GUIDELINES FOR DOWNLOADING MEL SPECTROGRAMS AND AUDIO FILES FROM MTG-JAMENDO

This document is a guide to downloading the Mel spectrograms and audio from MTG-Jamendo so you can work with them locally. To correctly download the Mel spectrogram files and the low-quality (.mp3) audio from the MTG-Jamendo dataset, you need to set up a Python environment, install the required dependencies, run the repository's official download script, and ensure sufficient storage space. This document describes the exact process we followed to complete the download successfully from scratch on a Windows system with Anaconda or Miniconda installed.

Therefore, the first step is to install Anaconda (https://www.anaconda.com/download). Once it's installed, you can proceed with the steps below.

First, you need to open Anaconda Prompt (not the regular Windows Command Prompt). Inside this terminal, create a new Conda environment called jamendo that uses Python 3.11 (this version plays nicely with modern numpy, pandas, and matplotlib needed for the project):

```
conda create -n jamendo python=3.11 -y
```

Once created, activate the environment:

```
conda activate jamendo
```

Next, we need to install the main packages with Conda to avoid compiling from source, which often causes issues on Windows when using pip directly.

```
conda install numpy=1.24 pandas=1.5 matplotlib=3.6 -y
```

After that, we need to install gdown with pip, because it isn't available in the Conda channels:

```
pip install gdown
```

Then we download the official dataset repository from GitHub with:

```
git clone https://github.com/MTG/mtg-jamendo-dataset.git
```

And moved into the project folder:

```
cd mtg-jamendo-dataset
```

Inside the repository, we deliberately avoided using the full requirements.txt, because it pins older versions of the core libraries that are incompatible with Python 3.11. Instead, we installed only the remaining lightweight dependencies manually (such as tqdm, requests, or scikit-learn):

```
pip install tqdm requests scikit-learn
```

With the environment ready, we proceeded to download the Mel spectrograms. We first created a destination folder on the computer:

```
mkdir "C:\Users\PORTATIL\Desktop\mtg_data\melspecs_extraidos"
```

Because our goal was to work with tracks that have at least one mood/theme tag, we chose the autotagging-moodtheme\melspecs subset. Finally, we need to run the repository's download script, which accepts parameters to select the dataset subset, data type, destination folder, and whether to automatically unpack and delete the .tar files. The final command we executed was:

```
python scripts/download/download.py ^
  --dataset autotagging_moodtheme ^
  --type melspecs ^
  --from mtg-fast ^
  --unpack --remove ^
  "C:\Users\PORTATIL\Desktop\mtg_data\melspecs_extraidos"
```

With this, the Mel spectrograms should have been downloaded into subfolders and stored at the path we selected. Now we move on to the audio. In my case, I had to download the audio to an external drive because my computer didn't have enough free space, so we created a different path. To do that, we created a destination folder on a drive with sufficient capacity:

```
mkdir "E:\TFM\mp3_low"
```

Then we ran the repository's official download script. The script accepts parameters to specify the dataset subset, the audio quality, the destination folder, and whether to automatically unpack and delete the .tar archives, as with the mel spectrograms. The exact command we used was:

```
python scripts/download/download.py ^
  --dataset raw_30s ^
  --type audio-low ^
  --from mtg-fast ^
  --unpack --remove ^
  "E:\TFM\mp3_low"
```

This command downloads the 89 .tar archives that make up the low-quality raw_30s dataset, unpacks them into the specified folder, and then deletes the .tar files to free up space.

And that's it! Now that everything is downloaded, we can start working with our audio files and Mel spectrograms.

If you prefer an alternative download method, you can find the relevant instructions in the README of the MTG-Jamendo GitHub repository.

https://github.com/MTG/mtg-jamendo-dataset