

# Audio-based Music Processing

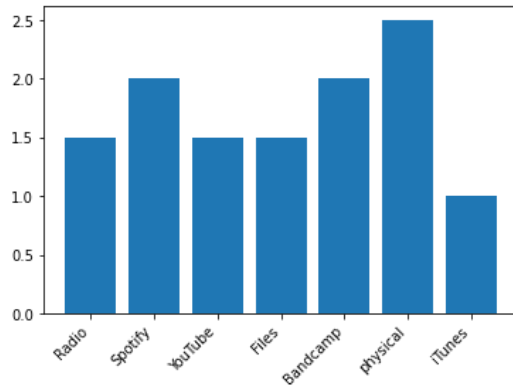
11.04.2023

# 1 Wie hört ihr Musik?

```
# - Radio (per App)      Ii
# - Spotify             II
# - YouTube             Ii
#   (downloader)       Ii
# - Bandcamp            II
# - CDs/Vinyls          Iii
# - iTunes              I

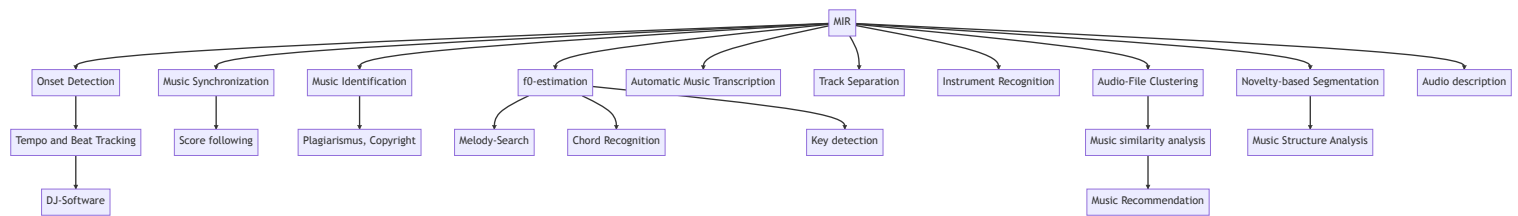
import matplotlib.pyplot as plt
media = [1, 2, 3, 4, 5, 6, 7]
media = ['Radio', 'Spotify', 'YouTube', 'Files', 'Bandcamp', 'physical', 'iTunes']
usage = [1.5, 2, 1.5, 1.5, 2, 2.5, 1]

plt.bar(media, usage, align='center')
plt.xticks(rotation=45, ha='right')
plt.show()
```



*"Streaming continued to be the biggest driver of growth with record levels of engagement in paid subscriptions, continued growth in ad-supported format revenues, and growing contributions from new platforms and services. Streaming grew 7% to \$13.3 billion, and comprised 84% of revenues in 2022." (2022 Year-End Music Industry Revenue Report | RIAA, n.d.)*

# Anwendungen



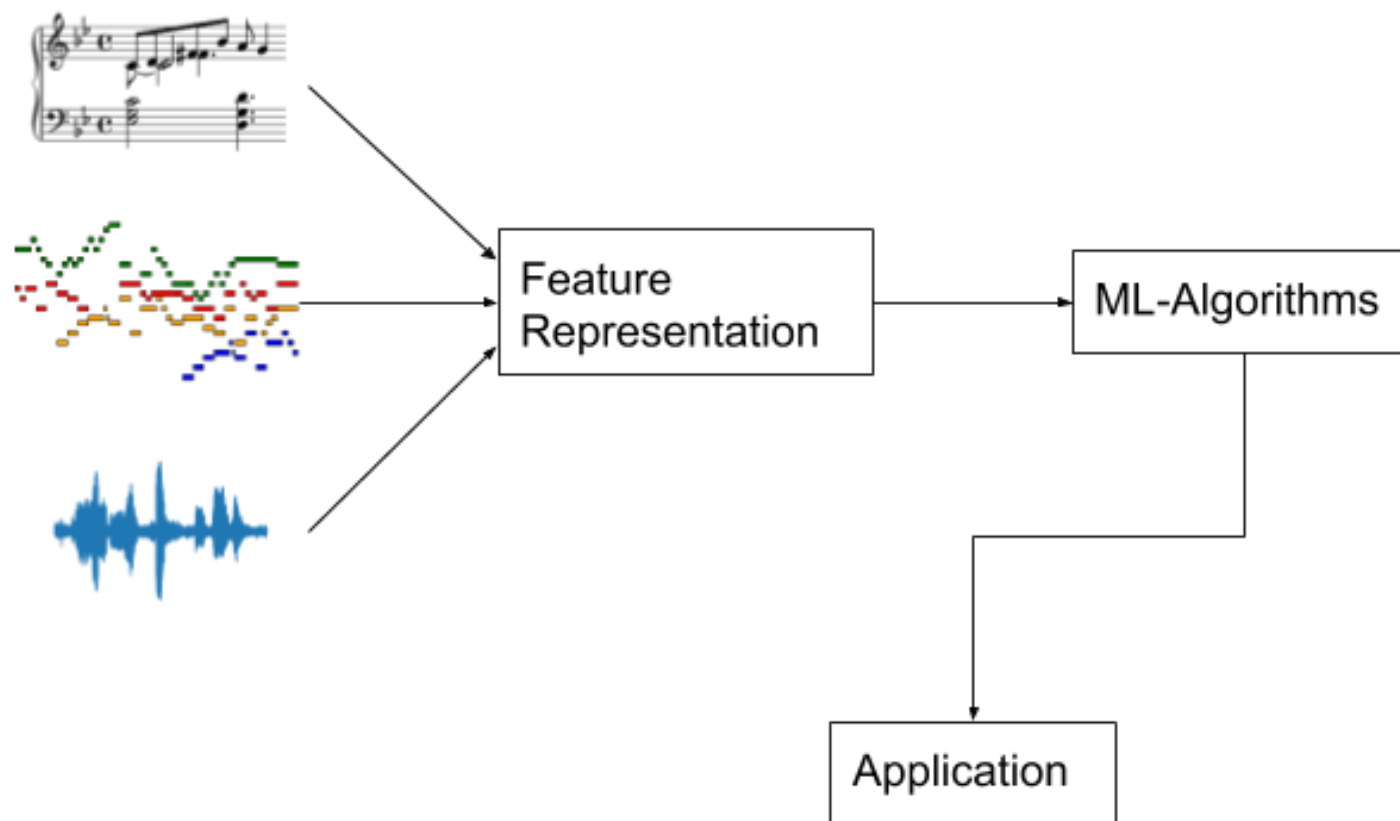
# Anwendungen



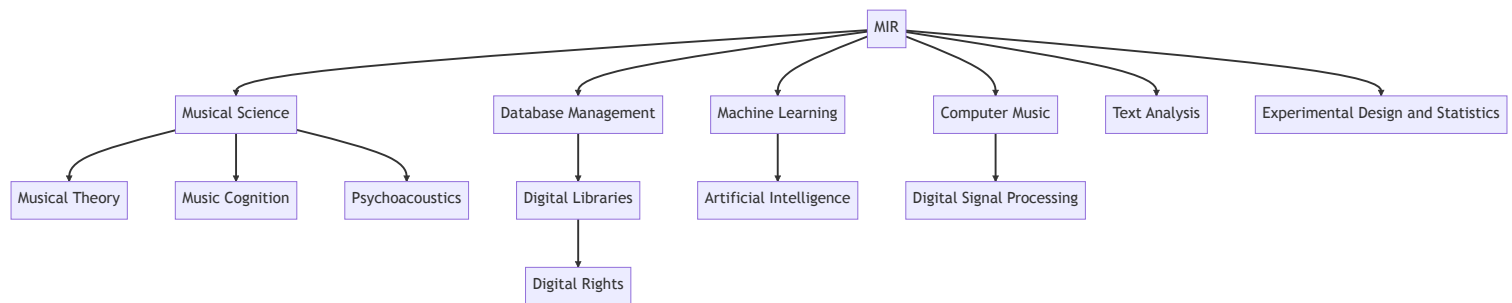
Anwendungen



# Methoden



# Domänen





# Tools

- Jupyter Notebooks
- moodle? <-> GitHub

# Ausblick

- Jupyter Notebook Installation
- Python Einführung
- Numpy Einführung
- Sinus Generieren
- Audio-Dateien lesen
- Time-Framing / Windowing
- Fourier Transformation (komplexe Zahlen, eulersche Zahl / e-Funktion --> Kreisbewegung)
- (Kurz-Zeit-) Spektrogramm
- Audio-Features
- Anwendungen (vereinfachte Versionen)
  - Chord-Recognition
  - Track Separation (Noise <-> Stochastik)
  - Novelty-based Segmentation
  - (Acoustic Environment Analysis)
  - Automatic Music Transcription
  - Onset Detection
  - Clustering / Classification
  - Harmonic Percussive Source Separation