

Öğrenci Not Sorgulama-Giriş Sanal Asistanı Tasarımı

Bu ödevde, öğrenci notu sorgulama ve öğrenci notu girişi işlemleri yapabilen bir sanal asistan tasarlanmış ve gerçekleştirilmiştir. Bu asistan İngilizce diline göre eğitilmiş model ve fonksiyonlar kullanan bir konsol uygulamasıdır.

Kullanılan Teknolojiler

spaCy

Bu modül, gelişmiş bir doğal dil işleme kütüphanesidir [1]. Akademik çalışmalardan ziyade gerçek yazılımsal ürünlerde kullanılmak üzere tasarlandığı söylenmektedir. NLP dünyasında SOTA olarak nitelendirebileceğimiz çözümlerden biri olarak literatürde yerini alıyor.



Bu kütüphane ile birlikte gelen birçok özellik bulunuyor. Etiketleme, ayrıştırma, sınıflandırma, NER (named entity recognition), vs. için eğitilmiş hazır yapay sinir ağları ile birlikte çeşitli ek fonksiyonlar da dahil edilmiştir.

PyInstaller

Yazılan bir Python kodunun tek bir EXE dosyası şeklinde çalıştırılabilir hale getirilmesi için bu modül sıkça kullanılır [2]. Kod içindeki bütün bağımlılıkları bir araya toplayarak ayrıca başka bir Python ortamının oluşturulmasına gerek kalmadan tek bir yürütülebilir dosya içinde programın çalışabilmesi sağlanır.



Bu modül Python kodunun EXE formatına dönüştürülüp demo sunumunda çalışmaya hazır bir uygulama çıkmış olması için kullanıldı.

Sanal Asistan Çalışma Mantığı

İlk aşamada öğrenciler, aldıkları dersler ve puanlar için tutulan bir veri yapısı oluşturuldu. Bu veri temelinde bir sözlükten ibaret olmakla birlikte, aşağıdaki örnekteki gibi bir yapıda tutulmaktadır:

```
students = {  
    "Burak": {  
        "Natural Language Processing": 90,  
        "Maths": 80,  
        "English": 70  
    }  
}
```

Sanal asistanın temelde yapacağı 2 işlem vardır: bir öğrencinin bir derste notunu söylemek ve bir öğrencinin bir derste aldığı notu sisteme kaydetmek. Bu işlemlerin gerçekleştirilebilmesi için gereken fonksiyonlar interaktif olacak biçimde yazılmıştır.

Uygulama ilk açıldığında bizi karşılayan ilk cümle şekildeki gibi olacaktır:

```
C:\Windows\System32\cmd.exe - python asistan.py
Microsoft Windows [Version 10.0.19044.2251]
(c) Microsoft Corporation. Tüm hakları saklıdır.

C:\Users\Burak\Desktop\BLG505-NLP\Öğrenci Not Sorgulama - Giriş Sanal Asistanı Tasarımı>python asistan.py
What do you want to do? (querying or entering student grades, 0 to exit, 1 to see all students' data)
```

Şekil 1. İlk Açılış

Örneğin öğrencilerden “Burak” için “Natural Language Processing” dersinden kaç aldığını sormak için şekildeki gibi bir soru sorup cevabını alabiliriz:

```
C:\Windows\System32\cmd.exe - python asistan.py
What do you want to do? (querying or entering student grades, 0 to exit, 1 to see all students' data)
What grade does Burak took from Natural Language Processing course?
Burak took 90 from Natural Language Processing course.
-----
What do you want to do? (querying or entering student grades, 0 to exit, 1 to see all students' data)
```

Şekil 2. Burak'ın Natural Language Processing Dersinden Aldığı Not

Yeni bir kişi için yeni bir not girmek istiyorsak asistana şekildeki gibi bir cümle kurabiliriz:

```
C:\Windows\System32\cmd.exe - python asistan.py
What do you want to do? (querying or entering student grades, 0 to exit, 1 to see all students' data)
John took 75 from NLP course.
The grade 75 for the course NLP is recorded for the student John successfully.
-----
What do you want to do? (querying or entering student grades, 0 to exit, 1 to see all students' data)
```

Şekil 3. John NLP Dersinden 75 Almıştır

Bazı durumlarda eğitilmiş modelin yapısı gereği öğrenci veya ders isimleri algılanmayabilir. Bu durumda asistan emin olmadığı şeyleri soracaktır.

Aşağıdaki örnekte öğrenci ismini tanımlayamadığı için kullanıcıya ayrıca bir soru yöneltilmiştir:

```
C:\Windows\System32\cmd.exe - python asistan.py
What do you want to do? (querying or entering student grades, 0 to exit, 1 to see all students' data)
Get Gülşen's NLP score.
Enter the name of the student: Gülşen
Gülşen does have no grade from NLP course.
-----
What do you want to do? (querying or entering student grades, 0 to exit, 1 to see all students' data)
```

Şekil 4. Öğrenci İsmi Tekrar Sorulmuştur ve Cevap Verilmiştir

Şekildeki örneklerde görüldüğü üzere, not sorgulama ve ekleme işlemleri bu sanal asistan vasıtasıyla yapılabilmektedir.

Yapılan İşlemler

Sanal asistan kullanıcıdan aldığı cümleyi öncelikle kelimesi kelimesine ayırır (tokenizing), ardından PoS tagging ve NER işlemlerini yürüterek öğrenci adı, ders adı ve notun çıkarımını sağlar.

Cümlelerin sorgulama mı yoksa not girişi mi olduğunu ise kelime vektörlerinden yola çıkarak önceden belirlenmiş sorgu ve not girişi kelimelerine yakınlıklarını ölçer.

Örneğin not girişi için sıklıkla kullanılabilecek kelimeler: enter, give, entry.

Not sorgulama için ise sık kullanılan kelimeler şunlardır: query, learn, know, get, see.

Bu kelimeler ile kullanıcı girdisinin kelimelerinin benzerlik oranları karşılaştırılarak en olası komut türü tahmin edilmiş olur.

Not: Bu sanal asistanda normalizasyon metodları uygulanmamıştır ve büyük-küçük harf duyarlı olarak çalışmaktadır.

Hazırlanan uygulama için Google Drive linki:

<https://drive.google.com/file/d/15oOSQcthydgQ9fUNNyMH8zkBTjsIRJg/view?usp=sharing>

Referanslar

- [1] “Industrial-Strength Natural Language Processing”, Explosion.AI, 2022. [Online]. Available: <https://spacy.io>.
- [2] D. Cortesi et al., “PyInstaller Manual”, 2022. [Online]. Available: <https://pyinstaller.org/en/stable/>.