FSP Finder

AI powered foul speech pattern (FSP) detector and automatic censoring tool

Members: Jared Able, Duncan Clark, Elzbieta Polak, Shuo Yan

Links: GitHub, Hugging Face Spaces

About this tool

FSP (Foul Speech Pattern) Finder is an AI-powered explicit content detector and automatic censoring tool useful for preparing music files for radio airplay. We use a fine-tuned version of OpenAI's automatic speech recognition model Whisper to transcribe the lyrics of uploaded music files (with word timestamps). Explicit terms (i.e., curse words, racial slurs, etc.) are identified in the transcript and using demucs the vocals stem is muted at the identified times producing an edited file suitable for the air.

This tool can process files one at a time or in batches. The web interface allows the user to view the full transcript of each track along with the words that will be censored. Additionally, you'll get a link to the Genius entry for the lyrics of the track, along with a similarity score (MER), for cross referencing accuracy.

Requirements

- `pip install -r requirements.txt` to install the necessary dependencies
- `ffmpeg` (for handling mp3 files)
- A Genius API key. This key should be placed in the `GENIUS_API_TOKEN` variable in `fsp.py` (or set as `GENIUS_API_TOKEN` in your system environment).

Starting the web interface: In the project directory, execute `python app.py` in the command line

On first execution `app.py` will convert the configuration files in `./lora_config` to a full Whisper model stored at `./whisper-medium-ft` (the full Whisper model is necessary for using Whisper-timestamped to produce word timestamps). Please note, running this app locally in any reasonable amount of time will require a CUDA enabled GPU with a minimum of 12GB of VRAM (recommended 16GB or more).

Training and methodology

We trained OpenAI/whisper-medium.en, and english-only automatic speech recognition model, on a portion of the DALI Dataset. We identified tracks in the DALI dataset that were (1) in English, (2) had a working link to YouTube to grab the audio file. DALI contains timestamped transcriptions of the tracks. We first separated the vocals-only stem from each track, then from that stem extracted only the segments identified in the DALI dataset "lines" entries, i.e., short (5-10s) clips from each track identified as having vocals present. These audio files were saved as mono `.wav` files with a 16 kHz sample rate.

From this training data we extracted only those lines that a fine-tuned toxicity version of the cardiffnlp toxicity classifier identified as being explicit. This resulted in a dataset of roughly 2000 audio chunks with timestamps. We split this dataset into train/val/test sets, and trained both (1) a LoRA adapter, and (2) the final LM (also called *proj_out*) layer of our Whisper model. Our fine-tuned Whisper model decreases the match error rate (MER) of the test set from 0.58424 to 0.48113 (with a similar decrease in word error rate from 0.64305 to 0.52992). These error rates don't tell the whole story though: by specifically training on explicit content our model has become very sensitive to explicit content, with the ultimate goal of minimizing false negatives (i.e., maximizing recall).

Training notebooks for creating the audio files and metdata for DALI, along with preparing the Whisper dataset, and fine-tuning the model can all be found in the `./notebooks` folder. In addition, the notebooks found in `./notebooks/line-dataset-normalizer` played a crucial role in cleaning our data: the lyrics transcriptions in the DALI dataset often contained spelling error, unneccesary spaces and word concatenations, or other punctuation which prevented Whisper from correctly identifying the transcript. The master lists containg the relevant metadata (filename, transcript, etc.) for each of the train/val/test sets is contained in `./data`.

Future implementation

- Option for the user to add their own words to the list of words to be censored by highlighting additional words in the full transcript provided by the model.
- Method for censoring "explicit sounds", i.e., non-vocals noises that may be offensive (gun shots, sexually explicit sounds, etc.).
- Use of a language model to detect context-dependent explicit words (i.e., references to specific drugs or firearms) within lines markes as explicit.

Credits

- This project was completed as part of the Erdos Institute's Deep Learning Bootcamp in Summer 2025.
- All training data comes from the DALI Dataset.