



**YILDIZ TEKNİK ÜNİVERSİTESİ
ELEKTRİK-ELEKTRONİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

BLM 4580 DOĞAL DİL İŞLEME

PROJE CHATBOT

Muhammet Ali ŞEN – 20011701
Muhammet Kayra BULUT – 20011901

1. CHATBOT NEDİR?

Chatbot, bir insanla yazılı olarak sohbet edebilen bir yazılım programıdır. Chatbotlar genellikle insan dilini anlayarak ve yanıt vererek sohbet etmeyi amaçlar. Chatbotlar, çeşitli amaçlar için kullanılabilirler, ancak yaygın olarak müşteri hizmetleri, rezervasyonlar ve bilgi dağıtımı gibi alanlarda kullanılırlar. Chatbotlar, genellikle bir web sitesi üzerinden ya da bir mesajlaşma uygulaması üzerinden kullanılırlar ve insanların sorularına cevaplar ya da gerekli işlemleri gerçekleştirirler.

1.1 Chatbotlar nasıl çalışır?

Chatbotlar, çeşitli yöntemlerle programlanırlar ve çalışırlar. Bir chatbot, bir insanla yazılı olarak sohbet edebilmek için özel bir dil öğrenir ve anlar. Bu dil, genellikle insan diline benzer bir dil olur, ancak bir chatbot tarafından anlaşılması için daha yapay bir dil olarak tasarlandığı da olabilir. Chatbotlar, insan dilini anladıklarında, bu dil üzerinden yapılan sorulara yanıt verirler. Chatbotların yanıtları, genellikle önceden programlanmış cevaplar veya veritabanından çekilen bilgilerden oluşur.

1.2 Chatbotların Doğal Dil İşleme İle İlgisi Nedir?

Doğal dil işleme (Natural Language Processing, NLP) teknolojisi, bilgisayar sistemlerine doğal dil kullanarak veri girişi yapılmasını ve bu verilerin anlaşılıp işlenmesini mümkün kılar. Bu sayede chatbotlar gibi yazılım araçları, insanların doğal dilini kullanarak soruları yanıtlayabilir ve görevler yerine getirebilir.

NLP teknolojisi, dilin yapısını, dilbilimsel özelliklerini ve dilin anlamını kavramaya yardımcı olan yöntemleri kullanır. Örneğin, NLP teknolojisi sayesinde chatbotlar, insanların sözcükleri ve cümleleri kullanarak yaptıkları soruları anlayabilir ve uygun yanıtlar vermeyi öğrenebilirler. NLP teknolojisi ayrıca, chatbotların dil öğrenmesini ve dil kullanımını daha etkili hale getirir, bu da chatbotların insanlarla daha doğal bir şekilde iletişim kurmasını sağlar.

Doğal dil işleme teknolojisi, chatbotların yanı sıra birçok farklı alanda da kullanılır. Örneğin, doğal dil işleme teknolojisi, metinlerin çeşitli dillere çevirilmesi, anlam çıkarılması ve sınıflandırılması gibi görevleri gerçekleştirir. Bu sayede, insanların doğal dilini kullanarak arama motorlarında arama yapabilir veya cep telefonlarında dil öğrenme uygulamaları kullanabilirler.

2. CHATBOT ÇEŞİTLERİ

2.1 Retrieval Based Chatbot

Retrieval Based Chatbot, önceden belirlenmiş bir veritabanı içinde yer alan yanıtları kullanarak insanların sorularını yanıtlamaya yönelik tasarıma sahip bir chatbot türüdür. Bu chatbot türü, insanların sorularını doğal dil olarak algılar ve veritabanında eşleşen yanıtları bulur. Eğer veritabanında bir eşleşme bulunamazsa, chatbot genellikle varsayılan bir yanıt verir.

Retrieval Based Chatbot'lar, programlanmış yanıtların bir veritabanında saklanması sayesinde, insanların doğal dil kullanarak yaptıkları soruları anlama ve uygun yanıt verme yeteneğine sahiptir. Bu chatbot türü, insanların belirli bir konuda bilgi edinmek istediği durumlarda kullanışlıdır. Örneğin, bir turizm chatbotu, insanların turistik yerler hakkında bilgi edinmek istediği soruları yanıtlamak için Retrieval Based Chatbot tasarımını kullanabilir.

Ancak, Retrieval Based Chatbot'ların sınırlı bir veritabanına dayanmaları nedeniyle, insanların doğal dil kullanarak yaptıkları daha karmaşık ve özel soruları yanıtlamakta zorluk çekebilirler. Bu nedenle, Retrieval Based Chatbot'ların yanı sıra, çeşitli dil modellemek için tasarımı olan Rule Based Chatbot'lar ve öğrenme tabanlı chatbot'lar da mevcuttur.

2.2 Generative Based Chatbot

Generative based Chatbot, insanların doğal dil kullanarak yaptıkları sorulara doğal dil olarak yanıt veren ve kendi yanıtlarını üretebilen bir chatbot türüdür. Bu chatbot türü, dil modelleme yöntemlerini kullanarak dilin yapısını ve anlamını öğrenir ve buna göre yanıtlar üretir. Generative based Chatbot'lar, insanların doğal dil kullanarak yaptıkları daha karmaşık ve özel soruları yanıtlamaya daha yatkındır ve daha doğal bir şekilde iletişim kurabilirler.

Generative based Chatbot'lar, öğrenme tabanlı chatbot'lar gibi, eğitim verilerine dayanarak çalışırlar. Bu chatbot türü, verilen eğitim verilerini inceler ve dil modellemek için kullanılan yöntemleri kullanarak dilin yapısını ve anlamını öğrenir. Daha sonra, insanların doğal dil kullanarak yaptıkları soruları anlar ve uygun yanıtlar üretebilir. Generative based Chatbot'lar, insanların doğal dil kullanarak yaptıkları sorulara yanıt verirken, insanların dilini kullanarak daha doğal bir şekilde iletişim kurarlar.

Generative based Chatbot'lar, insanların doğal dil kullanarak yaptıkları sorulara daha doğal bir şekilde yanıt verebildikleri için, birçok farklı alanda kullanılır. Örneğin, bir müşteri hizmetleri chatbotu, insanların doğal dil kullanarak yaptıkları sorulara daha doğal bir şekilde yanıt verebilir ve bu sayede insanlarla daha doğal bir şekilde iletişim kurabilir.

3. CHATBOT PROJEMİZ

BU projemizde derin öğrenme tekniklerini kullanarak bir chatbot oluşturacağız. Chatbot, kategorileri (intents), modeli ve yanıtları içeren veri kümesi üzerinde eğitilecektir. Kullanıcının mesajının hangi kategoriye ait olduğunu sınıflandırmak için özel bir tekrarlayan sinir ağı (LSTM) kullanıyoruz ve ardından bir cevap vereceğiz.

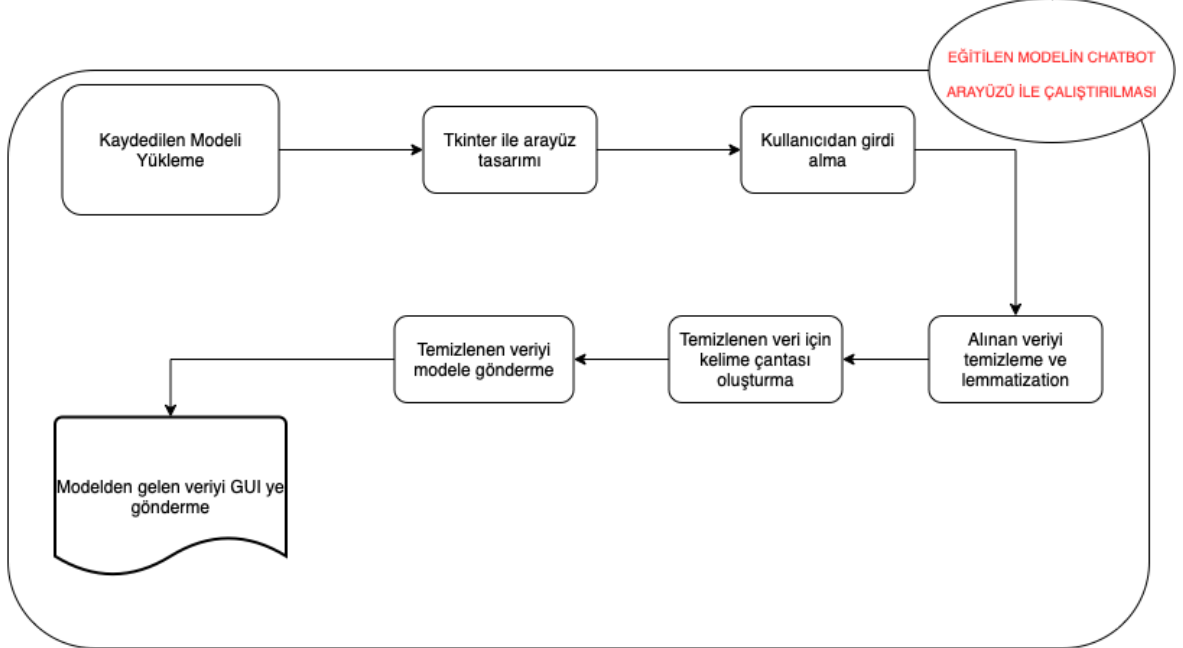
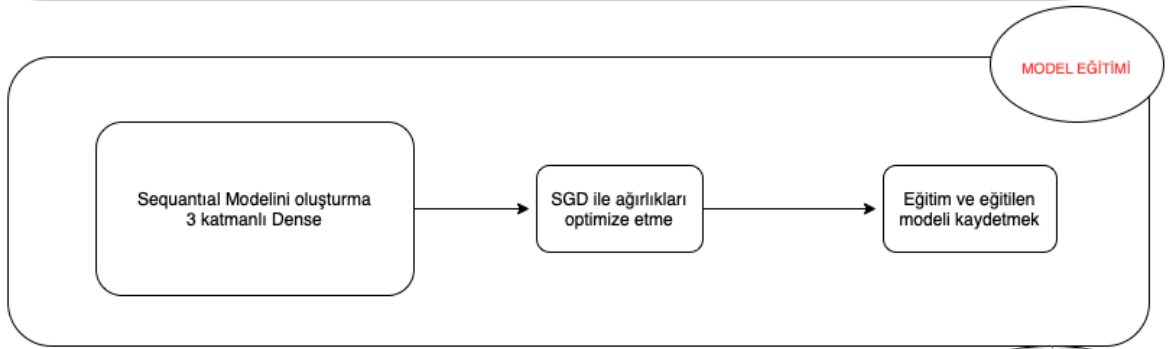
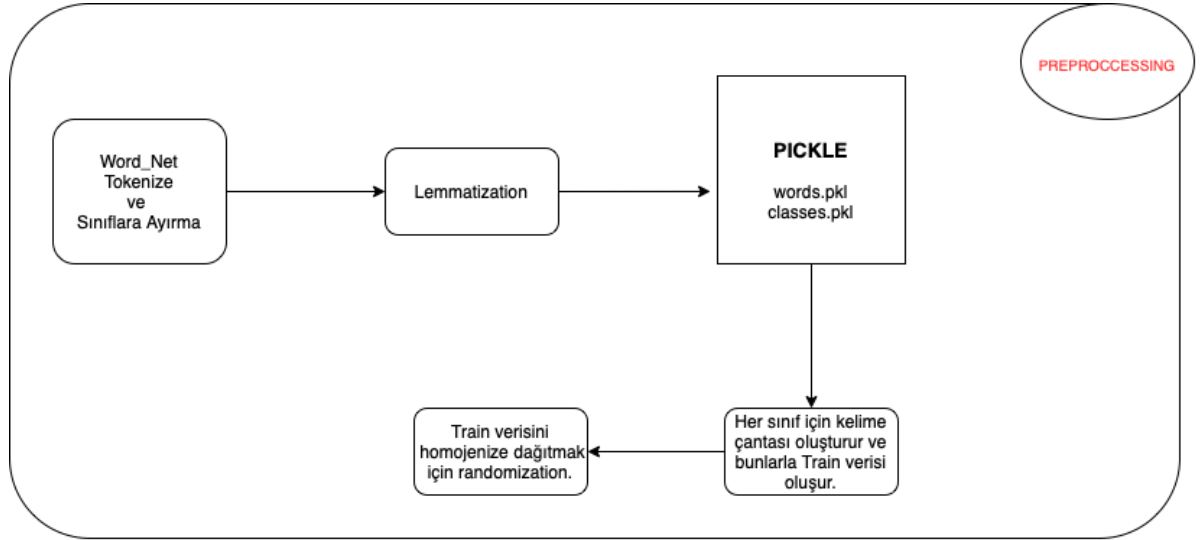
Retrieval Based Chatbot projemizde NLTK, Keras, Numpy, Pickle, Json gibi Python kütüphanelerini Python 3.9 versiyonunu kullandık.

Doysamız ve açıklamalarına kısaca değinirsek;

- **Intents.json** – Pattern ve cevapları içeren veri setimiz.
- **train_chatbot.py** – Modelimizi eğitmek için yazılmış python dosyası.
- **Words.pkl** – Kelime listemizi içeren Python objesi ‘Pickle’ dosyası. Kelimelerimiz bu dosyada tutmaktayız
- **Classes.pkl** – Kelime sınıflarını içeren Python objesi ‘Pickle’ dosyası. Kelimelerin bağıntılı olduğu etiket (tag) değerlerini barındırır.
- **Chatbot_model.h5** – Nöron ve ağırlıklardan oluşan eğitilmiş model dosyamız.
- **Chatgui.py** – Arayüz tasarımını Tkinter ile yaptığımız python dosyamız.

Projemizi kabaca özetlersek

1. Veri setimizi oluşturduk ve sorulara göre cevaplarımızı hazırladık
2. Veriler üzerinde bir dizi temizleme, gövdeleme (lemmatization), etiketleme (tokenization) işlemlerini gerçekleştirerek veri setimizi eğitime hazırladık.
3. Eğitimi modelledik. Bunun için Sequential ile 3 katmanlı bir yapay sinir ağı oluşturduk. LSTM ile oluşan modelimizi eğitim ve test veri setlerimizle eğittik ve test ettik.
4. Modelimizi kaydederek yeni sorgulara hazırladık.
5. Arayüz üzerinden yeni sorguları alarak eğitilen modelimizdeki oluşan tahmini cevapları ekrana gönderdik.



3.1 Veri Setini Oluşturma ve Yükleme

Öncelikle, içeriğini bizim oluşturduğumuz, temel soru ve cevaplardan oluşan train_chatbot.py isimli bir dosya oluşturduk. Chatbotumuz için gerekli paketleri import edip Python projemizde kullanacağımız değişkenleri yazdık.

Veri dosyası JSON biçimindedir, bu nedenle JSON dosyasını Python'a ayrıştırmak için json paketini kullandık.

intents.json dosyamız bu şekilde görünmektedir.

```
{
  "intents": [
    {
      "tag": "greeting",
      "patterns": [
        "Hi there",
        "How are you",
        "Is anyone there?",
        "Hello",
        "Good morning",
        "Good afternoon"
      ],
      "responses": [
        "Hello! How can I help you today?",
        "Hi there! What can I do for you?",
        "Hello, what can I assist you with today?"
      ]
    },
    {
      "tag": "yildiz_technique_university",
      "patterns": [
        "Tell me about the Yıldız technique",
        "What is the Yıldız technique?",
        "How do you do the Yıldız technique?"
      ],
      "responses": [
        "Yıldız Technical University is a public research university"
      ]
    },
    {
      "tag": "yildiz_technique_university",
      "patterns": [
        "Tell me about Yıldız Technical University",
        "What is Yıldız Technical University?",
        "How do I learn more about Yıldız Technical University?",
        "Can you tell me more about Yıldız Technical University?",
        "What programs does Yıldız Technical University offer?"
      ],
      "responses": [
        "Yıldız Technical University is a public research university",
        "Some popular programs at Yıldız Technical University include"
      ]
    }
  ]
}
```

3.2 Veri Setini Temizleme ve Önışlemden Geçirme

Metin verileriyle çalışırken makine öğrenimi veya derin öğrenme modeli yapmadan önce veriler üzerinde çeşitli ön işlemler gerçekleştirmemiz gerekir. Gereksinimlere bağlı olarak, verileri önceden işlemek için çeşitli işlemler uygulamamız gerekir.

Tokenization, metin verileri üzerinde yapabileceğiniz en temel ve ilk şeydir. Tokenization, tüm metni kelimeler gibi küçük parçalara ayırma işlemidir.

Burada kalıpları yineliyoruz ve `nltk.word_tokenize()` işlevini kullanarak cümleyi tokenize ediyoruz ve kelime listesindeki her kelimeyi ekliyoruz. Ayrıca etiketlerimiz için bir sınıf listesi oluşturuyoruz.

Sonrasında her kelimeyi lemmatize edeceğiz ve yinelenen kelimeleri listeden kaldıracğıız. Lemmatizing (gövdeleme), bir kelimeyi lemma formuna dönüştürme ve ardından tahmin yaparken kullanacağımız Python nesnelerini depolamak için bir 'Pickle' dosyası oluşturma işlemidir.

3.3 Eğitim ve Test Verilerini Oluşturma

Şimdi girdi ve çıktısını vereceğimiz eğitim verisini oluşturacağız. Girdimiz kalıp olacak ve çıktımız girdi kalıbımızın ait olduğu sınıf olacaktır. Ancak bilgisayar metni anlamıyor, bu yüzden metni sayısallaştırmamız gerekiyor.

Sayısallaştırılan ve eğitim için hazırlanan verilerimiz modelimize gönderilmek üzere hazırdır.

3.4 Modeli Oluşturma

Eğitim verilerimiz hazır olduktan sonra 3 katmanlı derin bir yapay sinir ağı kurduk. Bunun için Keras'ın API'ini kullanıyoruz. Modeli 200 döngü (epoch) eğittikten sonra modelimizde yüksek oranda doğruluk elde ettik. Modeli 'chatbot_model.h5' olarak kaydettik.

3.5 Cevap Tahmini

Eğitilmiş modeli yükleyerek ve ardından bottan gelen yanıtı tahmin edecek bir grafik kullanıcı arayüzü kullanacağız. Model bize sadece ait olduğu sınıfı söyleyecektir, bu yüzden sınıfı tanımlayan bazı fonksiyonları uyguluyoruz. Akabinde o sınıfa ait bir cevap alacağız.

Yine gerekli paketleri import edip modelimizi eğitirken oluşturduğumuz 'words.pkl' ve 'classes.pkl' 'Pickle' dosyalarını yükledik.

Sınıfı tahmin etmek için, eğitim sırasında yaptığımızla aynı şekilde girdi sağlamamız gerekecek. Bu yüzden, metin önışlemesini gerçekleştirecek ve ardından sınıfı tahmin edecek bazı işlevler oluşturduk.

Sınıfı tahmin ettikten sonra o sınıfta tek bir cevap varsa o cevap dönecek eğer birden fazla cevap varsa rastgele bir cevabı seçerek ekrana gönderecek bir dizayn tasarladık.

Grafik arayüzü için tonlarca faydalı kitaplıkla birlikte gelen Tkinter kitaplığını kullandık. Kullanıcıdan giriş mesajını alıp ve ardından bottan yanıtı almak ve GUI'de görüntülemek için oluşturduğumuz yardımcı fonksiyonları kullandık.

4. PROGRAMI ÇALIŞTIRMA

Chatbotu çalıştırmak için iki ana dosyamız var;

- train_chatbot.py
- chatapp.py.

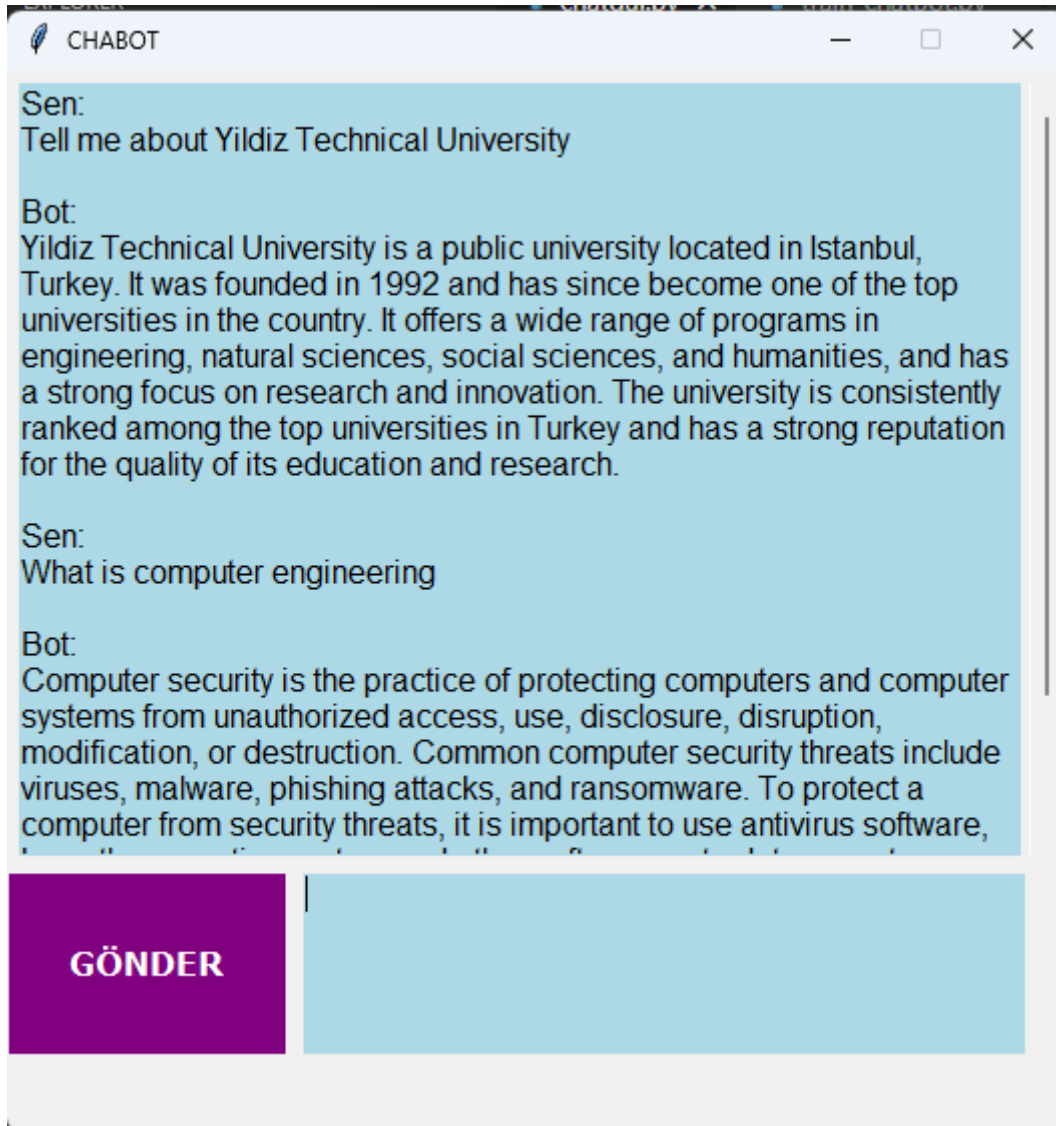
İlk olarak, train_chatbot.py dosyası modeli eğitiyoruz. Bunun için terminalde:

‘python train_chatbot.py’ script kodunu yürütüyoruz.

Eğitim sırasında herhangi bir hata görülmezse modeli başarılı bir şekilde oluşturulmuş demektir. Ardından uygulamayı çalıştırmak için ikinci dosyanın çalıştırılması gerekir. Bunun için terminalde:

‘python chatgui.py’ script kodunu yürütüyoruz.

Program birkaç saniye içinde bir GUI penceresi açılacaktır.



5. SONUÇ

Yaptığımız doğal dil işleme ve veri bilimi projesinde, sohbet robotlarını anlayarak doğal dil işleme dinamiklerini kavramaya çalıştık. Ayrıca Python'da bir sohbet robotunun derin öğrenme sürümünü uygulayarak derin öğrenme yöntemleri üzerinde araştırma yaptık. Bu projemizde verilerimiz iş gereksinimlerine göre özelleştirebilir. Bizde veri içerik dosyamızı Yıldız Teknik Üniversitesi ve Bilgisayar Bilimleri konseptiyle oluşturduk. Birçok veriyi elle kendimiz girerek oluşturmak bir hayli zaman aldı. Ancak çıktılar güzel olduğunu görünce verdiğimiz emeğe değdiğini anlamış olduk. Chatbot'lar her yerde kullanılıyor ve tüm işletmeler iş akışlarında bot uygulamayı dört gözle bekliyor.