

1906003132015

Doğal Dil İşleme

BAİBÜ Bilgisayar Müh.

Dr. Öğr. Üyesi İsmail Hakkı Parlak

ismail.parlak@ibu.edu.tr

Oda: 335

Model Yayınlama

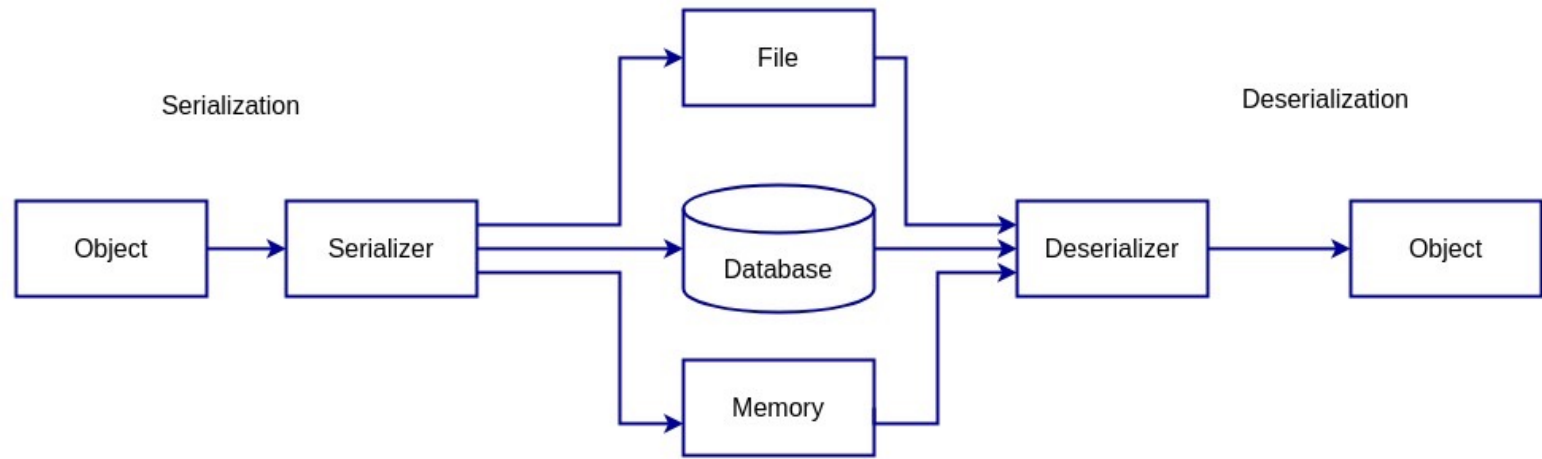
- Bir Python ortamında geliştirilen bir modeli internet üzerinden nasıl kullanıma açabiliriz?
- Modelin eğitim sürecindeki tüm ön işlemler kaydedilip saklanmalıdır.
 - Bu ön işlemler, dolgu sözcüklerini ayıklamak, noktalama işaretleri ve gereksiz metin öğelerini çıkarmak, kelimeleri küçük harflere dönüştürüp lematize etmek, ve dokümanları vektörleştirmek vb. aşamalardan oluşabilir.
 - Vektörleştirme aşamasında vektörleştirici kendi içinde bir sözlük oluşturduğundan, modelin eğitim sürecinde kullanılan vektörleştirici kaybedilmemelidir.

Model Yayınlama

- Eğitimi yapılan model istenilen başarı skoruna ulaştığında diske kaydedilir.
- Modelin sağladığı işlevsellik internet üzerinden kullanıma açıldığında, modele internet kullanıcıları üzerinden gelen sorgular, model eğitimi sırasında gerçekleştirilen tüm ön işlemlerden geçirilerek modelin tüketebileceği bir forma dönüştürülür.
- Dönüşümü yapılmış sorgu modele girdi olarak verilir ve modelin üreteceği çıktı sistemin kullanıcılarına sunulur.

Model Yayınlama

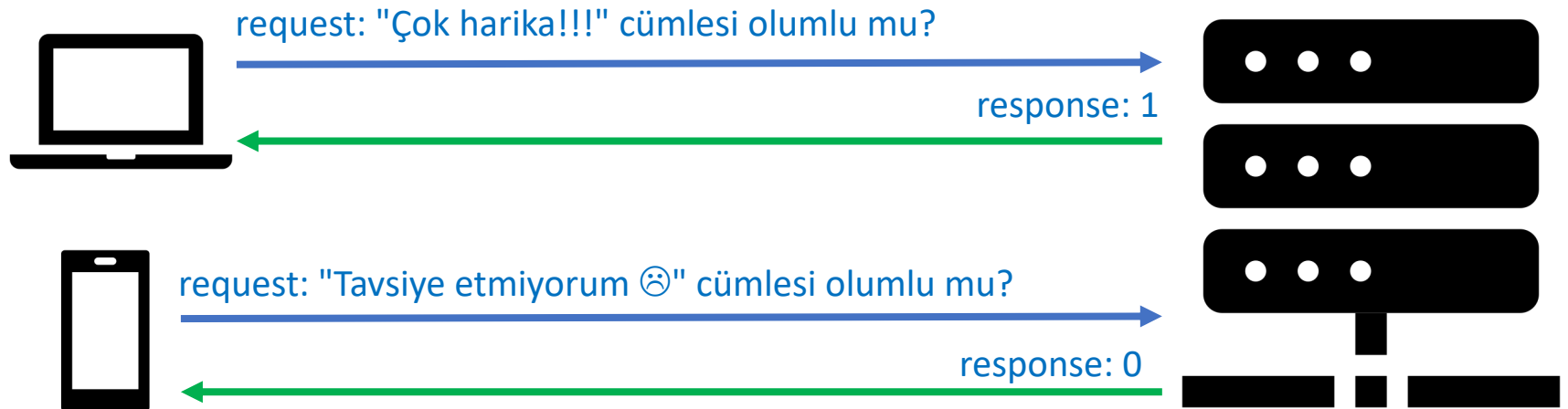
- Python'da List, Dict, Class gibi kompleks tipler istenildiği zaman diske kaydedilebilir (serialization).
- Daha sonra diskteki bu dosya başka bir sisteme taşınıp diskten okunarak tekrar bir Python nesnesine dönüştürülebilir (deserialization).



Model Yayınlama

- Serialization işlemi yapıp diske kaydedilen vektörleştirici, model, vb. ML elemanları bir web sunucusuna taşınıp kullanıma açılabilir.
- Web sunucusu oluşturmak için Python, NodeJS, PHP, .NET, Java, vb. kullanılabilir.
- Python ile bir web sunucusu gerçekleştirmek için Django, Flask, vb. framework'ler bulunmaktadır.

Model Yayınlama



- Web tarayıcısı uygulamaları, sunucuya isteklerini (request) gönderirken tarayıcıda bulunan fetch API'sini JavaScript ile kullanabilir.
- Mobil uygulamalar da geliştirildikleri ortamın onlara sağladığı istek gönderme araçlarını kullanabilirler.
- Web ve mobil istemcilerinde, sunucudan gelen cevabı kullanıcı ihtiyacına göre görselleştirmek istemci cihazında çalışan uygulamanın görevidir.

Flask Sunucusu

- Flask, bir Python sunucusu oluşturma framework'üdür.
- Hafif uygulamaların hızlıca geliştirilmesi için elverişlidir.
- Şablon (template) sunma, istek yollarını işleyebilme gibi kolaylıklar sağlar.

```
@app.route('/')  
def home():  
    return render_template("index.html")
```

Flask Sunucusu

```
app = Flask(__name__)
```

```
@app.route('/')
```

```
def home():
```

```
    return render_template("index.html")
```

```
@app.route('/users')
```

```
def home():
```

```
    return render_template("kullanici-listesi.html")
```

```
app.run(host="0.0.0.0", port=8080)
```


Flask Sunucusu

```
app = Flask(__name__)

@app.route("/classify-texts", methods=["POST"])
def classify():
    data = request.get_json()
    comments = data["comments"]
    comments_clean = [clean_text(c) for c in comments]
    vectors = vectorizer.transform(comments_clean)
    predictions = clf.predict(vectors).tolist()
    response = { "texts": comments, "classes": predictions }
    return jsonify(response)

app.run(host="0.0.0.0", port=8080)
```