En bild som visar text, logotyp, Teckensnitt, Grafik

Automatiskt genererad beskrivning

Muhannad Naser  
DS23

Projekt inom Voice Recognizing

Tillsammans med:

* **David Norman**: Ansvarade för applikationsutvecklingen och arbetade tillsammans med andra teammedlemmar på integrationen.
* **Ming Fondberg**: Arbetade med både applikationen och modellen samt samarbetade med integrationen.
* **Parsan Amani**: Arbetade med modellen och datasetet, och samordnade med teamet kring integrationen.

1. Inledning

Detta projekt syftar till att skapa en applikation som rekommenderar Spotify-spellistor baserat på användarens humör, vilket upptäcks från ett röstprov. Genom att analysera ljudets egenskaper förutser applikationen användarens humör och föreslår en lämplig spellista. Huvudkomponenterna inkluderar en FastAPI-applikation, en TensorFlow-modell för humörklassificering och Spotify API för spellister.

2. Projektöversikt

* **Använda teknologier**:
  + **Spotipy**: Ett Python-bibliotek för Spotify API, vilket möjliggör smidig integration med Spotify-spellistor.
  + **FastAPI**: Ett modernt, snabbt webb-ramverk för att skapa API

, använt här för att hantera förfrågningar och svar.

* + **Librosa**: Ett Python-bibliotek för ljudbearbetning och analys.
  + **TensorFlow**: Används för att skapa och köra maskininlärningsmodellen för humörförutsägelse.
  + **NumPy**: För datahantering och matematiska operationer.

Pipelineförklaring

* **Spotify API Credentials**: Vi använder Spotipy för att hantera Spotify API-autentisering med client\_id och client\_secret, vilket möjliggör åtkomst till spellistorna.
* **Modellinladdning och Initialisering**: Applikationen laddar en förtränad modell (final\_model.keras) som kan klassificera olika humör baserat på röstdata.
* **FastAPI-Setup**: FastAPI används för att skapa ett RESTful API med två huvudsakliga endpoints: / för att testa servern och /analyze-voice/ för röstanalys.
* **Humörförutsägelse**:
  + Funktionen predict\_mood bearbetar det uppladdade ljudet genom att extrahera MFCC-funktioner via librosa. Sedan gör modellen en förutsägelse baserat på dessa funktioner.
  + Humörkategorierna i denna modell är exempelvis: “calm”, “energetic” och “happy”. Förutsägelsen baseras på sannolikhetsvärden där modellen väljer det humör som har högst sannolikhet.
* **Spellistehämtning**: Funktionen get\_playlist\_for\_mood hämtar en URL till en spellista baserat på det förutspådda humöret. Spellistorna är mappade till olika humörkategorier i applikationen, t.ex. “calm” länkar till en lugn spellista.
* **Röstanalys-Endpoint**: Endpointen /analyze-voice/ tar emot en uppladdad röstfil, bearbetar ljudet, förutser humöret och returnerar en Spotify-spellista som passar humöret. Resultatet innehåller även ett meddelande som bekräftar att röstanalysen lyckades.

Utmaningar och Lösningar

Under projektets gång uppstod vissa utmaningar, exempelvis:

* **Utmaningar och Lösningar**
* En stor utmaning i projektet var att modellen alltid förutspådde samma humör, "calm," oavsett vilket ljud som analyserades. Det visade sig att modellen hade svårt att skilja mellan olika humör baserat på ljudet.
* Vi kom fram till att problemet kunde bero på att ljuddata inte gav tillräckligt med information eller att modellen var för inriktad på ett humör. För att lösa detta testade vi olika inställningar för ljudfunktionerna och justerade modellens parametrar. Vi planerar också att lägga till mer variation i träningsdatan så att modellen bättre kan identifiera olika humör.

5. Slutsats och framtida utveckling

Projektet levererar en röstaktiverad rekommendation av Spotify-spellistor baserat på humör. Framöver skulle projektet kunna utvecklas genom att lägga till fler humörkategorier, förbättra modellens förutsägelsesäkerhet eller lägga till fler spellistor för varje humörkategori.