

Konuşmacı Tanıma Sistemi Ayrıntılı Kod Raporu

Bu rapor, verilen Python kodunu derinlemesine inceleyerek, Konuşmacı Tanıma Sistemi'nin işlevselliğini ve uygulama detaylarını açıklamaktadır.

1. Genel Bakış:

Bu kod, konuşmacıları seslerinden tanıyan bir sistem oluşturmak için makine öğrenimi tekniklerini kullanmaktadır. Sistem, Mel Frekans Cepstral Katsayıları (MFCC) adı verilen ses özelliklerini çıkararak ve bunları bir Destek Vektör Makinesi (SVM) modeliyle eğiterek çalışır. Ek olarak, Google Speech Recognition API'sini kullanarak ses transkripsiyonu ve kelime sayımı gibi özellikler sunar.

2. Kod Analizi:

Kod, aşağıdaki ana bölümlere ayrılabilir:

2.1. Kütüphane İçe Aktarma:

Kod, öncelikle gerekli Python kütüphanelerini içe aktarır. Bu kütüphaneler şunlardır:

- + 'tkinter': Grafiksel kullanıcı arayüzü (GUI) oluşturmak için kullanılan standart Python kütüphanesi.
- + 'filedialog', 'messagebox', 'ttk': 'tkinter' kütüphanesinin alt modülleri, dosya seçimi, mesaj kutuları ve ilerleme çubuğu gibi GUI öğeleri sunar.
- + 'librosa': Ses analizi için kullanılan bir Python kütüphanesi. MFCC özelliklerini çıkarmak için kullanılır.
- + 'numpy': Bilimsel hesaplamalar için kullanılan bir Python kütüphanesi. Diziler ve matrislerle çalışmak için kullanılır.
- + 'sklearn': Makine öğrenimi algoritmaları ve araçları için kullanılan bir Python kütüphanesi. SVM modeli eğitimi, veri kümesinin bölünmesi ve model değerlendirme için kullanılır.
- + 'speech_recognition': Sesi metne dönüştürmek için kullanılan bir Python kütüphanesi. Google Speech Recognition API'sini kullanır.
- + 'pyaudio': Ses kaydı ve oynatma için kullanılan bir Python kütüphanesi.
- + 'wave': WAV formatındaki ses dosyalarıyla çalışmak için kullanılan bir Python kütüphanesi.

- + 'os': İşletim sistemi ile etkileşim için kullanılan bir Python kütüphanesi. Dosya ve izin işlemleri için kullanılır.
- + 'pydub': Ses dosyalarıyla manipülasyon yapmak için kullanılan bir Python kütüphanesi.
- + 'threading': Eş zamanlı kod yürütmeyi sağlamak için kullanılan bir Python kütüphanesi.
- + 'matplotlib.pyplot': Grafikler oluşturmak için kullanılan bir Python kütüphanesi. MFCC histogramı oluşturmak için kullanılır.

2.2. Fonksiyon Tanımları:

Kod, aşağıdaki fonksiyonları tanımlar:

2.2.1. 'extract_mfcc(audio_file)':

Bu fonksiyon, verilen ses dosyasından MFCC özelliklerini çıkarır ve bir histogram oluşturur.

- + Giriş: 'audio_file': Analiz edilecek ses dosyasının yolu.
- + Çıkış: MFCC özellikleri (bir NumPy dizisi) veya bir hata durumunda 'None'.

İşlevsellik:

1. 'librosa.load()' fonksiyonunu kullanarak ses dosyasını yükler. Bu fonksiyon, ses sinyalinin (y) ve örnekleme oranını (sr) döndürür.
2. Ses ön işleme adımında, ses sinyalinden ortalama değer çıkarılır. Bu, gürültü azaltma ve sinyalin DC bileşenini ortadan kaldırma amacıyla yapılır.
3. 'librosa.feature.mfcc()' fonksiyonu, ses sinyalinden 13 MFCC katsayısını çıkarır. Bu katsayılar, ses sinyalinin spektral zarfını temsil eder ve konuşmacı tanıma için faydalı özelliklerdir.
4. MFCC katsayılarının ortalaması alınarak tek bir özellik vektörü elde edilir.
5. 'matplotlib.pyplot' kütüphanesi kullanılarak MFCC özelliklerinin histogramı oluşturulur. Histogram, ses dosyasının MFCC dağılımını görselleştirmeye yardımcı olur.
6. Histogram, ses dosyasının adıyla ilişkilendirilmiş bir PNG dosyası olarak kaydedilir.
7. Fonksiyon, MFCC özelliklerini bir NumPy dizisi olarak döndürür. Herhangi bir hata durumunda 'None' değeri döndürür.

2.2.2. 'train_model()':

Bu fonksiyon, SVM modelini eğitir.

+ Giriş: Yok.

+ Çıkış: Eğitilmiş SVM modeli veya yeterli veri yoksa 'None'.

İşlevsellik:

1. MFCC özelliklerini ('X') ve konuşmacı etiketlerini ('y') depolamak için boş listeler oluşturur.
2. Geçerli dizinde ""_Ses"" ile biten tüm klasörleri bulur. Her klasör, bir konuşmacıya ait ses örneklerini içerir.
3. Her konuşmacı klasörü için, klasör adından konuşmacı adını çıkarır (örneğin, "Konuşmacı1_Ses" klasöründen "Konuşmacı1" adını alır).
4. Klasördeki her WAV dosyası için 'extract_mfcc()' fonksiyonunu kullanarak MFCC özelliklerini çıkarır.
5. MFCC özellikleri ve konuşmacı etiketi 'X' ve 'y' listelerine eklenir.
6. 'X' listesinde en az bir özellik vektörü varsa, veri kümesi eğitim ve test kümelerine ayrılır. 'sklearn.model_selection.train_test_split()' fonksiyonu, veri kümesini %80 eğitim ve %20 test oranıyla rastgele böler.
7. 'sklearn.svm.SVC()' fonksiyonu kullanılarak doğrusal çekirdekli bir SVM modeli oluşturulur.
8. 'svm_model.fit()' fonksiyonu kullanılarak SVM modeli, eğitim verileriyle eğitilir.
9. 'svm_model.predict()' fonksiyonu kullanılarak test verileri üzerinde tahmin yapılır.
10. 'sklearn.metrics.accuracy_score()' fonksiyonu kullanılarak modelin doğruluğu hesaplanır.
11. 'sklearn.metrics.classification_report()' fonksiyonu kullanılarak modelin performansını değerlendirmek için detaylı bir sınıflandırma raporu oluşturulur.
12. Modelin doğruluğu ve F1-skoru mesaj kutusu ile kullanıcıya gösterilir.
13. Fonksiyon, eğitilmiş SVM modelini döndürür. Yeterli veri yoksa 'None' değeri döndürür.

2.2.3. 'transcribe_audio(audio_file)':

Bu fonksiyon, sesi metne dönüştürür.

- + Giriş: 'audio_file': Transkribe edilecek ses dosyasının yolu.
- + Çıkış: Transkribe edilmiş metin veya bir hata durumunda hata mesajı.

İşlevsellik:

1. 'speech_recognition.Recognizer()' fonksiyonunu kullanarak bir 'Recognizer' nesnesi oluşturur.
2. 'speech_recognition.AudioFile()' fonksiyonunu kullanarak ses dosyasını açar.
3. 'Recognizer.record()' fonksiyonunu kullanarak sesi kaydeder.
4. 'Recognizer.recognize_google()' fonksiyonunu kullanarak sesi Google Speech Recognition API'si ile transkribe eder. Türkçe dilini kullanmak için 'language="tr-TR"' argümanı belirtilir.
5. Başarılı olursa transkribe edilmiş metni döndürür.
6. 'sr.UnknownValueError' hatası oluşursa, "Google Speech Recognition sesi anlayamadı" mesajını döndürür.
7. 'sr.RequestError' hatası oluşursa, hata mesajını döndürür.

2.2.4. 'count_words(text)':

Bu fonksiyon, verilen metindeki kelime sayısını hesaplar.

- + Giriş: 'text': Kelime sayısı hesaplanacak metin.
- + Çıkış: Metindeki kelime sayısı.

İşlevsellik:

1. Metni boşluklara göre bölerek kelime listesini oluşturur.
2. Listedeki kelime sayısını döndürür.

2.3. App Sınıfı:

Kod, 'App' adlı bir sınıf tanımlar. Bu sınıf, uygulamanın ana penceresini ve GUI öğelerini oluşturur.

2.3.1. '__init__(self)':

'App' sınıfının yapıcı fonksiyonu, pencere başlığını, boyutunu ve arka plan rengini ayarlar. Ayrıca, SVM modelini, kayıt durumunu ve ses verilerini depolamak için değişkenler tanımlar.

- + 'self.svm_model': Eğitilmiş SVM modelini depolar. Başlangıçta 'None' olarak ayarlanır.
- + 'self.recording': Kayıt durumunu belirten bir boole değişkeni. 'False' olarak başlatılır.
- + 'self.audio_frames': Kaydedilen ses verilerini depolamak için kullanılan bir liste.
- + 'self.p': 'pyaudio.PyAudio()' fonksiyonu kullanılarak oluşturulan bir PyAudio nesnesi.
- + 'self.stream': Ses akışını temsil eden bir PyAudio akış nesnesi. Başlangıçta 'None' olarak ayarlanır.

2.3.2. 'create_widgets(self)':

Bu fonksiyon, GUI öğelerini oluşturur. GUI, sol tarafta bir buton çerçevesi ve sağ tarafta bir sonuç çerçevesi olmak üzere ikiye ayrılmıştır.

Sol Çerçeve Öğeleri:

- + 'self.train_button': Modeli eğitmek için kullanılır. 'train_model()' fonksiyonunu çağırır.
- + 'self.record_button': Ses kaydını başlatmak için kullanılır. 'start_recording()' fonksiyonunu çağırır.
- + 'self.stop_button': Ses kaydını durdurmak için kullanılır. 'stop_recording()' fonksiyonunu çağırır. Başlangıçta devre dışıdır.
- + 'self.analyze_button': Kaydedilen sesi analiz etmek için kullanılır. 'analyze_recording()' fonksiyonunu çağırır. Başlangıçta devre dışıdır.

Sağ Çerçeve Öğeleri:

- + 'self.speaker_label': Tanımlanan konuşmacının adını gösterir.
- + 'self.word_count_label': Transkribe edilmiş metindeki kelime sayısını gösterir.
- + 'self.progress': Model eğitimi, ses analizi ve transkripsiyon gibi işlemlerin ilerleyişini gösteren bir ilerleme çubuğu.
- + 'self.text_label': Transkribe edilmiş metni gösteren bir metin kutusu.

2.3.3. 'train_model(self)':

Bu fonksiyon, 'train_model()' fonksiyonunu çağırarak SVM modelini eğitir.

- + İlerleme çubuğunu sıfırlar ve günceller.
- + 'train_model()' fonksiyonunu çağırarak modeli eğitir.
- + Model eğitimi tamamlandığında ilerleme çubuğunu %100'e ayarlar.
- + Model başarıyla eğitildiyse bir bilgi mesajı kutusu görüntüler.

2.3.4. 'start_recording(self)':

Bu fonksiyon, ses kaydını başlatır.

- + 'self.recording' değişkenini 'True' olarak ayarlar.
- + 'self.audio_frames' listesini temizler.
- + 'pyaudio.PyAudio.open()' fonksiyonunu kullanarak bir ses akışı açar. Akış, 44.1 kHz örnekleme oranıyla mono olarak yapılandırılır.
- + 'self.record_button' devre dışı bırakılır.
- + 'self.stop_button' etkinleştirilir.
- + 'self.record()' fonksiyonunu ayrı bir thread'de başlatır.

2.3.5. 'record(self)':

Bu fonksiyon, ses kaydını gerçekleştirir.

- + 'self.recording' değişkeni 'True' olduğu sürece çalışır.
- + 'self.stream.read()' fonksiyonunu kullanarak ses verilerini okur.
- + Okunan ses verilerini 'self.audio_frames' listesine ekler.

2.3.6. 'stop_recording(self)':

Bu fonksiyon, ses kaydını durdurur.

- + 'self.recording' değişkenini 'False' olarak ayarlar.
- + 'self.stream.stop_stream()' fonksiyonunu kullanarak ses akışını durdurur.
- + 'self.stream.close()' fonksiyonunu kullanarak ses akışını kapatır.
- + 'self.record_button' etkinleştirilir.
- + 'self.stop_button' devre dışı bırakılır.
- + 'self.analyze_button' etkinleştirilir.
- + Kaydedilen ses verilerini "kaydedilen_ses.wav" adlı bir WAV dosyasına yazar.
- + Kayıt işleminin tamamlandığını bildiren bir bilgi mesajı kutusu görüntüler.

2.3.7. 'analyze_recording(self)':

Bu fonksiyon, kaydedilen sesi analiz eder.

+ SVM modeli eğitilmişse:

- + İlerleme çubuğunu sıfırlar ve günceller.
- + 'extract_mfcc()' fonksiyonunu kullanarak kaydedilen ses dosyasından MFCC özelliklerini çıkarır.
- + MFCC özellikleri başarıyla çıkarıldıysa:
 - + Eğitilmiş SVM modelini kullanarak konuşmacıyı tahmin eder.
 - + Tahmin edilen konuşmacının adını 'self.speaker_label' etiketinde görüntüler.
 - + İlerleme çubuğunu %33'e ayarlar.
 - + 'transcribe_audio()' fonksiyonunu kullanarak sesi transkribe eder.
 - + Transkribe edilmiş metni 'self.text_label' metin kutusunda görüntüler.
 - + İlerleme çubuğunu %66'ya ayarlar.
 - + 'count_and_display_words()' fonksiyonunu ayrı bir thread'de başlatır.
- + MFCC özellikleri çıkarılamadıysa, bir hata mesajı kutusu görüntüler ve ilerleme çubuğunu %100'e ayarlar.
- + SVM modeli eğitilmemişse, kullanıcıyı modeli eğitmeleri konusunda uyan bir uyarı mesajı kutusu görüntüler.

2.3.8. 'count_and_display_words(self, text)':

Bu fonksiyon, transkribe edilmiş metindeki kelime sayısını hesaplar ve GUI'de görüntüler.

- + 'count_words()' fonksiyonunu kullanarak kelime sayısını hesaplar.
- + Hesaplanan kelime sayısını 'self.word_count_label' etiketinde görüntüler.
- + İlerleme çubuğunu %100'e ayarlar.

2.3.9. 'predict_speaker(self, audio_file)':

Bu fonksiyon, verilen ses dosyası için konuşmacıyı tahmin eder.

- + SVM modeli eğitilmişse:
 - + 'extract_mfcc()' fonksiyonunu kullanarak ses dosyasından MFCC özelliklerini çıkarır.
 - + MFCC özellikleri başarıyla çıkarıldıysa:
 - + Eğitilmiş SVM modelini kullanarak konuşmacıyı tahmin eder ve tahmin edilen konuşmacının adını döndürür.
 - + MFCC özellikleri çıkarılamadıysa, 'None' değerini döndürür.
- + SVM modeli eğitilmemişse, kullanıcıyı modeli eğitmeleri konusunda uyan bir uyarı mesajı kutusu görüntüler ve 'None' değerini döndürür.

2.4. Ana Program:

Kodun son kısmı, ana programı içerir.

- + 'if __name__ == "__main__":' ifadesi, kodun yalnızca doğrudan çalıştırıldığında değil, bir modül olarak içe aktarıldığında çalıştırılmamasını sağlar.
- + 'App()' sınıfından bir nesne oluşturur.
- + 'app.mainloop()' fonksiyonunu çağırarak GUI'nin olay döngüsünü başlatır.

3. Sonuç:

Bu rapor, verilen Python kodunun ayrıntılı bir analizini sunmaktadır. Kod, MFCC özellikleri ve SVM modeli kullanarak konuşmacı tanıma işlemini gerçekleştirmektedir. Ayrıca, Google Speech

Recognition API'si ile ses transkripsiyonu ve kelime sayımı gibi ek özellikler sunmaktadır. GUI, kullanıcıların modeli eğitmesine, ses kaydı yapmasına, sesi analiz etmesine ve sonuçları görüntülemesine olanak tanır.