Gender Classification Using Audio Embeddings

This project aims to build a pipeline for gender classification using audio embeddings extracted from various pre-trained models. The repository contains a Jupyter notebook for building and testing the pipeline, as well as a Streamlit application for demonstrating the inference process.

Files Description

1. `Gender_Classification_Pipeline_Building.ipynb`

This Jupyter notebook includes the entire pipeline for gender classification based on audio embeddings. The notebook is structured as follows:

- **Data Preparation**: Loading, cleaning, and preprocessing the audio data from the VoxCeleb_gender dataset.
- **Noise Reduction**: Applying noise reduction techniques using `librosa`.
- **Speech Detection**: Detecting voice activity in the processed audio using the Silero VAD model.
- **Audio Embeddings Extraction**: Extracting audio embeddings using multiple pre-trained models, including Wav2Vec2 and Whisper.
- **Building Classifiers**: Training several classifiers (XGBoost, Random Forest, SVM, etc.) using the extracted embeddings to classify gender.
- **Model Evaluation**: Performing grid search to optimize the hyperparameters of the models and evaluating their performance.

The notebook also includes visualizations and diagnostics to test the effectiveness of the pipeline. This is essential for evaluating the accuracy of the gender classification and for debugging.

2. `task.py` (Streamlit Application)

This Python script provides a Streamlit-based demo for gender classification. It allows users to upload audio files, processes them using the pre-trained models, and predicts the gender of the speaker.

Features:

- **Audio Recording or Upload**: The user can either record their voice directly or upload an audio file.
- **Noise Reduction and VAD**: The script applies noise reduction and voice activity detection to the uploaded audio.
- **Embedding Extraction**: The preprocessed audio is passed through the Whisper Base model to extract embeddings.
- **Gender Prediction**: The embeddings are fed into the trained classifiers (XGBoost and SVM), and the predicted gender is displayed.

How to Run the Streamlit Demo

Step 1: Install the required dependencies

You can install the required Python libraries by using the `requirements.txt` file provided:

```
pip install -r requirements.txt
```

Step 2: Run the Streamlit Application

After installing the dependencies, you can run the Streamlit app with the following command:

```
streamlit run task.py
```

Step 3: Upload or Record Audio

Once the Streamlit app is running, you can interact with the demo by either recording your voice or uploading an audio file. The application will preprocess the audio, extract embeddings, and predict the gender of the speaker.

Summary

- The **notebook** provides the full pipeline for audio preprocessing, embedding extraction, and classifier training.
- The **Streamlit demo** allows for real-time inference using pre-trained models and displays the gender prediction based on audio input.