بسم الله الرحمن الرحیم

شرکت مهندسی نرم افزاری هلو

گزارش شناسایی گویندگان از یک فایل صوتی

کاری از امیرعلی نسیمی

جریان کاری برنامه

۱. وارد کردن کتابخانه‌ها و تنظیم مسیرها

- کتابخانه‌های ``psutil`، `os`، `sys`، time`` و `pathlib` وارد می‌شوند.
- مسیر پروژه به `sys.path` اضافه می‌شود تا امکان دسترسی به ماژول‌های پروژه فراهم شود.

۲. تعریف تابع `main`

- مسیر فایل صوتی `Zan.wav` تعیین می‌شود.
- شیء `SpeakerDiarization` با استفاده از مدل `pyannote/speaker-diarization-3.1` و توکن دسترسی ساخته می‌شود.
- زمان شروع پردازش ثبت می‌شود.

۳. پردازش فایل صوتی

- متد `process_audio` از کلاس `SpeakerDiarization` برای پردازش فایل صوتی فراخوانی می‌شود.
- زمان پایان پردازش محاسبه و مدت زمان پردازش محاسبه می‌شود.

٤. نمایش نتایج

- نتایج تشخیص گوینده‌ها نمایش داده می‌شود.
- مدت زمان پردازش و درصد استفاده از حافظه نیز نمایش داده می‌شود.

٥. اجرای تابع `main`

- اگر فایل به طور مستقیم اجرا شود، تابع `main` فراخوانی می‌شود. دقت ۱۰۰ درصدی، سرعت ۰.۶۲ ثانیه و استفاده ۷۶ درصدی از حافظه از ویژگی های مربوط به این مورد می باشد.

توضیح مختصر از فایل `diarization.py`

۱. تعریف کلاس `SpeakerDiarization`

- کلاس شامل یک سازنده است که مدل `pyannote` را بارگیری می‌کند و دستگاه مناسب (GPU یا CPU) را تنظیم می‌کند.
- متد `process_audio` فایل صوتی را پردازش کرده و نتایج تشخیص گوینده‌ها را برمی‌گرداند.

جریان کلی کار

- برنامه از فایل `run.py` شروع می‌شود.
- تابع `main` فراخوانی می‌شود که فایل صوتی را پردازش می‌کند.
- نتایج پردازش به صورت شروع و پایان هر بخش و گوینده مربوطه نمایش داده می‌شود.
- مدت زمان پردازش و میزان استفاده از حافظه نیز نمایش داده می‌شود.

نوع شبکه عصبی و نحوه آموزش

در این کد، شبکه عصبی که برای تشخیص گوینده‌ها استفاده می‌شود، از مدل‌های پیش‌آموزشی شده‌ی `pyannote/speaker-diarization-3.1` بهره می‌برد. این مدل‌ها در حوزه شناسایی گوینده و جدا کردن بخش‌های گفتاری مختلف به گوینده‌های مختلف استفاده می‌شوند.

۱. نوع شبکه عصبی

- شبکه‌های عصبی بازگشتی (RNN) و Long Short-Term Memory (LSTM):

این نوع شبکه‌ها به دلیل توانایی‌شان در مدل‌سازی توالی‌ها و داده‌های ترتیبی، در تشخیص گفتار و تفکیک گوینده‌ها کاربرد زیادی دارند. شبکه‌های LSTM قادر به یادگیری وابستگی‌های بلندمدت در داده‌های ترتیبی هستند که برای تحلیل سیگنال‌های صوتی بسیار مفید است.

- شبکه‌های عصبی کانولوشنی (CNN):

شبکه‌های CNN معمولاً در تشخیص ویژگی‌های مکانی و زمانی در سیگنال‌های صوتی استفاده می‌شوند. این شبکه‌ها با اعمال فیلترهای کانولوشنی به داده‌های ورودی می‌توانند ویژگی‌های پیچیده و مهم صوتی را استخراج کنند.

۲. نحوه آموزش

- پیش‌آموزش (Pre-training):

مدل‌های `pyannote` از قبل روی مجموعه داده‌های بزرگ و متنوع صوتی آموزش دیده‌اند. این فرایند شامل مراحل زیر است:

- جمع‌آوری داده‌ها: شامل مجموعه‌های بزرگ از داده‌های گفتاری متنوع از منابع مختلف.
- پیش‌پردازش داده‌ها: شامل پاک‌سازی نویز، نرمال‌سازی و برچسب‌گذاری داده‌های گفتاری.
- آموزش مدل: مدل‌ها با استفاده از تکنیک‌های بهینه‌سازی مانند الگوریتم‌های نزول گرادیان و با استفاده از مجموعه داده‌های بزرگ، پارامترهای خود را تنظیم می‌کنند تا بتوانند به خوبی الگوهای گفتاری و تفکیک گوینده‌ها را بیاموزند.

- استفاده از مدل پیش‌آموزشی:

در این کد، مدل پیش‌آموزشی `pyannote` بارگیری و مورد استفاده قرار می‌گیرد. این مدل‌ها بدون نیاز به آموزش مجدد، قابلیت تشخیص و تفکیک گوینده‌ها را دارند و به کمک توکن دسترسی مخصوص، از سرویس‌های `Hugging Face` استفاده می‌کنند.

جمع‌بندی

مدل‌های `pyannote` برای تشخیص گوینده‌ها از ترکیبی از شبکه‌های عصبی پیشرفته استفاده می‌کنند که شامل RNN، LSTM و CNN است. این مدل‌ها از قبل بر روی مجموعه داده‌های بزرگ و متنوع آموزش دیده‌اند و در این کد، تنها با بارگیری و استفاده از این مدل‌های پیش‌آموزشی، عملیات تشخیص گوینده‌ها انجام می‌شود.