BİTİRME PROJESİ

Haftalık Rapor – 08.04.2022

8 NİSAN 2022

KIRIKKALE ÜNİVERSİTESİ – BİLGİSAYAR MÜHENDİSLİĞİ İÖ

AHMET MUNGAN – 160255081

İÇİNDEKİLER

ÖZET

Yemeksepeti’nden elde edilen veri seti, metin madenciliği süreçlerinden geçirilmiştir. Lemmatization ve stemming işlemlerinin bir kısmını, zemberek gibi hazır kütüphanelere bağlı kalmadan, Türkçe dilinin dil bilgisini kullanarak genel örüntüler çıkarılmaya çalışılmıştır. Bu bağlamda uygulaması yapılmıştır.

METİN MADENCİLİĞİ SÜRECİ DEVAMI

Tokenization İşlemi

Herbir restoranın yorumları birer doküman olarak düşünülecektir. Fakat kelime sayısı çok olacağından öncesinde geçen hafta durak kelimelerin silinmesi önemlidir.

Tokenization işleminin projenin bu noktasında yapılması algoritmanın maliyetinde iyimser bir değişim yaratacağı kesindir.

Code 1

```
>>> for s in sw:
>>>     for i in range(len(yorumlar)):
>>>         for j in range(len(yorumlar[i])):
>>>             if s == yorumlar[i][j]:
>>>                 print(yorumlar[i][j])
>>>                 yorumlar[i][j] = " "
```

Code 1'de maliyet kabaca hesaplanırsa n^3 gibi maliyetli olacaktır. Dolayısıyla bir sefer bu listeyi parçalayıp o şekilde işlemlerin yapılması daha iyi olacaktır. Code 2'yi kullanmak mantıklı değildir.

Code 2

```
>>> kelimeler = []
>>> gecici = []
>>> for yorum in yorumlar:
>>>     kelimeler.append(yorum.split())
>>> for kelime in kelimeler:
>>>     for k in kelime:
>>>         gecici.append(k)
>>> kelimeler = gecici
>>> len(kelimeler)
3746
>>> kelimeler
['aşırı',
 'soğuk',
 've',
 'tadıda',
 'çok',
 'kötüydü',
 'bu',
... ]
```

Code 2'de tokenization işlemi görülmektedir. Cümleler kelimelere bölünmüştür.

Çoğul Eklerin (-ler, -lar, -leri, -ları vb.) Temizlenmesi

Türkçe sondan eklemeli bir dil olduğu için, dil bilgisi olarak bazı örüntüler bulmak mümkündür. Örneğin Türkçe dilinde "-ler, -lar" genellikle çokluk anlamına gelmektedir fakat bazı istisnalar mevcuttur. İstisnalar dışında "-ler, -lar" ekleri başka anlamlar da kattığı gerçektir. Örneğin: "Okullar açıldı." cümlesinde okulun çokluk anlamına gelirken, "Ahmet beyler nasıl?" cümlesinde beyler hem çokluk anlamına gelirken hem de saygınlık belirten bir ek olduğu göze çarpmaktadır. Dolayısıyla her nasıl olursa olsun, cümle anlamını yüksek oranda değiştirmesi de söz konusu olsa bile kelime olarak büyük bir anlam değişikliği olmamaktadır. Ayrıca çoğul ekleri çekim eklerinden olduğundan temizlenebilir.

Ayrıca çoğul eklerinden sonra gelen eklerin tamamı çekim ekleri olacağından algoritma buna göre kurulmalıdır. Bu duruma örnek verilecek olursa: "O evlerindeydi." cümlesinde kök olarak "ev" olan kelimede çoğul eklerinden sonra gelen tüm ekler "ev" kelimesinin anlamını değiştirmediği açıktır.

Code 3

```
>>> kelimeler
['harikalar',
 'güzeldi',
 'teşekkürler',
 'patatesler',
 ...]
>>> en_uzun_kelime =
len("muvaffakiyetsizleştiricileştiriveremeyebileceklerimizdenmişsinizces
ine")
>>> cokluk_ekleri = ["ler","lar"]
>>> for i in range(len(kelimeler)):
>>>     for j in cokluk_ekleri:
>>>         try:
>>>             for k in range(en_uzun_kelime):
>>>                 if kelimeler[i][-k] + kelimeler[i][-(k-1)] +
kelimeler[i][-(k-2)] == j:
>>>                     kelimeler[i] = kelimeler[i][0:-k]
>>>                     break
>>>         except:
>>>             pass
>>> len(kelimeler)
2734
```

```
>>> for i in range(len(kelimeler)-1, -1, -1):
>>>     for stopword in sw:
>>>         if stopword == kelimeler[i]:
>>>             kelimeler.pop(i)
>>> len(kelimeler)
2621
>>> kelimeler
['harika',
 'güzeldi',
 'teşekkür',
 'patates',
 ...]
```

Code 3'te özel bir algoritma ile çoğul ekleri tespit edilmeye çalışılmıştır. Türkçede bir üne kavuşmuş olan en uzun kelimenin uzunluğu baz alınarak, bir nevi diğer kelimelerin uzunluğu önemsenmeden kontrollü bir yapı ile çoğul ekleri silinmiştir. Burada try-except yapısının kurulmasının sebebi kelime uzunluğunun yetmediği kısımlarda "list index out of range" hatası almamak içindir. Ayrıca çoğul eklerinin tespit edildiği anda maliyetin artmaması için döngülerin kırılması için "break" komutu verilmiştir.

Sayı İçeren Kelimelerin Silinmesi

Sayıların silinmesi ile, sayıların kelimenin içerisinde geçmesi gibi durumların da temizlenmesi gerekmektedir. Öyle ki; sayılar tek başına bir anlam ifade etmiyorsa, kelimelerin içerisinde sayıların geçmesi de bir anlam ifade etmemesi gerekir. Örnek olarak: "1kg" olarak hatalı yazılmış ve bir kilogramı ifade eden kelimenin anlamsız olması beklenir. Ayrıca gerçekten anlamsız olanların da atılması gerekir. Örnek olarak: "0ldu" gibi bir kelime verilebilir. Klavyede "O" harfine yakın olan veya şekli itibariyle benzetilmeye çalışılan harfler olabilmektedir. Bunlar hatalı bir kullanıma kesin çizgilerle örnektir, silinmesi gerekir.

Code 4

```
>>> sw_contains = ["0","1","2","3","4","5","6","7","8","9"]
>>> for i in range(len(kelimeler)-1,-1,-1):
>>>     for j in range(len(kelimeler[i])):
>>>         if kelimeler[i][j] in sw_contains:
>>>             kelimeler.pop(i)
>>>             break
>>> len(kelimeler)
2735
```

Code 4’te, sayı içeren kelimelerin silinmesine dair program kodu görülmektedir.

REFERANS VE KAYNAKÇA

[1] Dil bilgisi için klavuz TDK. 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.)