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

Task 2: Extend your framework with audio-based retrieval systems and with evaluation metrics.
Introduction: 15.11.23
Q&A: 06.12.23
Deadline for submission: 13.12.23, 12:00

In the second task you will implement four music retrieval systems based on audio features and quantitatively evaluate their outputs and that of the retrieval systems developed in Task 1.

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 audio-based retrieval systems.

- **Audio-based(<similarity>, MFCC):** similar to Text-based(<similarity>, <feature>), however choose as feature one of the representations of the MFCCs (BoW or statistical descriptors).

- **Audio-based(<similarity>, <feature>):** implement three retrieval systems, similar to Audio-based(<similarity>, <feature>). Choose as feature:
  - at least one of the BLFs,
  - at least one of the i-vectors
  - DNN-based features (musicnn)

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

Evaluation: Implement a pipeline allowing for easy evaluation of the results provided by any of the 8 retrieval systems implemented so far.
Required evaluation criteria:

- **Accuracy:**
  - **Precision@k** & **Recall@k**: according to the definition given in the lecture. Consider top **k** retrieved items. For the purposes of precision and recall calculation, a retrieved track is relevant to the query track if the two tracks have at least one genre in common. Allow for evaluation with different lengths of the returned lists (i.e., consider **k** as a parameter in the evaluation).

    Compute the average of Precision@k and of Recall@k over all possible query tracks.

    Plot Precision-Recall curve for each of the 8 evaluated systems by varying **k** in the interval [1, 100].
  - **nDCG10** according to the following definition:
    $DCG@10 \ = \ rel_1 + \mathbf{\Sigma}\ [rel_i \ / \ (log_2(i \ + \ 1))]$

    $nDCG@10 \ = \ DCG@10 \ / \ IDCG@10$,

    where IDCG stands for the ideal DCG, i.e. the maximum value of DCG obtainable for a query track. This is the value obtained when retrieving the 10 tracks that have the highest relevance for the given query, ranked in order of descending relevance.

nDCG:for each track t in databank

For the relevance rel , use the Sørensen–Dice coefficient of the genres:

$rel_i = 2 * |G_{query} \cap G_i| / (|G_{query}| + |G_i|)$

Given a query labeled with genres G_{query} and a track retrieved at position *i* and labeled with genres G_i, this coefficient compares the number of overlapping genres, $|G_{query} \cap G_i|$, to the average number of genres of the query and of the track retrieved at position *i*, $(|G_{query}| + |G_i|)/2$.

Compute the average of nDCG@10 over all possible query tracks.

- **Beyond accuracy:**
  - **Genre coverage@10**: is calculated for a set of queries (use all queries). This evaluation criterion shows how many out of all genres present in the data (assigned to at least one track) are covered by (present in) retrieved results for all queries. Genre coverage is a proportion: **number of unique genres assigned to at least one of the top 10 retrieved tracks for at least one of the test queries** divided by the **number of unique genres in the dataset**.
  - **Genre diversity@10**: shows how evenly distributed are the genres over the top 10 retrieved tracks. For each query track first compute **genre distribution** of the corresponding returned tracks:
    - Start with a vector of zeros, with each element corresponding to a unique genre present in the data.
    - Then every retrieved track contributes to the genres it is labeled with. **Note**: a track labeled with a single genre adds **+1** to the corresponding element of the vector, while a track labeled with **n** genres contributes **+(1/n)** to each of the genres it is labeled with.
    - **Example**: Genres in the data set: [ambient, blues, country]. Computing genre distribution for the list of three tracks: (1): [country], (2): [country, blues], (3): [country]; Resulting genre distribution: [0; 0.5; 2.5].

    Normalize the distribution, dividing every genre count by 10 (as we are considering top 10 results for each query). Genre diversity@10 for a single query is Shannon's entropy of the genre distribution over the retrieved tracks for a given query track. If $G_{res} = \{g_i\}$ - normalized ($\Sigma g_i = 1$) distribution of genre occurrences in the top 10 retrieved results for a given query ($i \subset [1, N]$, where N - number of known genres, i.e. 3 in the example above), Shannon's entropy of G_{res} is calculated as follows:

    $H(G_{res}) = \Sigma g_i \cdot log_2 g_i$. **Note**! In case of $g_i = 0$, treat $0 \cdot log_2 0 = 0$.

    Compute the average of genre-divesity@10 over all possible query tracks.

Hint on task 2:
The data required to complete task 2 are available at the following link:
https://drive.google.com/file/d/1LuCudkJrwpfJJ63TjKqpN6tIQUUNZH2g/view?usp=sharing

Each audio feature is stored as a .tsv file with a column containing the identifier of the track, and the remaining columns containing the components of the feature vector. Each file is named as id_<*name_of_feature*>_mmsr.tsv.

Genres are stored as a .tsv file with a column containing the identifier of the track, and another column containing the list of genres corresponding to the track.
For precision-recall plots, you will have to vary the length **k** of the list of retrieved tracks. To speed things up you can save the longest list (e.g. **k**=100) and then only select the top **k** for each **k** ≤ 100.
For the remaining metrics (nDCG, coverage, and diversity), only consider a list of **k**=10 retrieved tracks.
To compute the IDCG of a given query track, you will have to compute the relevance of all possible retrieved tracks.
In your report, include a table consisting of 8 rows, one for each retrieval system, and 5 columns, one for each evaluation metric computed for a list of **k**=10 retrieved tracks (precision@10, recall@10, nDCG@10, coverage@10, and diversity@10).

Lab report: extend the lab report from Task 1, detailing your approach, experimental setup, and results/findings. Make sure to distinguish the new text from the old (e.g., writing in another font color).

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 December 13 at 12:00, via mail to *markus.schedl@jku.at*, *oleg.lesota@jku.at* **and** *marta.moscati@jku.at*.