Multimedia Search and Retrieval (MMSR), Winter Term 2023/24

The main objective of this year's practical part of the MMSR class is to create a *content-based music retrieval system* leveraging various kinds of multimedia data. To achieve this goal, you will have to solve three tasks. The dataset we will use throughout the course is a subset of the Music4All-Onion dataset, which is available from https://zenodo.org/record/6609677#.Y0ff7XZBxD8. The dataset is explained in detail in this paper: https://bit.ly/3eqK6Pz. To facilitate your work, we will also provide some additional files here: https://drive.google.com/file/d/18bzjBNNeTWKGA38dm7xSOofRGQz9ZQ_D/view?usp=share_link . These files are sufficient to solve the first task. Before you start working on the task, please first familiarize yourself with the dataset.

Task 1: Simple text-based similarity and retrieval
Introduction: 18.10.23
Q&A: 08.11.23 (given by Marta Moscati & Oleg Lesota)
Deadline for submission: 15.11.23, 12:00

In the first task, you will implement a number of music retrieval systems (mostly based on text features), then qualitatively compare their outputs and report the results.

Retrieval systems: The input (query) to the system is the title and artist of a song (track). The output of the system should be a list of songs (title and artist) of length N that are similar to the query song. In this exercise, we will investigate four retrieval systems: random baseline and three text-based systems.

- **Random Baseline**: regardless of the query track, this retrieval system randomly selects N tracks from the rest of the catalog. Make sure that the system produces new results for each query / run.
- **Text-based(cos-sim, tf-idf)**: given a query, this retrieval system selects the N tracks that are most similar to the query track. The similarity is measured as cosine similarity between the tf-idf representations of the lyrics of the tracks. I.e.

$$sim(query,\ target\_track)\ =\ cos(tf\_idf(query),\ tf\_idf(target\_track))$$

- **Text-based(cos-sim, <feature>):** similar to Text-based(cos-sim, tf-idf), however choose a different text-based feature instead of tf-idf (i.e., word2vec or BERT representations of the lyrics)

$$sim(query,\ target\_track)\ =\ cos(<feature>(query),\ <feature>(target\_track))$$

- **Text-based(<similarity>, <feature>):** similar to Text-based(cos-sim, <feature>), however choose a new *combination* of similarity measure and text-based feature (e.g., you can use cos-sim with a representation of the lyrics not selected for previous systems yet)

$$sim(query,\ target\_track)\ =\ <similarity>(<feature>(query),\ <feature>(target\_track))$$

Qualitative analysis: Select as queries three tracks you are familiar with and retrieve 10 tracks with each system, including the random baseline. This will result in the following number of lists:

$$N_{lists}\ =\ N_{tracks}\ *\ N_{retrieval\ systems}\ =\ 3\ *\ 4\ =\ 12$$

For each query track, qualitatively compare the retrieved tracks (with the query and other tracks in the result list), analyzing for instance whether the list includes tracks by the same artist or of the same genre. Also, investigate the relevance of the retrieved tracks for the query, i.e., given the query can you speculate why the tracks in the result list have been retrieved?

Hints on implementation:

- Make sure to make your implementation reusable in the future: allow for easy switching between different similarities measures, feature sets and values of N (number of retrieved results).
- Implement functionality to store the ids of retrieved songs for each query song; this will greatly facilitate any further processing/analysis that you will have to do for other tasks, and save you from having to re-run similarity computations many times.

Hint on dataset: There are roughly 10,000 tracks labeled by an alphanumerical id. Textual features are stored as a .tsv file with a column containing the identifier of the track, and the remaining columns containing the component of the feature vector. Each file is named as id_lyrics_<*name_of_feature*>_mmsr.tsv. Information regarding the title, artist, and album of the tracks are stored in the file id_information_mmsr.tsv with a column containing the identifier of the track, and the remaining columns containing the artist, song, and album name.

Lab report: Write a short lab report (2-3 pages) in ACM format, https://www.acm.org/publications/proceedings-template), detailing your approach, experimental setup, and results/findings.

Files:

1. Your report as a .pdf
2. Source code (e.g., link to Github)
3. (Optional): link to Overleaf version of the report

Submission: Deadline is November 15 at 12:00, via mail to *markus.schedl@jku.at*, *oleg.lesota@jku.at* **and** *marta.moscati@jku.at*.