1

Mail 1'de görüleceği üzere, gerekli izin alınmıştır. Bu sayede veri toplama işlemleri gerçekleştirilecektir.

Program Kodu İle Veri Çekme

Code 1

```
>>> from bs4 import BeautifulSoup
>>> import requests
>>> link = input("Restoran linki giriniz: ")
Restoran linki giriniz:
https://www.yemeksepeti.com/pizzala-kecioren-baglum-karakaya-mah-ankara
>>> header = {"Supervisory-YS" : "???" } #hidden content for security
>>> istek = requests.get(link, headers = header)
>>> soup = BeautifulSoup(istek.content, "lxml")
>>> degerler = soup.find_all("div", attrs = {"class" : "point ys-invert
point_good"})
>>> restoran = soup.find_all("div", attrs = {"class" : "breadcrumb"})
>>> yemek = restoran[0].find_all("span")
>>> yemek = yemek[0].find("a").text
>>> sehir = restoran[0].find("a").text.split()
>>> sehir = sehir[0]
>>> adres = restoran[0].find_all("span")
>>> adres = str(adres[1])
>>> adres = adres[7:-7]
>>> adres, yemek, sehir
('Pizza'la, Keçiören (Bağlum Karakaya Mah.)', 'Pizza', 'Ankara')
>>> hiz_degeri = degerler[0].find("span", attrs = "point").text
>>> servis_degeri = degerler[1].find("span", attrs = "point").text
>>> lezzet_degeri = degerler[2].find("span", attrs = "point").text
>>> hiz_degeri, servis_degeri, lezzet_degeri
8.7, 8.8, 8.6
```

Code 1'de¹ görüldüğü üzere restoran linki verilip ilk etapta restorana ait temel başlık niteliği taşıyan kısımlar siteden çekilmiştir.

Code 2

```
>>> altLinkler = []
>>> sayfa_sayisi = 1
>>> yorum_sayfa_sayisi = int(input("Integer bir biçimde kaç sayfa yorum
olduğunu belirtiniz: "))
Integer bir biçimde kaç sayfa yorum belirtiniz: 3
>>> for sayfa in range(1, yorum_sayfa_sayisi + 1):
>>>     geciciLink = link + "?section=comments&page=" + str(sayfa)
>>>     altLinkler.append(geciciLink)
>>>     geciciLink = ""
>>> altLinkler
['https://www.yemeksepeti.com/pizzala-kecioren-baglum-karakaya-mah-
ankara?section=comments&page=1',
 'https://www.yemeksepeti.com/pizzala-kecioren-baglum-karakaya-mah-
ankara?section=comments&page=2',
 'https://www.yemeksepeti.com/pizzala-kecioren-baglum-karakaya-mah-
ankara?section=comments&page=3']
>>> istekler = []
>>> for l in altLinkler:
>>>     istek = requests.get(l, headers = header)
>>>     istekler.append(istik)
>>> istekler
[<Response [200]>, <Response [200]>, <Response [200]>]
>>> users = []
>>> for istek in istekler:
>>>     soup = BeautifulSoup(istik.content, "lxml")
>>>     user = soup.find_all("div", attrs = {"class" : "user"})
>>>     users.append(user)
>>> yorumlar = []
>>> servisler = []
>>> lezzetler = []
>>> hizlar = []
>>> restoranCevabi = []
>>> yorumSayisi = 1
>>> for user in users:
>>>     for i in range(len(user)):
>>>         yorumSayisi += 1
>>>         icerik = user[i].find_all("div", attrs = {"class": "userName"})
>>>         icerik = str(icerik)
>>>         if icerik[88:-14] == 'Yemeksepeti':
>>>             print("Yemeksepeti tarafından kaldırılan yorumdur.")
>>>         elif icerik[24:-8] == 'Restoran Cevabı':
>>>             print("-> Restoran cevabıdır.")
>>>         elif icerik[24:-8] != 'Restoran Cevabı' and icerik[88:-14] !=
```

¹ Code 1'de header'a karşılık gelen değişkenin değer kısmı rapor özelinde paylaşılmaya müsait olmaması sebebiyle soru işareti ile ifade edilmiştir. Yemeksepeti özel olarak paylaştığı için ve gizlilik içerisinde bazı kısımların yürütülmesi gerektiğinden raporda belirtilmeyecektir. Ayrıca link değişkenine rastgele bir restoran verilmiştir.

```

'Yemeksepeti':
>>> yorum = user[i].find_all("p")
>>> puan = user[i].find_all("div", attrs = "restaurantPoints
col-md-12" )
>>> servis = str(puan)
>>> lezzet = str(puan)
>>> hiz = str(puan)
>>> lezzet = lezzet.split("Lezzet: ")
>>> lezzet = lezzet[1][0:2]
>>> if "<" in lezzet:
>>>     lezzet = lezzet[0]
>>>     servis = servis.split("Servis: ")
>>>     servis = servis[1][0:2]
>>> if "<" in servis:
>>>     servis = servis[0]
>>>     hiz = hiz.split("Hız: ")
>>>     hiz = hiz[1][0:2]
>>> if "<" in hiz:
>>>     hiz = hiz[0]
>>>     yorum = str(yorum)
>>>     yorum = yorum[4:-5]
>>>     print(yorumSayisi,". Yorum: ",yorum, "|","Servis:
",servis,"|","Lezzet: ",lezzet,"|","Hız: ",hiz)
>>>     yorumlar.append(yorum)
>>>     servisler.append(servis)
>>>     lezzetler.append(lezzet)
>>>     hizlar.append(hiz)
>>> else:
>>>     print("İÇERİK HATALIDIR!")
Yemeksepeti tarafından kaldırılan yorumdur.
Yemeksepeti tarafından kaldırılan yorumdur.
3 . Yorum: Doyurucu ve lezzetli. Biraz yağlı ama oda normal. Çok
beğendim. Kullanılan malzeme taze olduğu sürece ışinizde rast gider.
Böyle devam. | Servis: 10 | Lezzet: 10 | Hiz: 10
4 . Yorum: Çok geç geldi bi ara hiç gelmiyecek sanmıştım. Dondurulmuş
pizzadan farkı yoktu hatta daha kötüydü buz gibiydi hamuru dünden kalmış
gibiydi ve yanmıştı. Pişmanlıl. | Servis: 2 | Lezzet: 1 | Hiz: 1
5 . Yorum: Pizza sıcacık geldi kaşarı hala erimiş haldeydi lezzet
mükemmel. | Servis: 10 | Lezzet: 10 | Hiz: 10
-> Restoran cevabıdır.
.
.
.
52 . Yorum: Bekledigimden çok çok çok daha iyiydi elinize kolunuza
sağlık yanında gnderdiginiz sos çok yakışmış pizzaya. | Servis: 10 |
Lezzet: 10 | Hiz: 10
-> Restoran cevabıdır.
>>> genel_bilgiler = {
>>>     "Adres" : adres,

```

```
>>> "Yemek" : yemek,  
>>> "Şehir" : sehir,  
>>> "Hız" : hiz_degeri,  
>>> "Servis" : servis_degeri,  
>>> "Lezzet" : lezzet_degeri,  
>>> "Yorum Sayısı" : yorumSayisi  
>>> }  
>>> genel_bilgiler  
{'Adres': "Pizza'la, Keçiören (Bağlum Karakaya Mah.)",  
'Yemek': 'Pizza',  
'Şehir': 'Ankara',  
'Hız': '8,7',  
'Servis': '8,8',  
'Lezzet': '8,6',  
'Yorum Sayısı': 53}
```

Code 2'de² kaç sayfa yorum varsa o sayı verilerek bir restoranın tüm yorumları çekilebiliyor. Bu sayede yorum atlamadan, en iyi veriden ziyade tüm verileri eksiksiz almak asıl amaçtır. Code 2 uzun ve çıktıları da anlamsız geliyor olabilmekte fakat ince düşünülmüş ve yaygın olarak piyasada görülmeyecek bir biçimde veri çekme işlemi Yemeksepeti firması özelinde yapılmasından kaynaklanmaktadır.

Code 2'de bazı yorumların karşılık olarak restoranın kullanıcıya verdiği cevaplar vardır. Sonuçta kullanıcının yorumları önemseniyor fakat Code 2'de restoran yorumlarının yönetildiği kontrol noktasında restoranın cevabı da tutulabilir. Restoranın cevabından yapılabilecek bir madencilik veya analiz süreci var ise, kontrol noktasında restoran cevabı da veri olarak tutulabilir.

Code 2'de fark edilebileceği üzere kullanıcıların bilgileri elde edilebilmesine rağmen, kişiden bağımsız bir veri toplama gerçekleştirilmeye çalışılmıştır. Kişilerin birbirinden ayırt edilmesi (kişilere benzersiz anahtarlar verilmesi) veri seti içerisinde bir anlam ifade etmeyecektir.

Code 2'de öznitelik çıkarımı veri toplarken yapılabilir. Örneğin hız, lezzet, servis gibi nümerik değerlerin ortalaması alınıp kullanıcının restorana yönelik genel puanlandırmasına bakılabilir. Ayrıca metin madenciliği sürecine az etkisi olacağı öngörülürse öznitelik seçimi yapılarak kullanıcıların nümerik puanları dikkate alınmayabilir.

² Code 2'de çıktıların bazıları dökümanın uzamaması açısından noktalar şeklinde kısaltılmıştır.

Projenin gelecek haftalarında bu program kodunun gereksinimlere göre düzenlenmesi ve son kararların verilmesi üzerine; Numpy dizileri veya Pandas serileri gibi yapıların kullanılıp .csv veya farklı bir formatta kaydedilmesi sağlanmalıdır. Bu sayede veriler, Python dilinde dosya işlemleri ile saklanabilir ve sonradan işlenebilir hale getirilebilir. Bu sayede azımsanmayacak büyüklükte bir veri seti elde edilecektir.

REFERANS VE KAYNAKÇA

- [1] Yemeksepeti aydınlatma metni. Link için [tıklayınız](#). (Güvenlidir.)
- [2] Yemeksepeti gizlilik politikası. Link için [tıklayınız](#). (Güvenlidir.)
- [3] Yemeksepeti kişisel verilerin korunumu. Link için [tıklayınız](#). (Güvenlidir.)
- [4] BeautifulSoap kütüphanesi dökümanları. Link için [tıklayınız](#).

EKLER

Bitirme Projesi 2'ye ait doküman, program kodu, haftalık rapor ve ek bilgilerin paylaşıldığı github linki için [tıklayınız](#). (Güvenlidir.)