İbrahim Okan Akveç

1238105104

02.04.2024

Doğal Dil İşleme: Metin Önişleme Uygulama Ödevi

Doğal Dil İşleme dersi Metin Önişleme Uygulama Ödevi kapsamında seçilen bir corpus üzerinde Türkçe dilinde aşağıda listelenmiş adımlar taker taker uygulanmıştır.

1. Sentence Segmentation
2. Tokenization
3. Stemming
4. Lemmazation
5. Extract Stopwords
6. Part of Speech
7. Remove Punctuations

Ödev Google Colab üzerinden uygulanmıştır. [COLAB LİNK](https://colab.research.google.com/drive/1ZwlGDD4gGToTJ4VY-ybDHhZSjG8ulO7m#scrollTo=3u43-wDekg-u)

https://colab.research.google.com/drive/1ZwlGDD4gGToTJ4VY-ybDHhZSjG8ulO7m#scrollTo=3u43-wDekg-u

Metin önişleme işlemleri için Stanford NLP Group tarafından 2020 yılı içerisinde ilk sürümü yayınlanmış olan STANZA kütüphanesi kullanılmıştır.

STANZA kütüphanesi PYTHON dili için geliştirilmiş bir uygulama kütüphanesidir. 70 farklı dil ile çalışabilecek şekilde tasarlanmıştır. Desteklediği diller için free-format metinden kelime ve hece sentezleme, morphologic analiz ve (N)ER gibi özellikler sunmaktadır

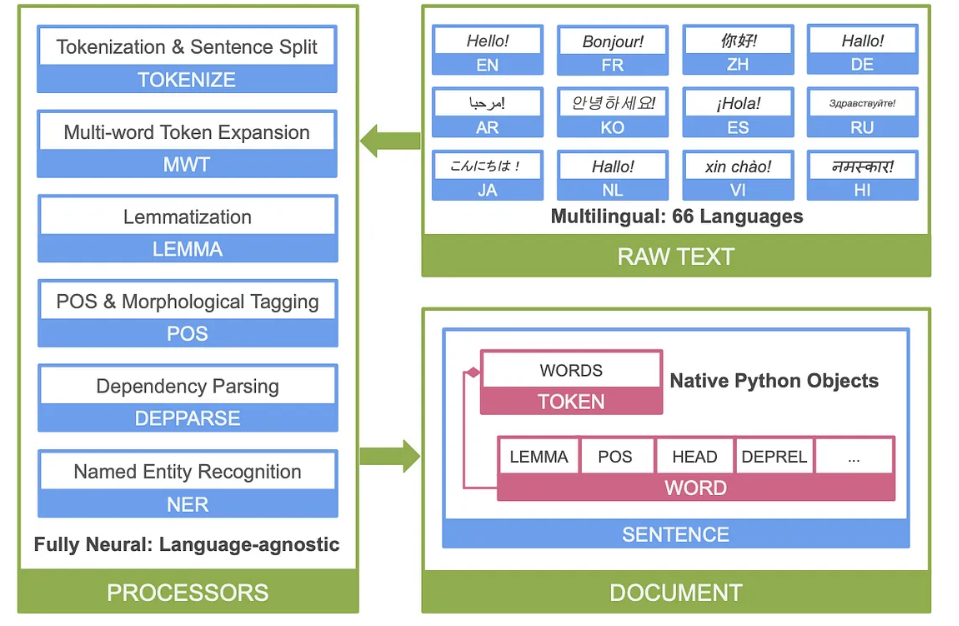
A white box with black check marks

Description automatically generated

STANZA’nın mümkün olan en iyi performansını açığa çıkarabilmesi için, eğitim süreçlerinde yapay sinir ağı tekniklerinden faydalanmaktadır. Eğitim süreçlerinde kullanılan bu modüller Pytorch kütüphanesi üzerine kurulmuştur. Dolayısıyla eğitim sürecinde GPU donanımına sahip bir makinede kullanılırsa çok daha verimli ve yüksek bir performans elde edilebilir.

Ek olarak Stanza, Stanford CoreNLP uygulamasına erişim ve oradaki fonksiyonlarında kullanılabilmesi için bir Python ara katmanı sunmaktadır. Bu sayede Java dilinde olan CoreNLP uygulamasının paketleri ve oradaki tüm methodları kullanılabilir. Örneğin linguistic pattern matching gibi fonksiyonları şuan da Stanza içerisinde barındırmadığı için CoreNLP üzerinden sorunsuz kullanılabilmektedir.

Stanza teknik altyapısı ise aşağıdaki gibidir.



STANZA’nın şuanki sürümü TOKENIZE, LEMMA, POS, NER işlemlerini yerine getirebilmektedir. Bu yüzden ödevin uygulama süreçinde bazı konularda NLTK kütüphanesi de kullanılmıştır.

1. **Sentence Segmentation**

Cümlelere Ayırma işlevi için STANZA pipeline’ı türkçe dili ile “tokenize” processor kullanılarak inşa edilmiştir. Daha sonra bu pipeline’a corpus verisi geçilmiş, ve dönen document nesnesi üzerinden her bir cümleye erişilmiştir.

A screen shot of a computer

Description automatically generated

A close up of text

Description automatically generated

1. **Tokenization**

Birimlere Ayırma işlevi için cümlelere ayırma işlevinde olduğu gibi, her bir cümlenin tokens fonksiyonu ile erişilmiştir.

A computer code with text

Description automatically generated with medium confidence

A screenshot of a computer code

Description automatically generated

1. **Stemming**

STEMMİNG için önce STANZA kütüphanesinden tokenlar alınmış, daha sonra NLTK kütühanesinin PorterStemmer nesnesi ile stem’lere erişilmiştir.

A computer code with text

Description automatically generated

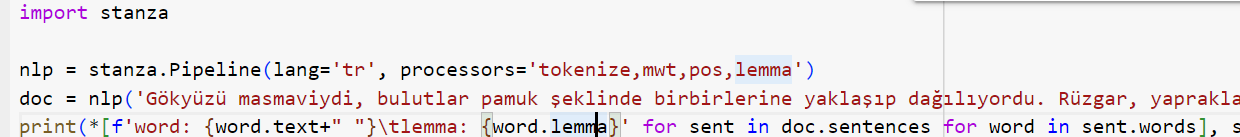
A white background with black text

Description automatically generated

1. **Lemmazation**

Stanza kütüphanesi ile pipeline oluştururlurken lemma processor’u de kullanılmış,

Doc.sentences.words.lemma özelliği ile her bir lemma’ya erişimiştir.



A white background with black text

Description automatically generated

1. **Extract Stopwords**

Etkisiz Sözcük çıkarımı için STANZA’nın desteği olmadığından NLTK kütüphanesi kullanılmıştır. NLTK tükrçe stopwords kelimeleri, corpusta tek tek gezilerek corpustan çıkartılmıştır.

A close-up of a computer screen

Description automatically generated

NLTK kütüphanesinin türkçe için birkaç stopwords örneği:



1. **Part of Speech**

Sözcük türü etiketleme işlevi için yine lemmalama bölümündeki gibi her bir kelimeye erişilmiş, fakat bu kez her bir word’un feats özelliği ile part of speech değerlerine erişilmiştir.

A close-up of a computer code

Description automatically generated

A screenshot of a computer program

Description automatically generated

1. **Remove Punctuations**

Noktalama işaretleri kaldırma işlevi için STANZA kütüphanesi desteklemediğinden dolayı NLTK kütüphanesi kullanılmıştır. Burada yine türkçe stopwords gibi noktalama işaretleri NTLK sözlüğünden alınmış, ve corpus gezilerek tek tek çıkarılmıştır.