Music Recommenders

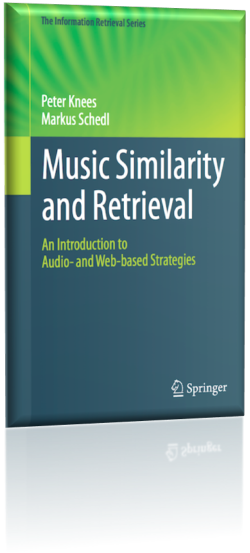
September 10th 2019

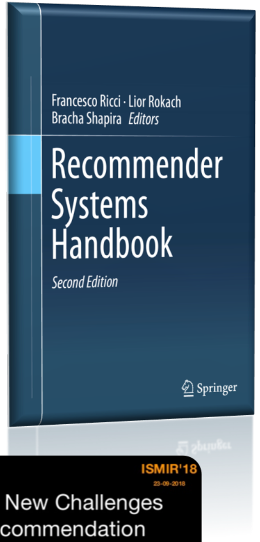
Peter Knees
peter.knees@tuwien.ac.at

TU WIEN Informatics

Outline

Sources





Course Material:
https://www.ifs.tuwien.ac.at/~knees/teaching/rsss2019/

Music Similarity and Retrieval:
An Introduction to Audio and Web-based Strategies
by P. Knees and M. Schedl. Springer, 2016.

Recommender Systems Handbook (2nd ed.)
Chapter 13: Music Recommender Systems
by M. Schedl, P. Knees, B. McFee, D. Bogdanov, M. Kaminskas. Springer, 2015.

Overview and New Challenges of Music Recommendation Research in 2018
Tutorial
by M. Schedl, P. Knees, F. Gouyon. ISMIR'18.

Intro

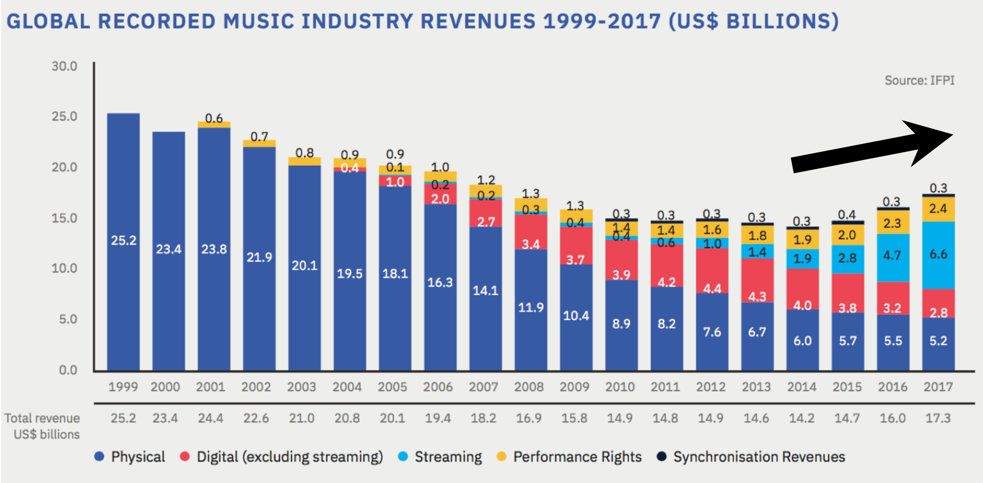
What's special to music recommendation?

- More and more relevant to the Music Industry with **rise of streaming**
- Wide range of duration of items (2+ vs. 90+ minutes),
  Lower commitment, items more "disposable", **low item cost**
  → "bad" recommendations maybe not as severe
- Magnitude of available data items (Millions) & data points (Billions)
- **Diversity of modalities** (audio, user feedback, text, etc.)
- **Various types of items** to recommend (songs, albums, artists, audio samples, concerts, venues, fans, etc.)
- Recommendations relevant for various actors (listeners, producers, performers, etc.), **multistakeholder** recommendation scenarios

What's special to music recommendation?

- Very often consumed in **sequence**
- **Re-recommendation** often appreciated (in contrast to e.g. movies)
- Often consumed passively (while working, background music, etc.)
- Yet, highly emotionally connoted (in contrast to products, e.g. home appliances)
- Different consumption locations/settings: static (e.g., via stereo at home) vs. variable (e.g., via headphones during exercise), alone vs. in group, etc.
- **Listener intent and context** are crucial
- Importance of social component
- Music often used for self-expression

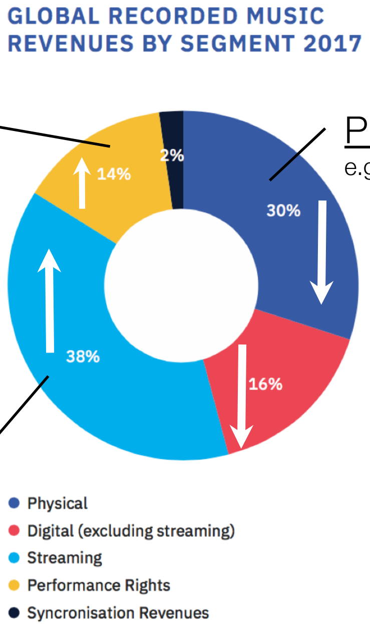
Music Consumption



PERFORMANCE RIGHTS

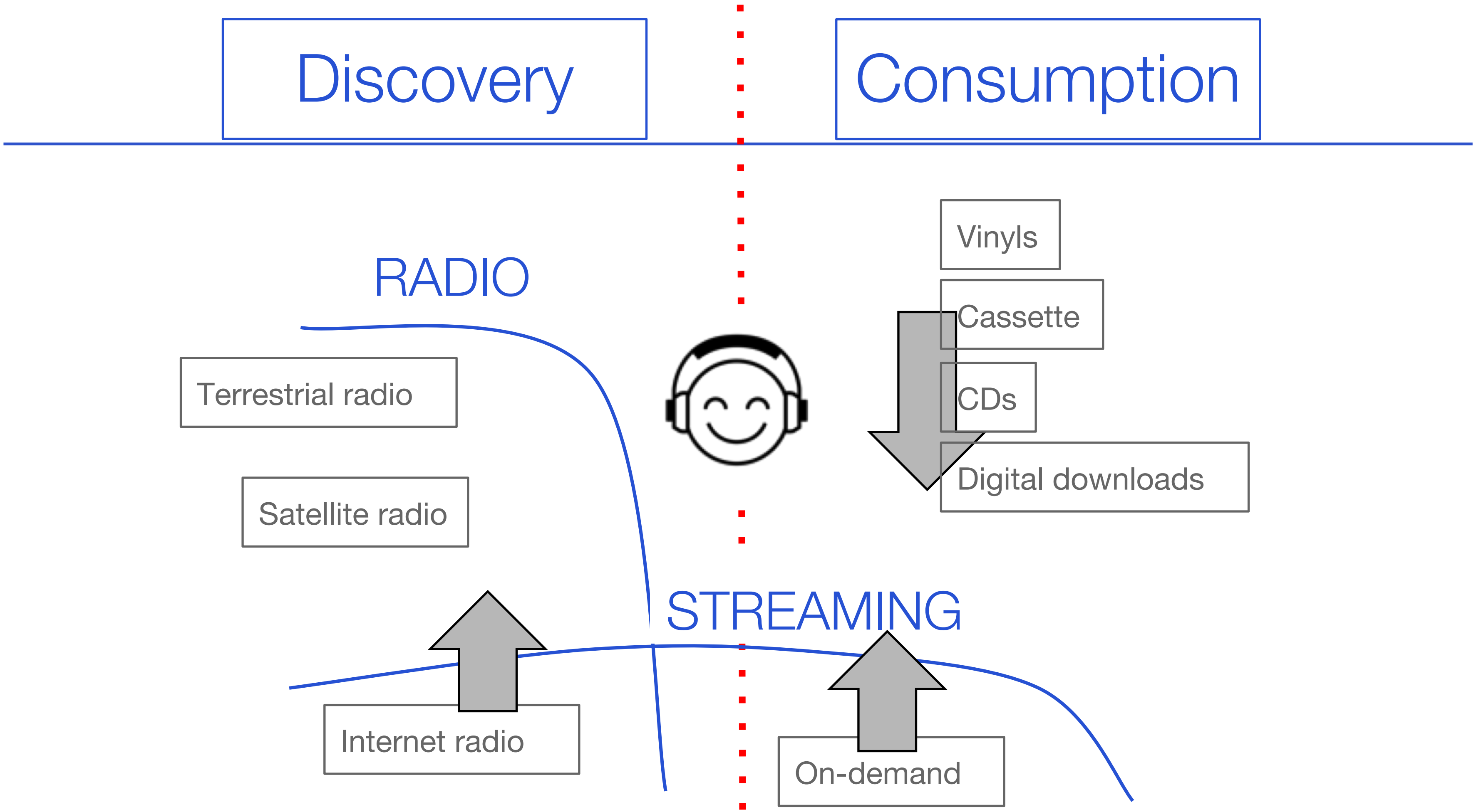
Revenue from music reproduction:
- on AM/FM radio
- at public venues

(NB: Excluding perf. rights from Streaming)



PHYSICAL
e.g. CDs

STREAMING
- Internet radio & on-demand
- Ad-supported & subscriptions


Discovery
Consumption
RADIO
Terrestrial radio
Satellite radio
Internet radio
Vinyls
Cassette
CDs
Digital downloads
STREAMING
On-demand

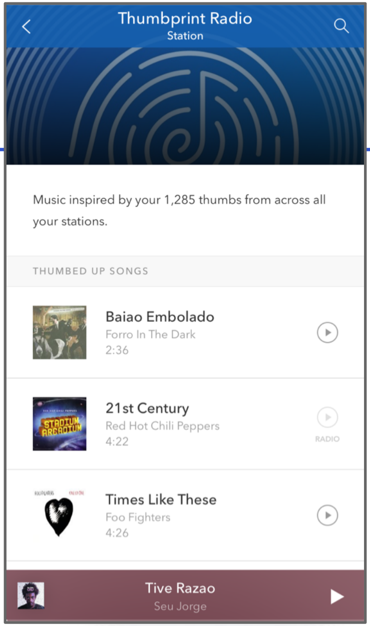
Music Discovery

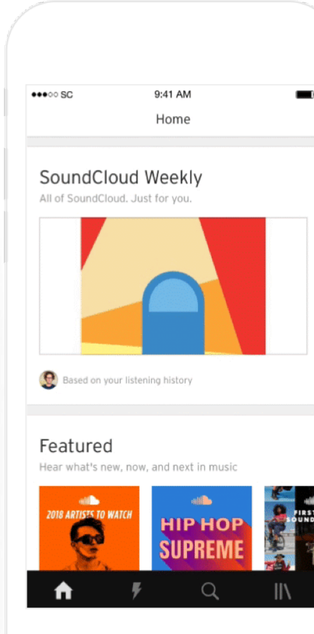
- Streaming “taking over” physical & downloads
- But competing with terrestrial radio, too

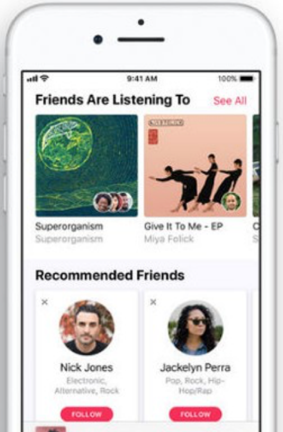
The Quest for “Discovery”

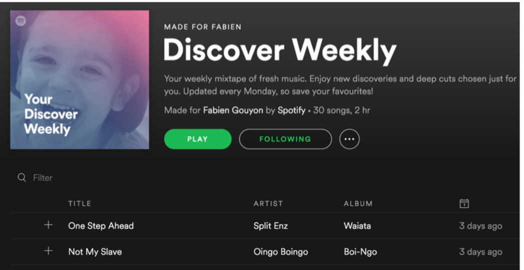
Ongoing quest for defining listening format calls for:

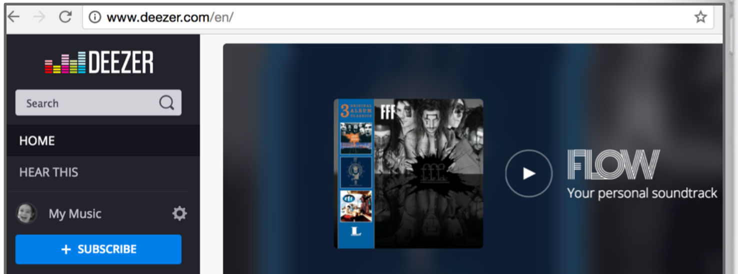
- Innovative Discovery features
- Right balance between lean-in & lean-back experiences







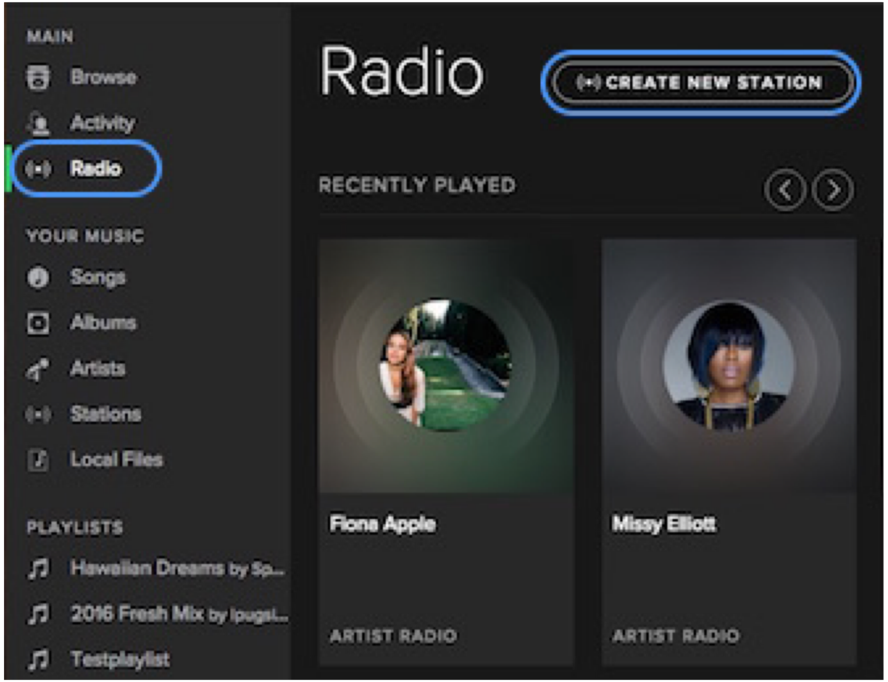




Part I: Challenges in Building a Real-World Music Recommender

(using material by Fabien Gouyon, Pandora)

Automatic Playlists/Radio Stations



spotify.com

- Personalized radio stations, e.g.
  - Spotify radio
  - Apple Music
  - YouTube Music
  - Deezer
  - Pandora
  - Last.fm
- Continuously plays similar music
- Based on content and/or collaborative filtering
- Optionally, songs can be rated for improved personalization

Automatic Radio Station Generation Problem

- A <u>continuation</u> problem
- Given a listener enjoying a particular musical experience (defined by the music itself, but also contextual factors and the listener’s intent), what recommendations can we make to **extend this experience in the best possible way** for the listener?

A “good” recommendation?

What makes a good recommendation:

- Accuracy
- Good balance of:
  - Novelty vs. familiarity / popularity
  - Diversity vs. similarity
- Transparency / Interpretability
- Listener Context

Influential factors:

- Listener
- Musical anchor
- Focus / Intent

It’s about recommending a listening experience

[Celma, 2010] *Music Recommendation and Discovery: The Long Tail, Long Fail, and Long Play in the Digital Music Space*, Springer

[Celma, Lamere, 2011] *Music Recommendation and Discovery Revisited*, ACM Conference on Recommender Systems

[Jannach, Adomavicius, 2016] *Recommendations with a Purpose*, RecSys

[Amatriain, Basilico, 2016] *Past, Present, and Future of Recommender Systems: An Industry Perspective*, RecSys

Accuracy (is not enough)

- Typically, recommendations are based on predicting the relevance of unseen items to users. Or on item ranking.
- For recommendations to be accurate, optimize to best predict general relevance
  - e.g. optimizing on historical data from all *users*

- Too much focus on accuracy → biases (i.e. **popularity** and **similarity** biases)
  - Tradeoff popularity vs. personalization (is pleasing both general user base *and* each individual even possible?...)
  - Particular risk of selection bias when RecSys is the oracle (e.g. station)
  - Single-metric Netflix Prize (RMSE) → only one side of the coin

[Jannach, et al. 2016] *Biases in Automated Music Playlist Generation: A Comparison of Next-Track Recommending Techniques*, UMAP

Novelty

- Introducing novelty to balance <u>against popularity (or familiarity) bias</u>
- <u>Both</u> are key: Listeners want to hear what's hype (or what they already know).
  But they also need their dose of novelty… Once in a while.
  - How far novel? ("correct" dose?)
  - How often?
  - When?, etc…

	"Yep, novelty's fine"	*"No novelty, please!"*
Listener	Jazz musician	My mother
Musical anchor ("query")	Exploring a new friend's music library	Playlist for an official high-stake dinner
Focus	Discovery	Craving for my hyper-personalized stuff

Diversity

- Introducing diversity to balance against similarity bias
- Similarity $\cong$ accuracy
  - Trade-off accuracy vs. diversity
  - As for Novelty, adding Diversity is a useful means for personalizing and contextualizing recommendations

	"Yep, bring on diversity"	*"No diversity, please!"*
Listener	A (good) DJ	Exclusive Metal-head
Musical anchor ("query")	Station anchored on "90's & 00's Hits"	Self-made playlist anchored on "Slayer"
Focus	Re-discovery, hyper-personalized	"Women in Post-Black Metal"

[Parambath, Usunier, Grandvalet, 2016] *A Coverage-Based Approach to Recommendation Diversity on Similarity Graph*, RecSys

Exploration vs. Exploitation

- Exploit:
  - **Data** tells us what works best now, let's play exactly that
  - Play something **safe now**, don't worry about the future
  - Lean-back experience
  - "Don't play music I am not familiar with"

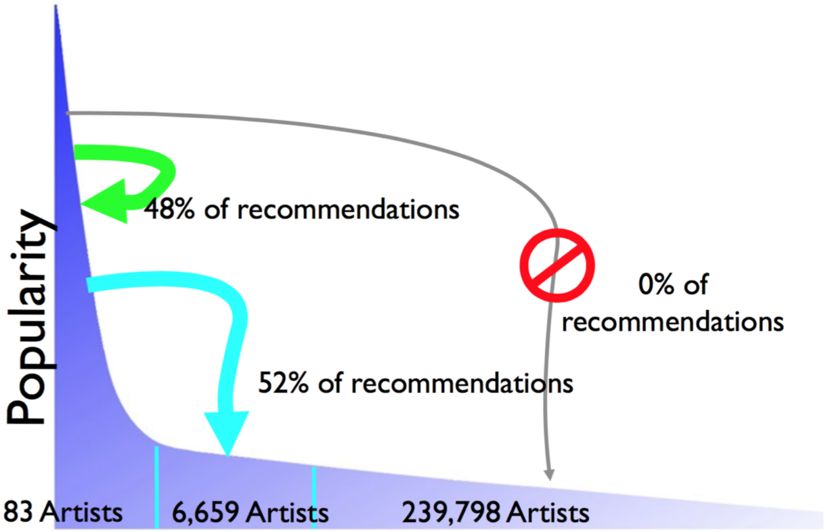
Short-term reward

Long-term reward

- Explore:
  - Let's **learn** (i.e. gather some more data points on) what **might** work
  - Play something **risky now**, preparing for tomorrow
  - Lean-in experience
  - "I'm ready to open up. Just don't play random stuff"

[Xing, Wang, Wang, 2014] *Enhancing Collaborative Filtering Music Recommendation by Balancing Exploration and Exploitation*, ISMIR

Exploration vs. Exploitation



(CF-based recommendations, Last.fm data)

Helps alleviate limited reach of some recsys:

- Coldplay, Drake, etc. vs. “Working-class” musicians (long-tail)
- Radio typically plays 10’s artists per week
- Streaming has the potential to play 100k’s artists per week
- Caveat of collaborative filtering-based algorithms

[Celma, 2010] *Music Recommendation and Discovery: The Long Tail, Long Fail, and Long Play in the Digital Music Space*, Springer

Transparency / Interpretability

- *“Why am I recommended this?”*

If you like Bernard Herrmann

You might like “Gimme some more” by Busta Rhymes

Transparency / Interpretability

- *“Why am I recommended this?”*

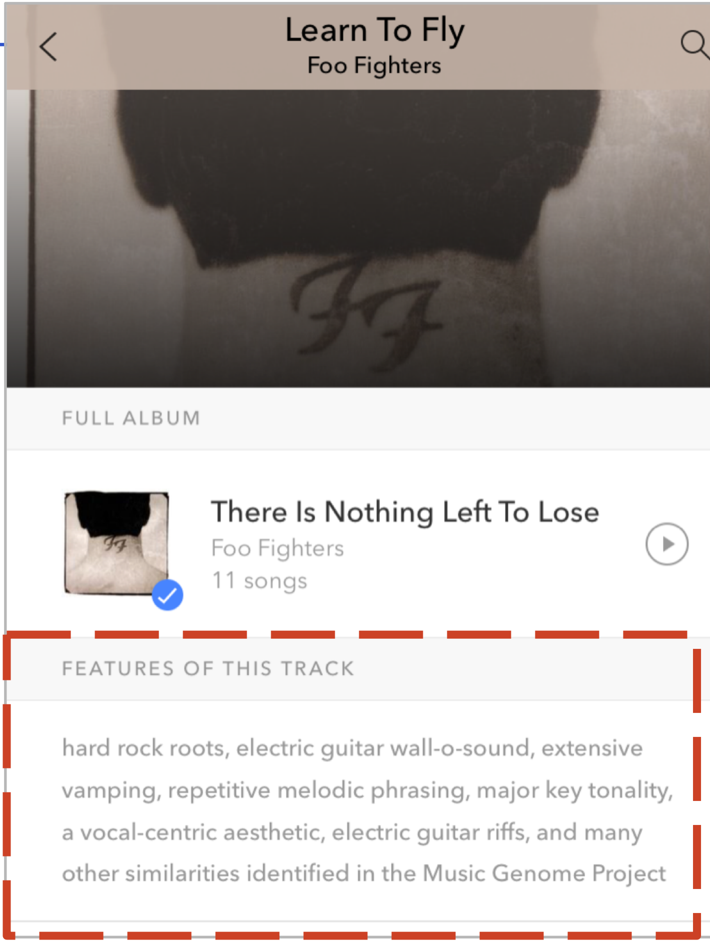
If you like Bernard Herrmann

You might like “Gimme some more” by Busta Rhymes

Because:
He sampled Herrmann’s work

Transparency / Interpretability

- Explain how the system works: transparency
- Increases users' confidence in the system: trust
- Facilitates persuasion
- Fun factor → increases time spent listening
- Increases personalization
  (e.g. *"because you like guitar"*)
- Better experience overall
- Caveat: Users will then want to correct potentially erroneous assumptions
  → Extra level of interactivity needed



[Tintarev, Masthoff, 2015] *Explaining Recommendations: Design and Evaluation*, Recommender Systems Handbook (2nd ed.), Kantor, Ricci, Rokach, Shapira (eds), Springer

[Musto, Narducci, Lops, de Gemmis, Semeraro, 2016] *ExpLOD: A Framework for Explaining Recommendations based on the Linked Open Data Cloud*, RecSys

[Chang, Harper, Terveen, 2016] *Crowd-based Personalized Natural Language Explanations for Recommendations*, RecSys

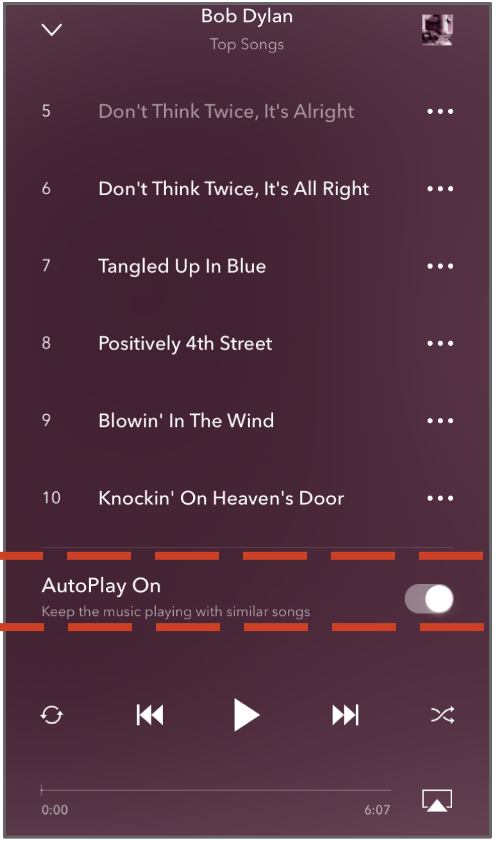
Listener Context

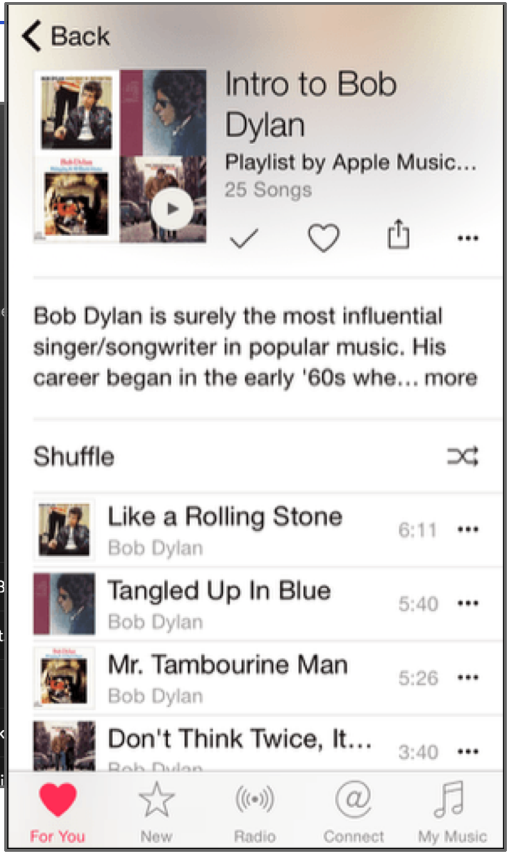
- Special case of **explicit listener focus/ listener intent**, e.g.:
  - Focus on newly released music (new stuff)
  - Focus on activity (e.g. workout)
  - Focus on discovery (new *for me*)
  - On re-discovery (throwback songs)
  - Hyper-personalized (extreme lean-back, *my* best-of)
  - etc.

→ Each specific focus defines:

- Which recommendations are best?
- Which **vehicle** for recommendations is best (**HOW** to recommend)?

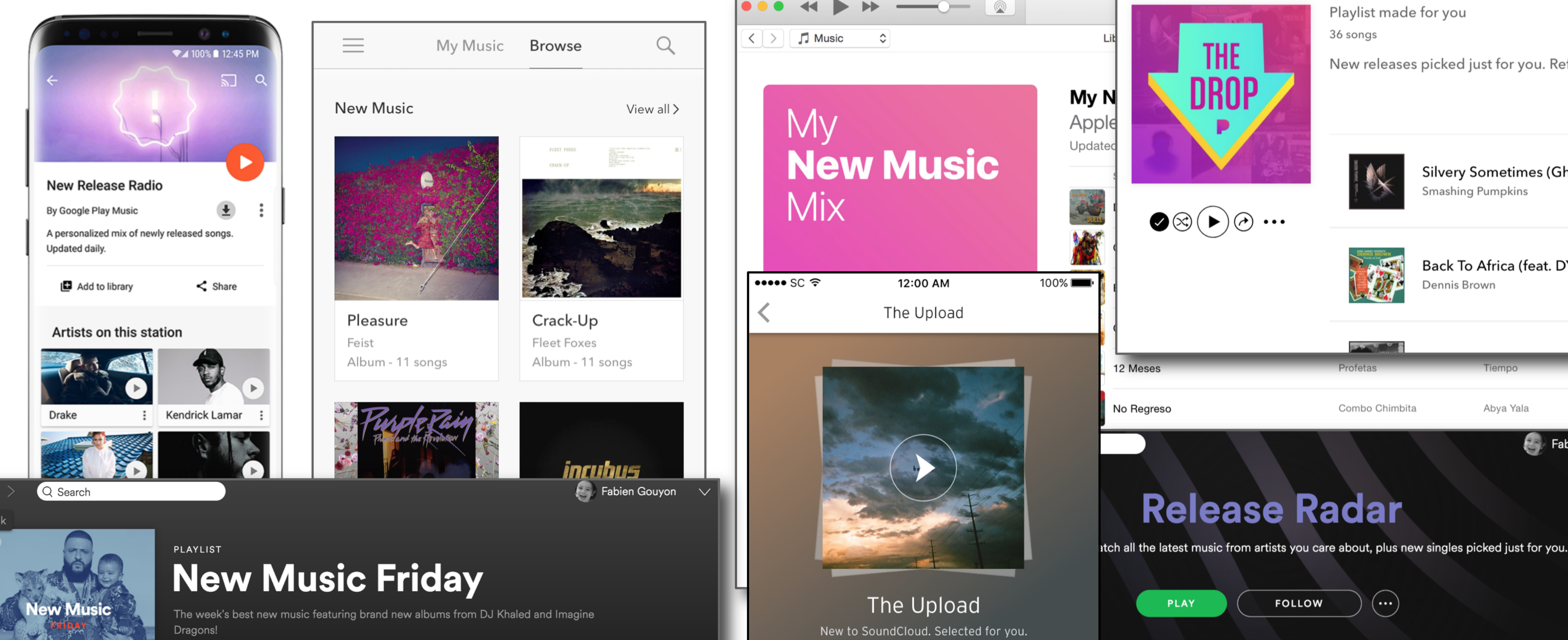
Focus on: Discovering an artist



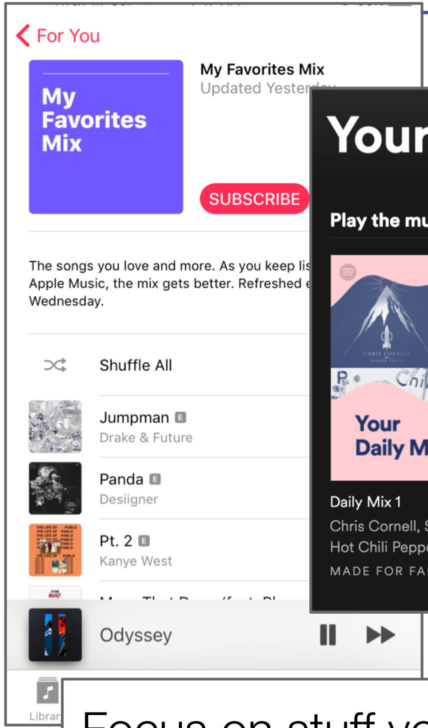


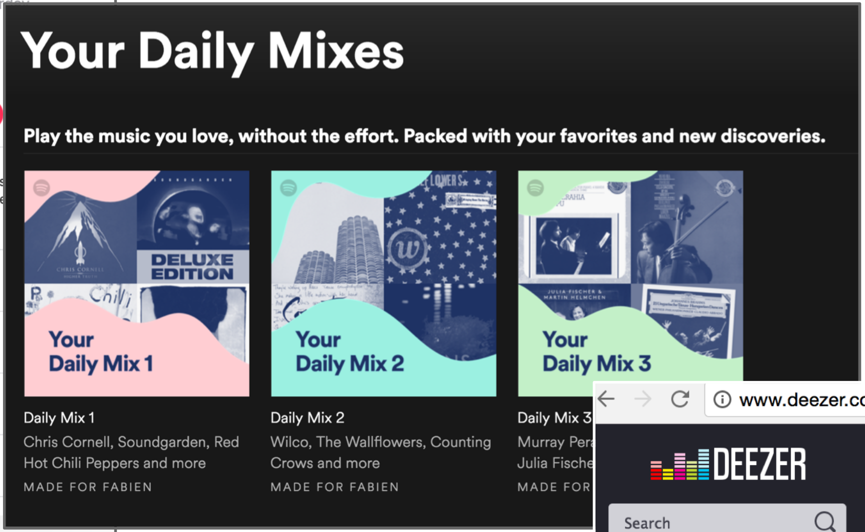
Focus on: New music

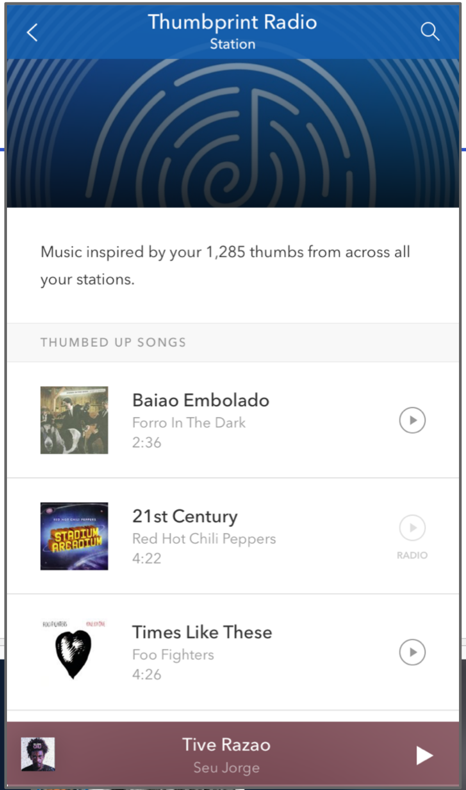
Non-personalized vs. Personalized

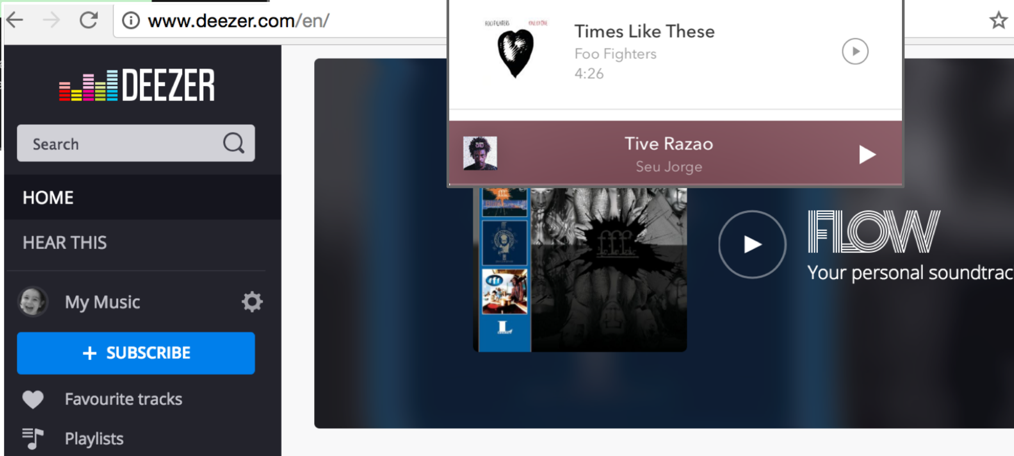


Focus on: Re-discovery



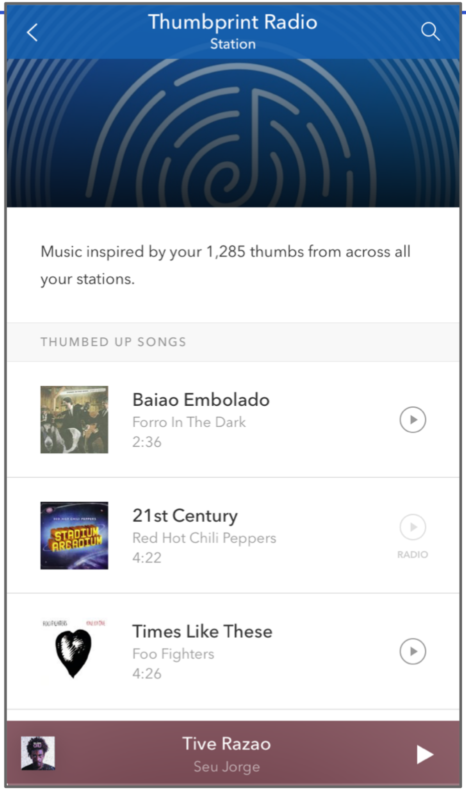


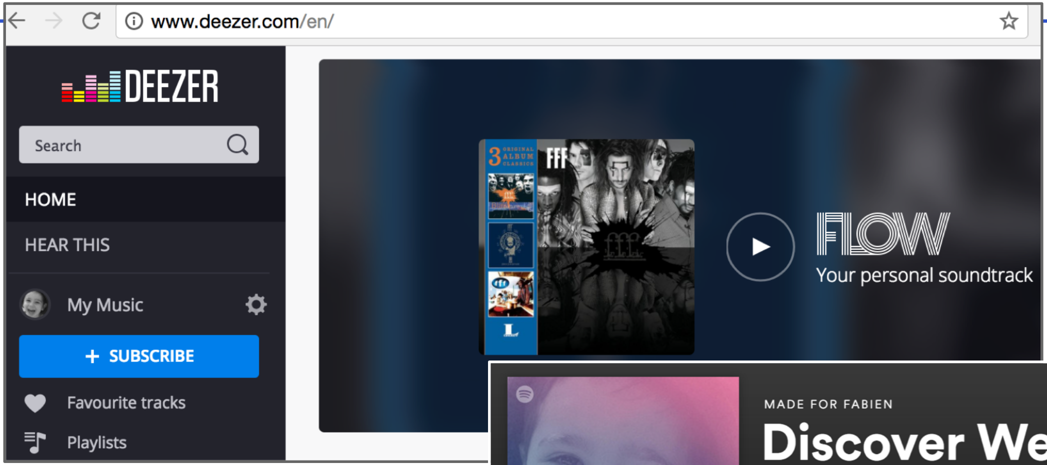




Focus on stuff you know you like
Personalized, leaning towards exploit

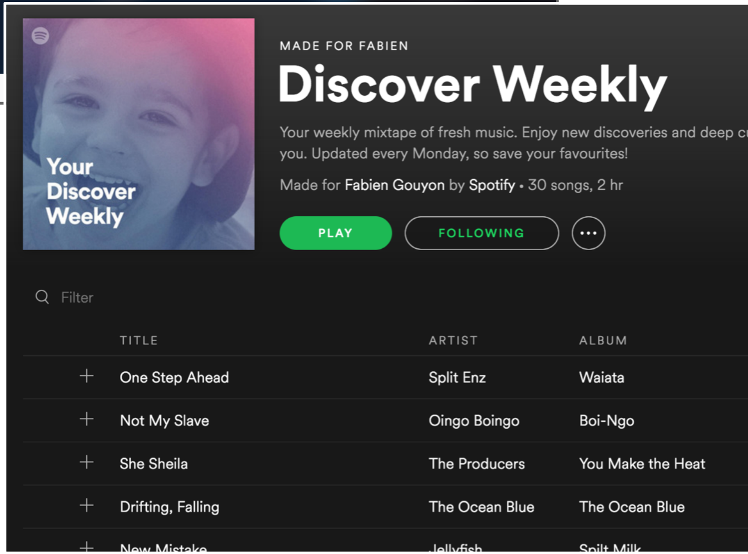
Focus on: Hyper-personalized Discovery





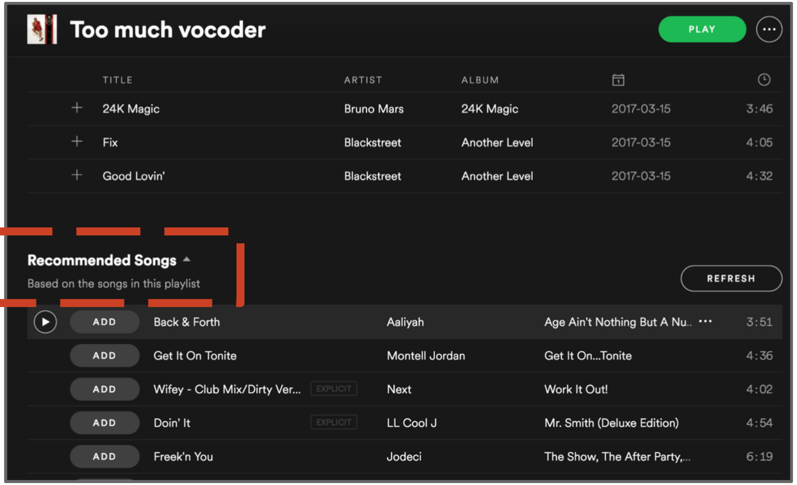
About discovering new stuff.
Intended to feel like it's curated. Just. For. Me.

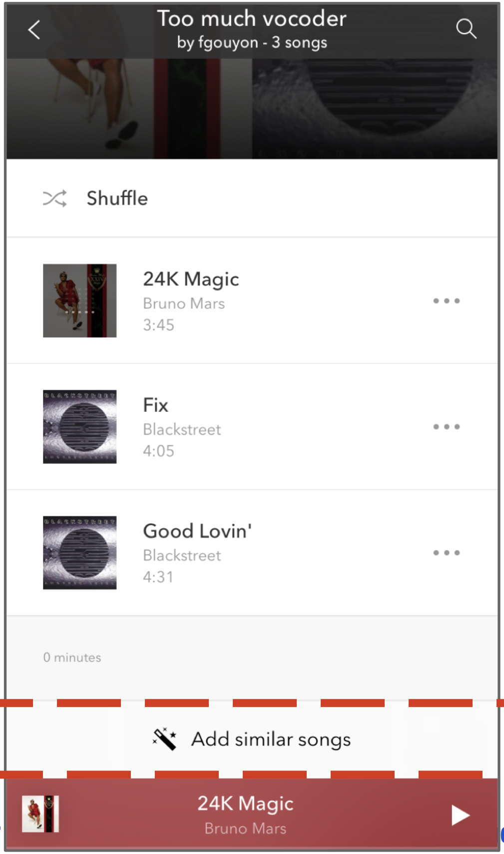
Leaning towards explore



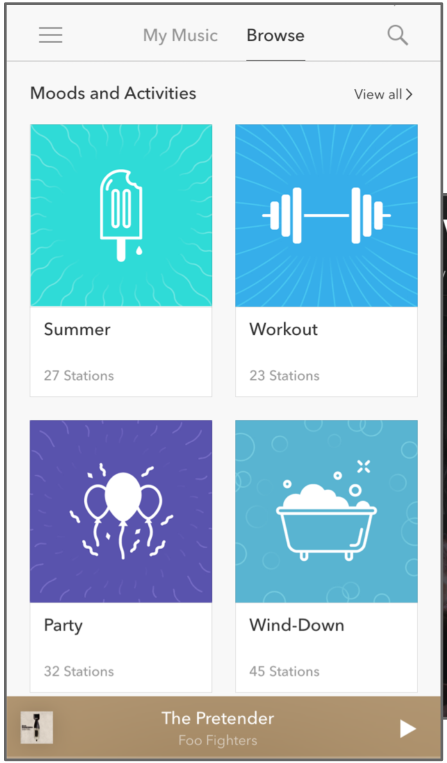
Focus on: Lean-in experience

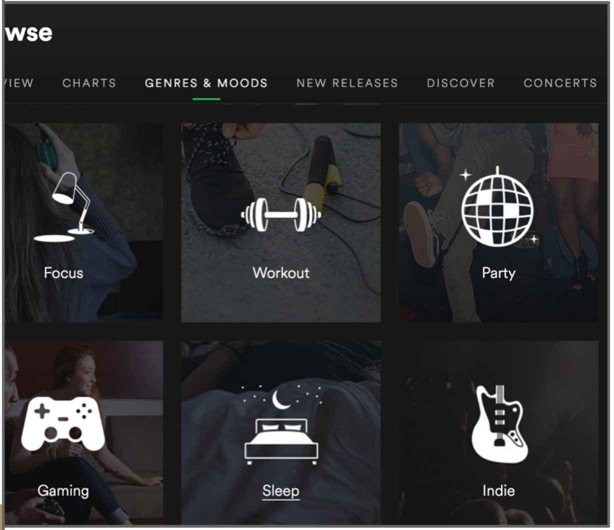
Lean in:
Building Playlists

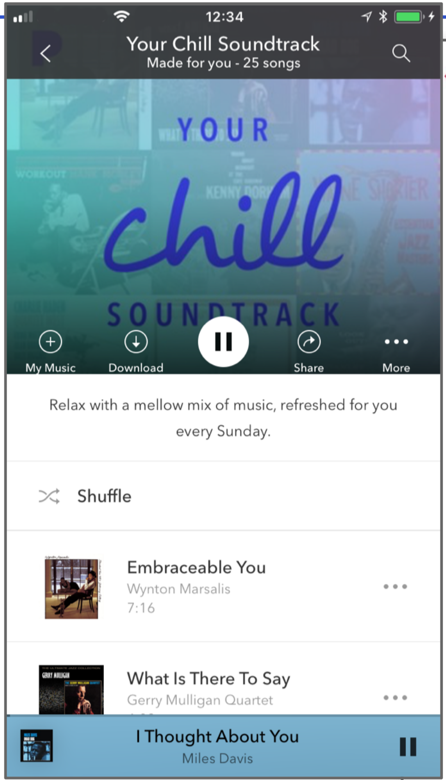


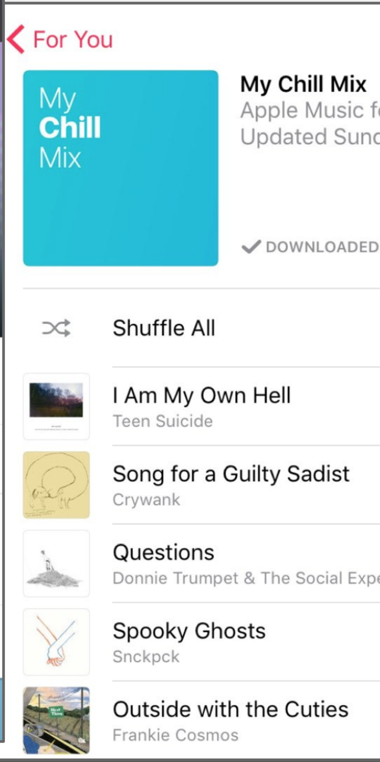


Focus on: Mood /Activity









Non-personalized vs. Personalized

Part II: Data, Algorithms, Platforms

Data fuels recommenders

Interaction Data

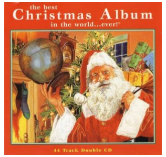
- Listening logs, listening histories
- Feedback (“thumbs”), purchases

User-generated

- Tags, reviews, stories

Curated collections

- Playlists, radio channels
- CD album compilations



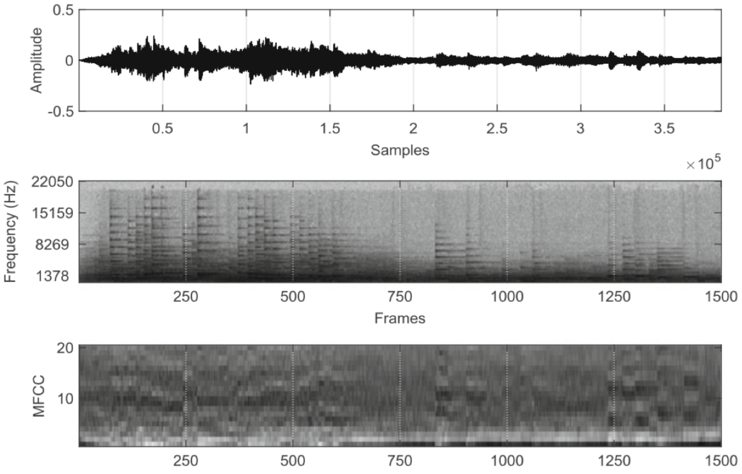
Data fuels recommenders

Content (audio, symbolic, lyrics)

- Machine listening/content analysis
- Human labelling

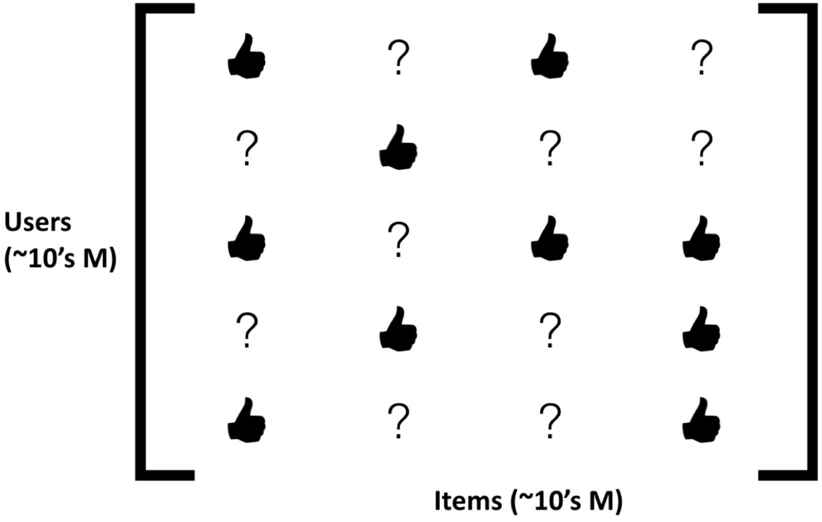
Meta-data

- Editorial
- Curatorial
- Multi-modal (album covers etc.)



Collaborative Filtering (CF)

- Exploits **interaction data**
- *"People who listened to track A, also listened to track B"*
- Main underlying assumption: users that had similar taste in the past, will have similar taste in the future
- Stemming from "usage" of music
  → close to "what users want"



Factors Hidden in the Data

Original assumption of first matrix factorization-based recommender systems:

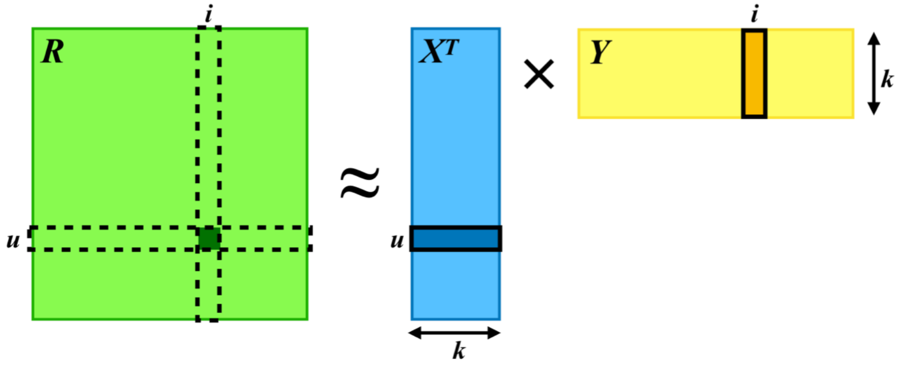
- Observed ratings/data are interactions of 2 factors: users and items
- Latent factors are representation of users and items

Matrix Factorization (cf. SVD)

- Decompose rating matrix into user and item matrices of lower dimension k
- Learning factors from given ratings using stochastic gradient descent

$$\min_{x_\star, y_\star} \sum_{r_{u,i} \text{ is known}} (r_{ui} - x_u^T y_i)^2 + \lambda(\|x_u\|^2 + \|y_i\|^2)$$

- Prediction of rating: inner product of vectors of user u and item i
- Factors not necessarily interpretable (just capture variance in data)



[Funk/Webb, 2006] *Netflix Update: Try this at home*, http://sifter.org/~simon/journal/20061211.html
[Koren et al., 2009] *Matrix Factorization Techniques for Recommender Systems*, Proceedings of the IEEE.

Matrix Factorization for Music Recommendation

- For music, variants deal with specifics in data, e.g.,
- Learning factors and biases using hierarchies and relations in data
  cf. [Koenigstein et al. 2011]

$$b_{ui} = \mu + b_{u,type(i)} + b_{u,session(i,u)} + b_i + b_{album(i)} + b_{artist(i)} + \frac{1}{|genres(i)|} \sum_{g \in genres(i)} b_g + c_i^T f(t_{ui})$$

[Koenigstein et al., 2011] *Yahoo! music recommendations: modeling music ratings with temporal dynamics and item taxonomy*, RecSys.

- Special treatment of implicit data (*preference* vs. *confidence*)

$$\min_{x_\star, y_\star} \sum_{u,i} c_{ui}(p_{ui} - x_u^T y_i)^2 + \lambda \left(\sum_u \|x_u\|^2 + \sum_i \|y_i\|^2 \right)$$

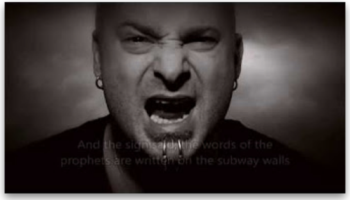
preference: $p_{ui} = \begin{cases} 1 & r_{ui} > 0 \\ 0 & r_{ui} = 0 \end{cases}$

confidence: $c_{ui} = 1 + \alpha r_{ui}$

[Hu et al., 2008] *Collaborative Filtering for Implicit Feedback Datasets*, ICDM.

Example of Collaborative Filtering Output

People who liked **Disturbed — The Sound of Silence**, also liked…



1. Bad Wolves — Zombie

2. Five Finger Death Punch — Bad Company

3. Disturbed — The Light

4. Metallica — Nothing Else Matters

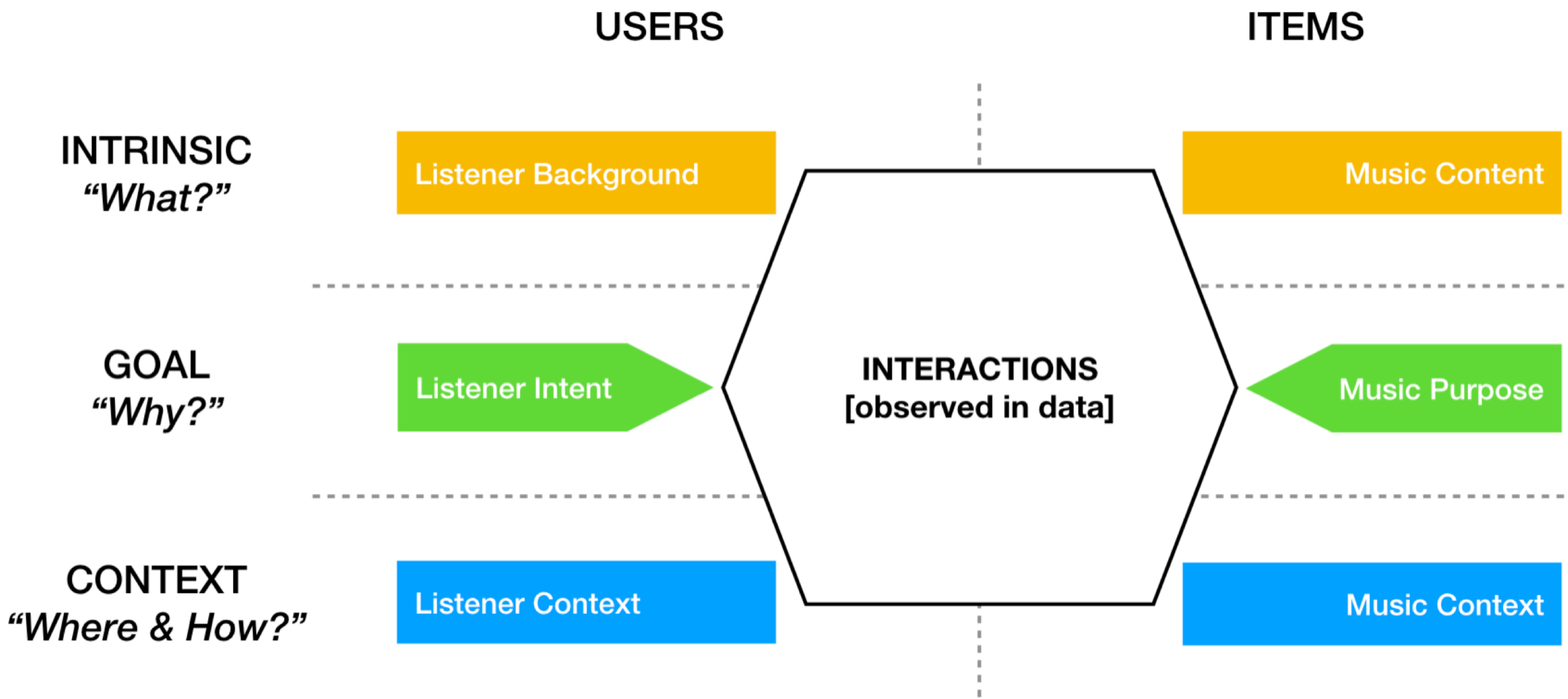
Factors Hidden in the Data

Original assumption of first matrix factorization-based recommender systems:

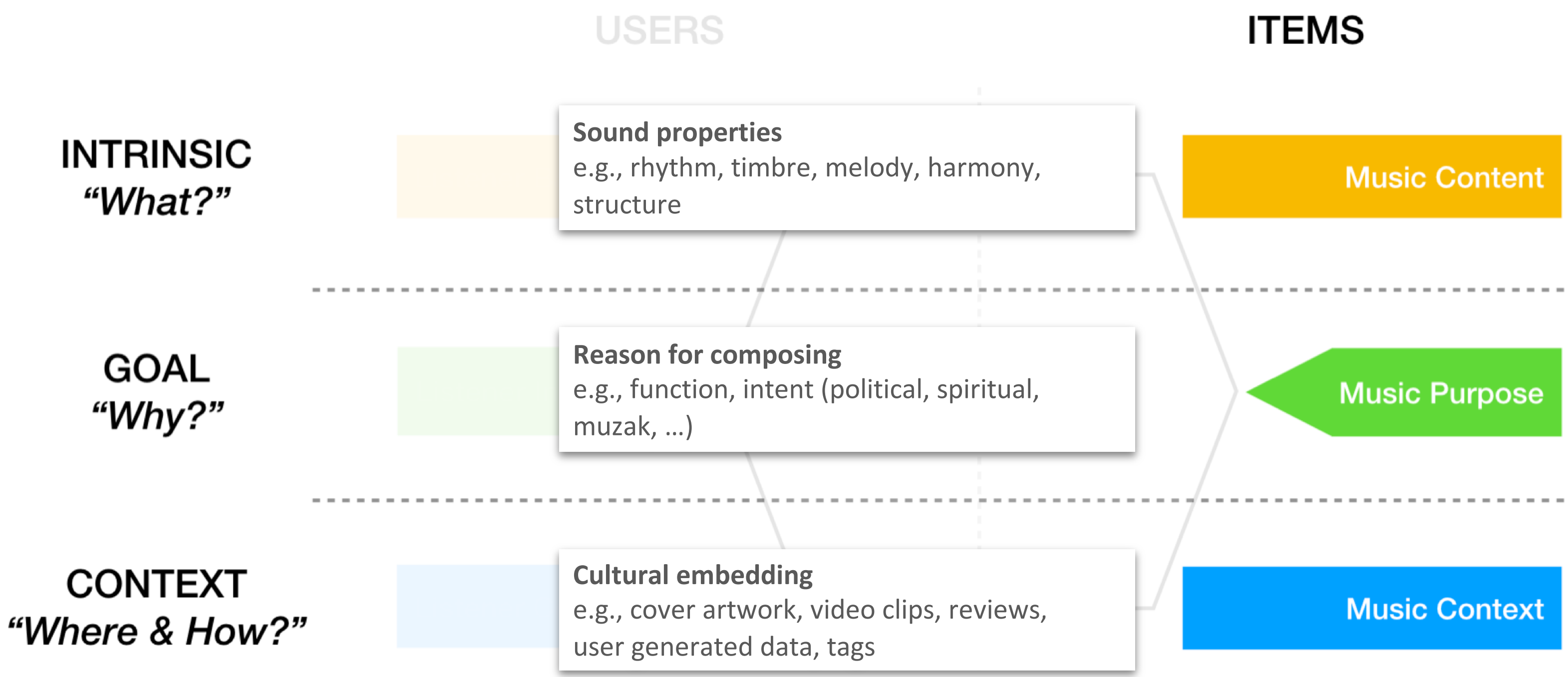
- Observed ratings/data are interactions of 2 factors: users and items
- Latent factors are representation of users and items

- But it’s a bit more complex...

Factors Hidden in the Data



Factors Hidden in the Data



Audio Content Analysis

- Features can be extracted from any audio file **NO COLD START!**
  → no other data or community necessary
  → no cultural biases (no popularity bias, no subjective ratings etc.)
- Learning of high-level semantic descriptors from low-level features via machine learning
- Deep learning now the thing
  (representation learning and temporal modeling directly from the signal, without hand-crafting features → CNNs, RNNs)
- In contrast to e.g., movies: **true content-based recommendation!**

[Choi et al., 2017] *A Tutorial on Deep Learning for Music Information Retrieval*, arXiv:1709.04396.

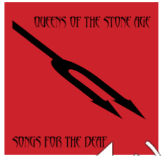
[Casey et al., 2008] *Content-based music information retrieval: Current directions and future challenges,* Proc IEEE 96 (4).

[Müller, 2015] *Fundamentals of Music Processing: Audio, Analysis, Algorithms, Applications*, Springer.

Audio Content Analysis: Selected Features

Disturbed
The Sound of Silence

- Beat/downbeat → Tempo: 85 bpm
- Timbre (→ MFCCs)
  e.g. for genre classification,
  "more-of-this" recommendations



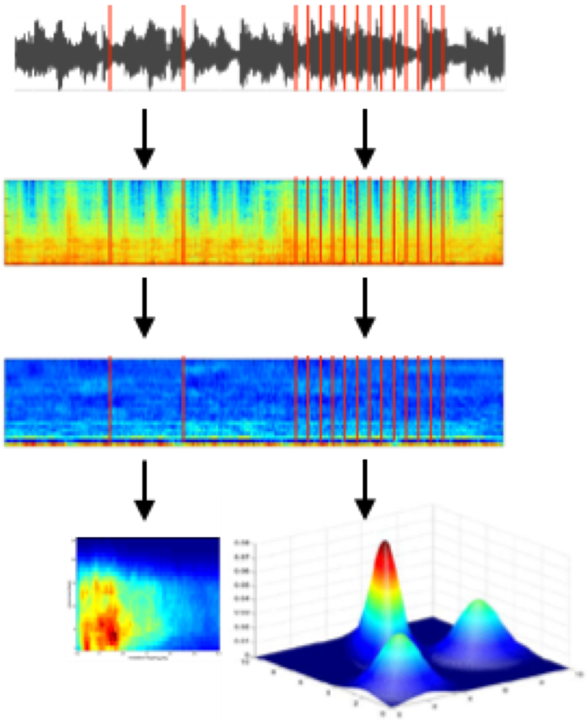
- Tonal features (→ Pitch-class profiles)
  e.g. for melody extraction,
  cover version identification

Different versions of this song:
Simon & Garfunkel - The Sound of Silence
Anni-Frid Lyngstad (ABBA) - En ton av tystnad
etc.

- Semantic categories via machine learning:
  `not_danceable, gender_male, mood_not_happy`

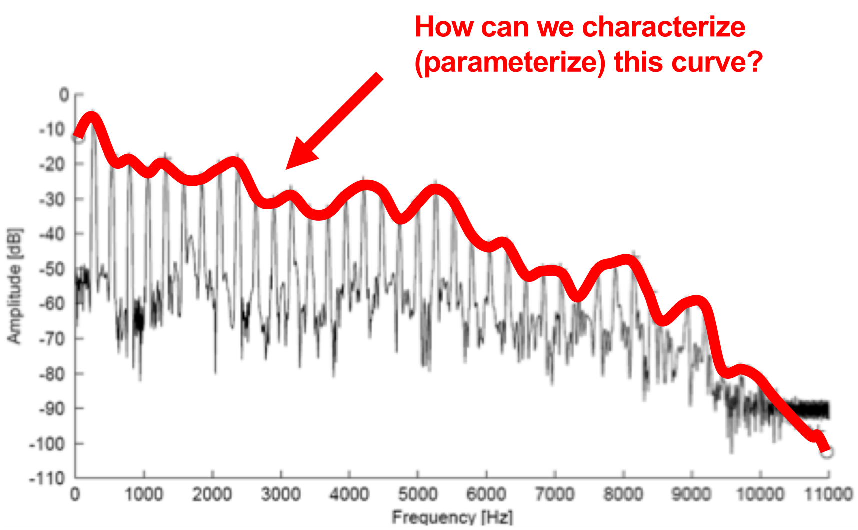
Audio Features: Basic Processing Steps

- Convert signal from time domain to *frequency domain*, e.g., using a Fast Fourier Transform (FFT)
- *Psychoacoustic transformation* (Mel-scale, Bark-scale, Cent-scale, ...): mimics human listening process (not linear, but logarithmic!), removes aspects not perceived by humans, emphasizes low frequencies
- Extract features
  - *Block-level* (large time windows, e.g., 6 sec)
  - *Frame-level* (short time windows, e.g., 25 ms) needs model distribution of frames
- Calculate similarities between feature vectors/models



MFCCs

- Mel Frequency Cepstral Coefficients (MFCCs) have their roots in speech recognition and are a way to represent the envelope of the power spectrum of an audio frame
  - the spectral envelope captures perceptually important information about the corresponding sound excerpt (*timbral aspects*)
  - sounds with similar spectral envelopes are generally perceived as "sounding similar"

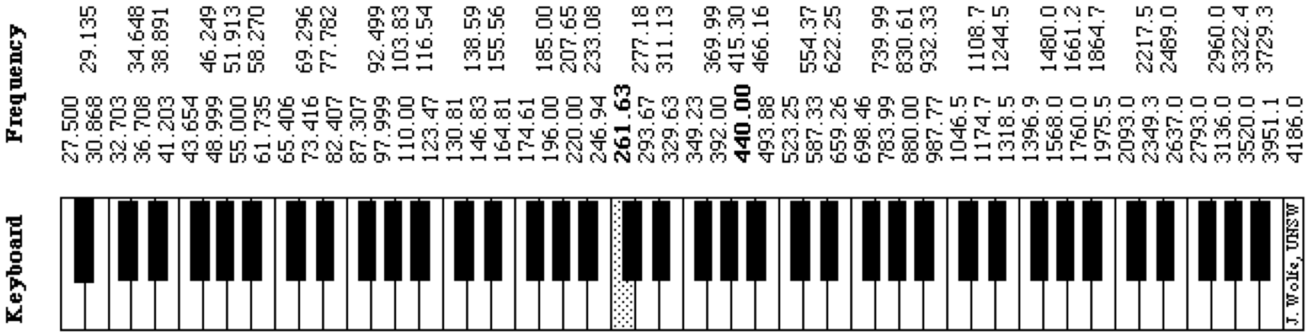


Pitch Class Profiles (aka *chroma vectors*)

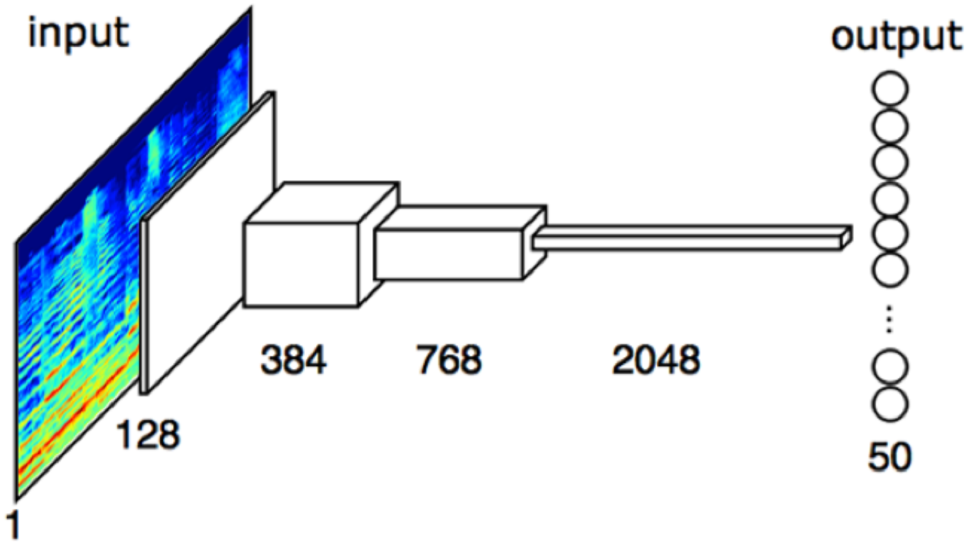
- Transforming the frequency activations into well known musical system/representation/notation

(Fujishima; 1999)

- Mapping to the equal-tempered scale (each semitone equal to one twelfth of an octave)
- For each frame, get intensity of each of the 12 semitone (pitch) classes



End-to-End Learning for Tags



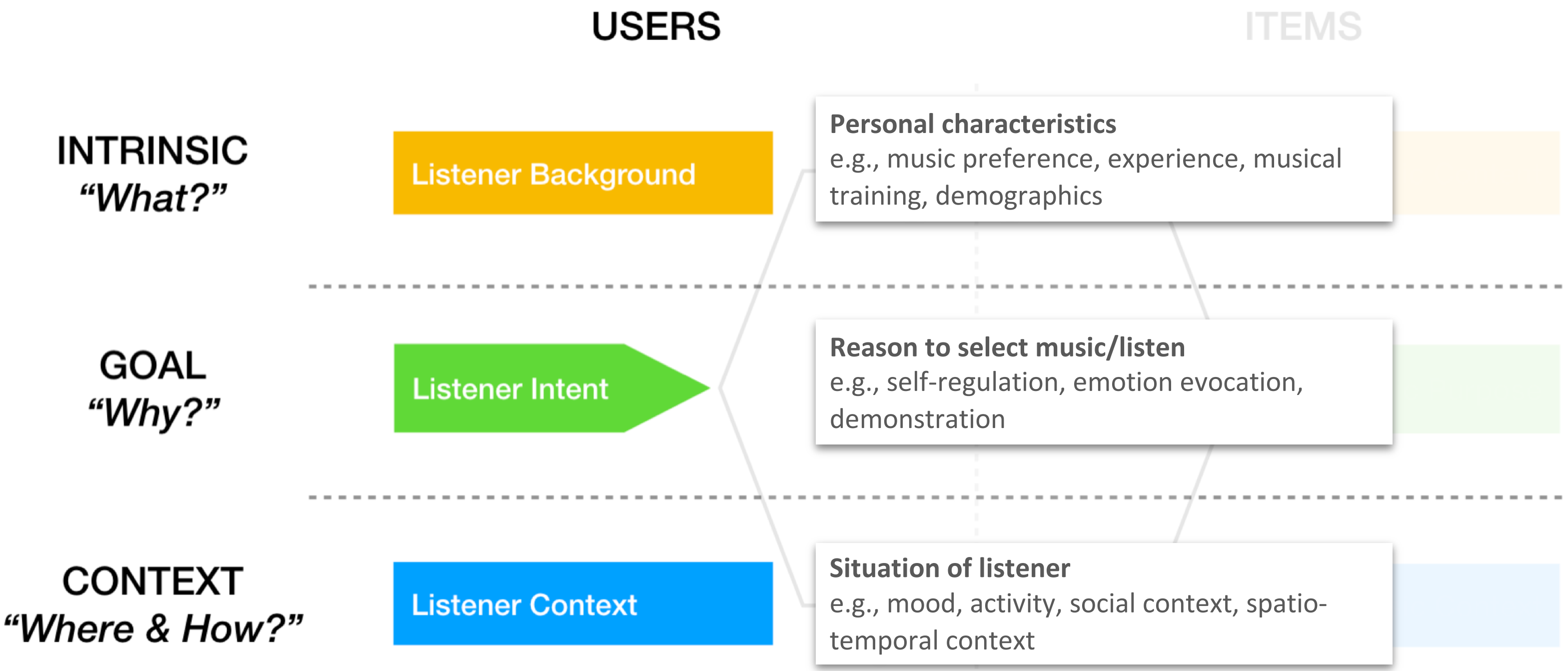
[Choi et al., 2016]

- Automatic learning of audio features for tagging with CNN
- CNN properties:
  - translation, distortion, and locality invariance
  - → musical features/events relevant to tags can appear at any time or frequency range

Practical: Toolboxes for Music Content Analysis

- Essentia (C++, Python): http://essentia.upf.edu
- Librosa (Python): https://github.com/librosa
- madmom (Python): https://github.com/CPJKU/madmom
- Marsyas (C++): http://marsyas.info
- MIRtoolbox (MATLAB): https://www.jyu.fi/hytk/fi/laitokset/mutku/en/research/materials/mirtoolbox
- jMIR (Java): http://jmir.sourceforge.net
- Sonic Visualiser (MIR through VAMP plugins): http://sonicvisualiser.org

Factors Hidden in the Data



Listener Background

- Psychology- and sociology research driven area
- Goals: more predictive user models; dealing with user cold start
- Gathering information on **user personality, music preference, demographics, cultural context**, etc. (e.g., via questionnaires or predicted via other source)

Some findings: • age (taste becomes more stable);
• when sad: *open* & *agreeable persons* want happy, *introverts* sad music;
• *individualist cultures* show higher music diversity; etc.

[Rentfrow, 2012] *The role of music in everyday life: Current directions in the social psychology of music*. Social and personality psychology compass, 6(5).
[Laplante, 2015] *Improving Music Recommender Systems: What Can We Learn From Research On Music Tastes?*, ISMIR.
[Ferwerda et al., 2015] *Personality & Emotional States: Understanding Users' Music Listening Needs*. Ext. Proc UMAP.
[Ferwerda et al., 2016] *Exploring music diversity needs across countries*. UMAP.

Listener Context

- **Context categories and acquisition:** various dimensions of the user context, e.g., time, location, activity, weather, social context, personality, etc.

Environment-related context

- Exists irrespective of a particular user
- Ex.: time, location, weather, traffic conditions, noise, light

User-related context/background

- Is connected to an individual user
- Ex.: activity, emotion, personality, social and cultural context

[Schedl et al., 2015] ch. *Music Recommender Systems*, Recommender Systems Handbook, Ricci et al. (eds.), 2nd ed.

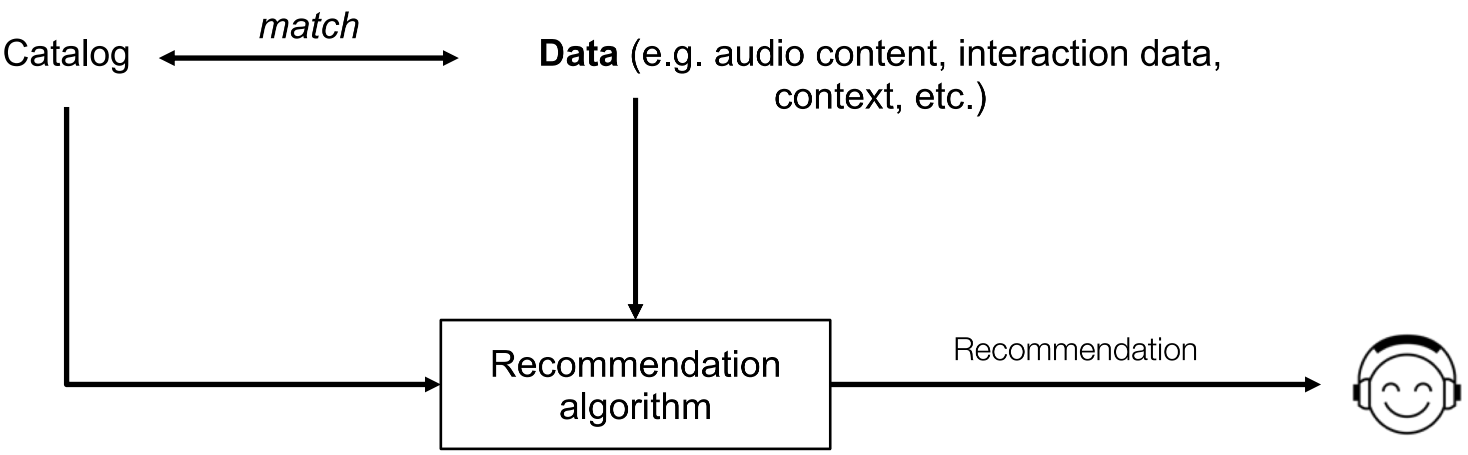
[Bauer & Novotny, 2017] *A consolidated view of context for intelligent systems*. Journal of Ambient Intelligence and Smart Environments 9(4).

Obtaining context data

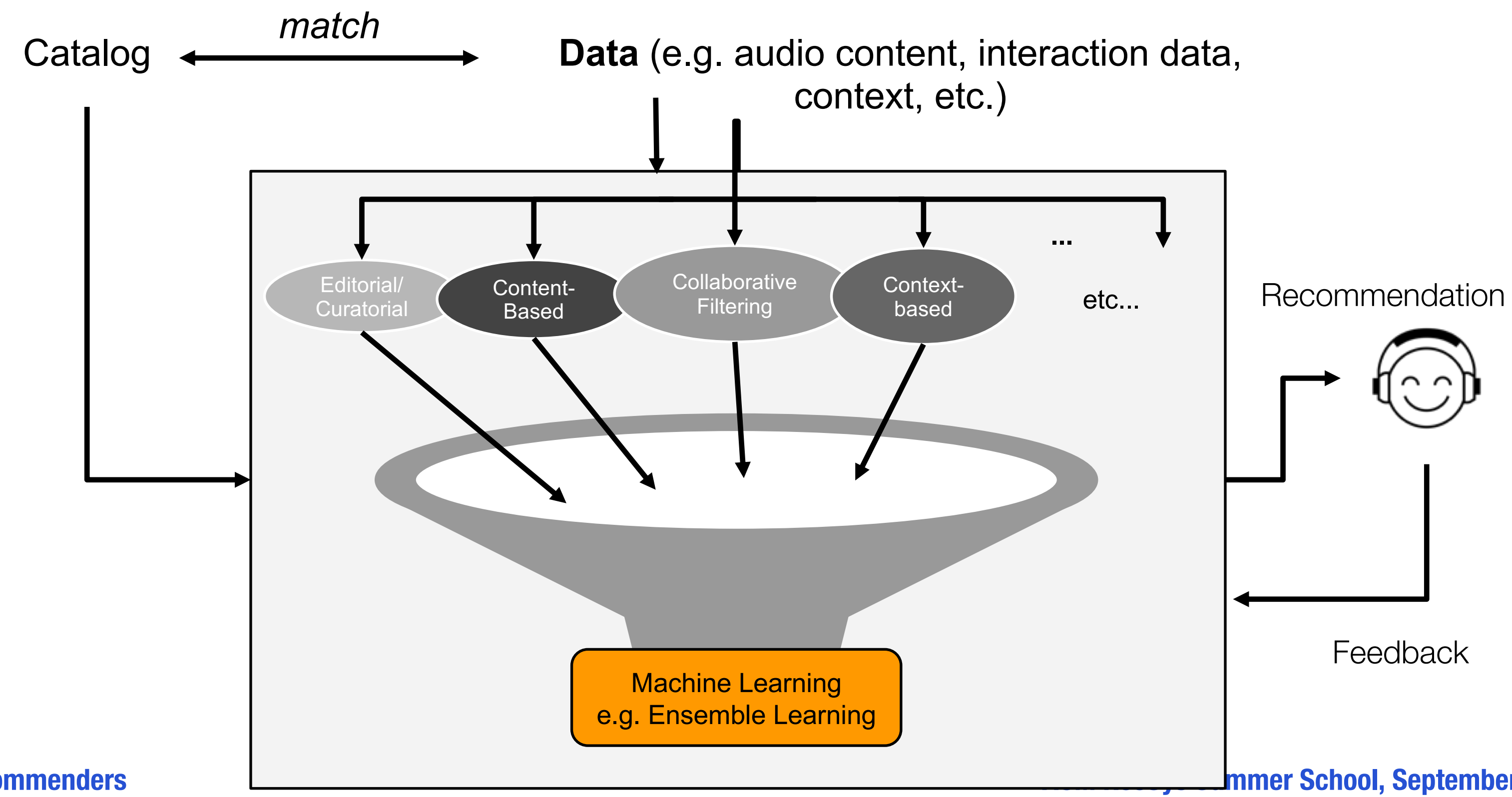
- **Explicitly**: elicited by direct user interaction (questions, ratings in context)
  Ex.: asking for user's mood or music preference (Likert-style ratings)
- **Implicitly**: no user interaction necessary
  Ex.: various sensor data in today's smart devices (heart rate, accelerometer, air pressure, light intensity, environmental noise level, etc.)
- **Inferring** (using rules or ML techniques):
  Ex.: time, position → weather; device acceleration (x, y, z axes), change in position/movement speed → activity; skipping behavior → music preferences

[Adomavicius & Tuzhilin, 2015] ch. *Context-Aware Recommender Systems*, Recommender Systems Handbook, Ricci et al. (eds.), 2nd ed.

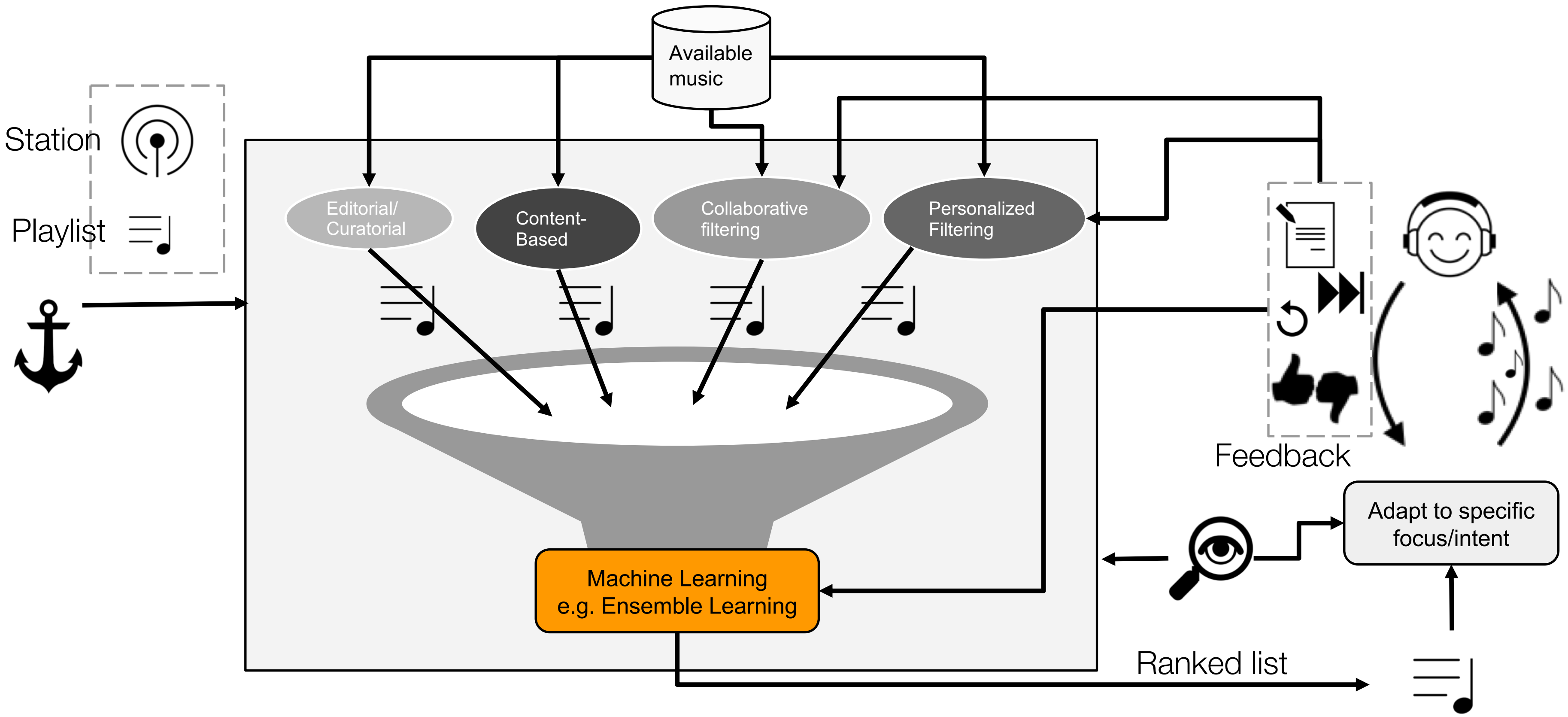
Putting it together



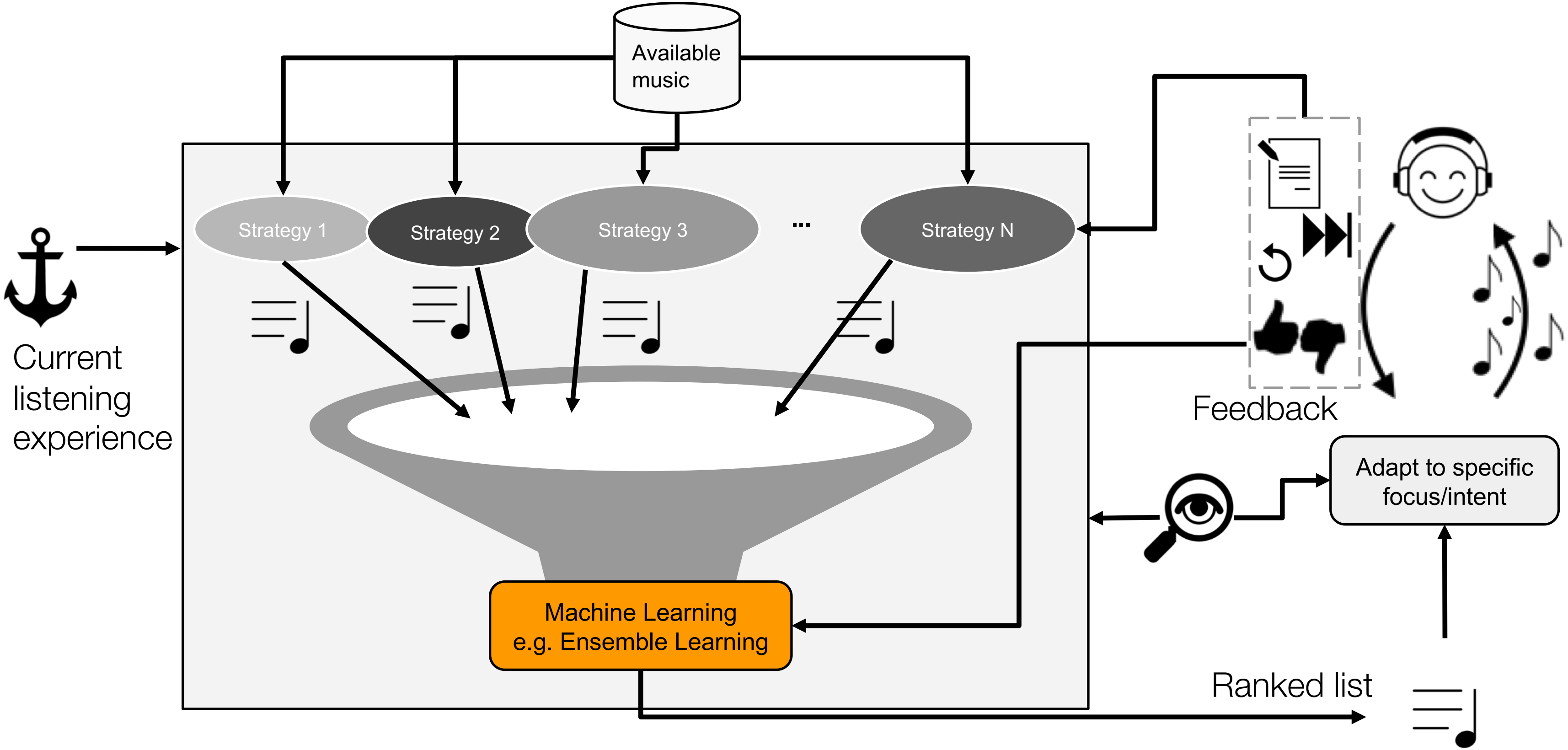
Putting it together



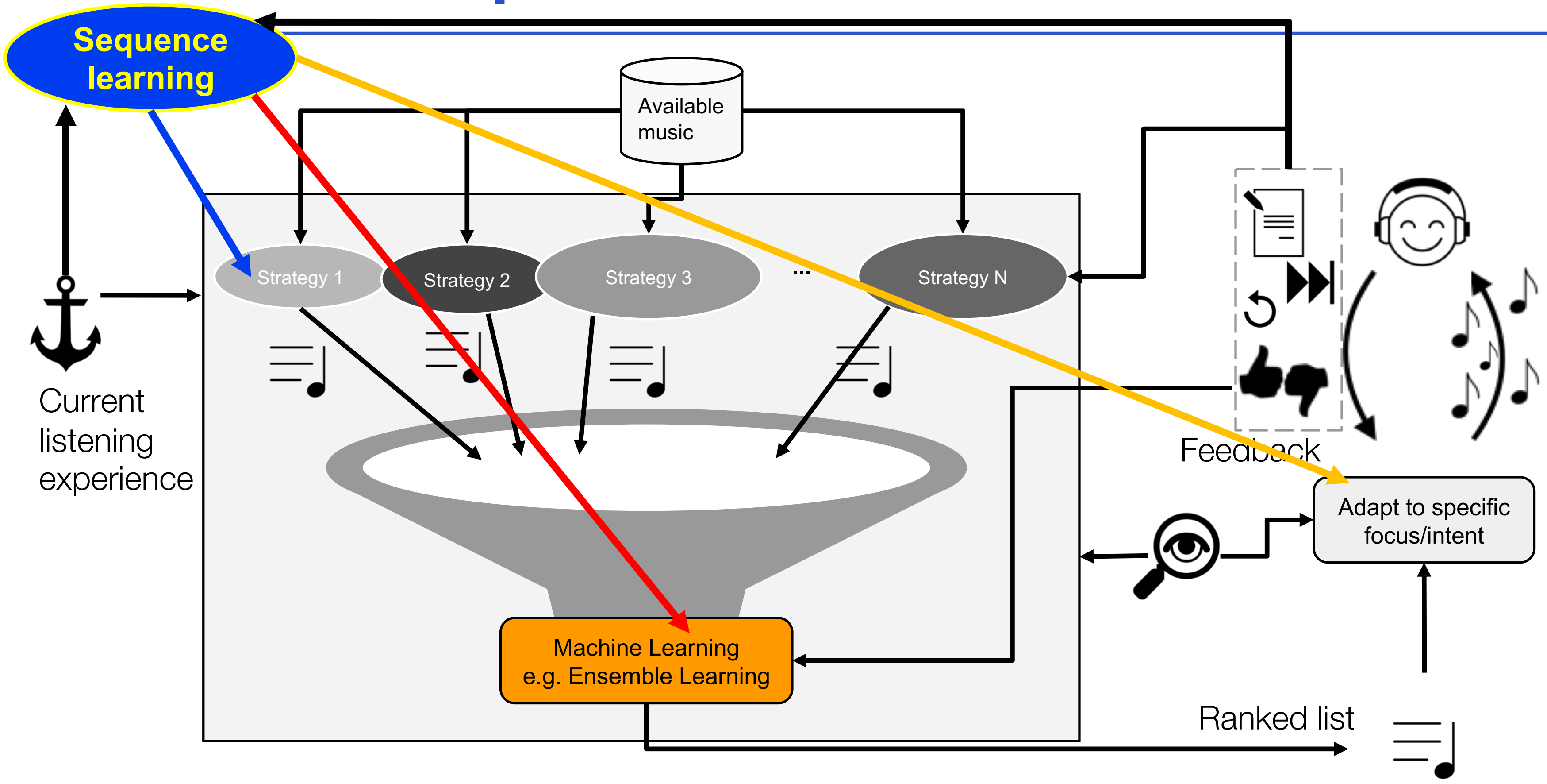
Recommendation Pipeline



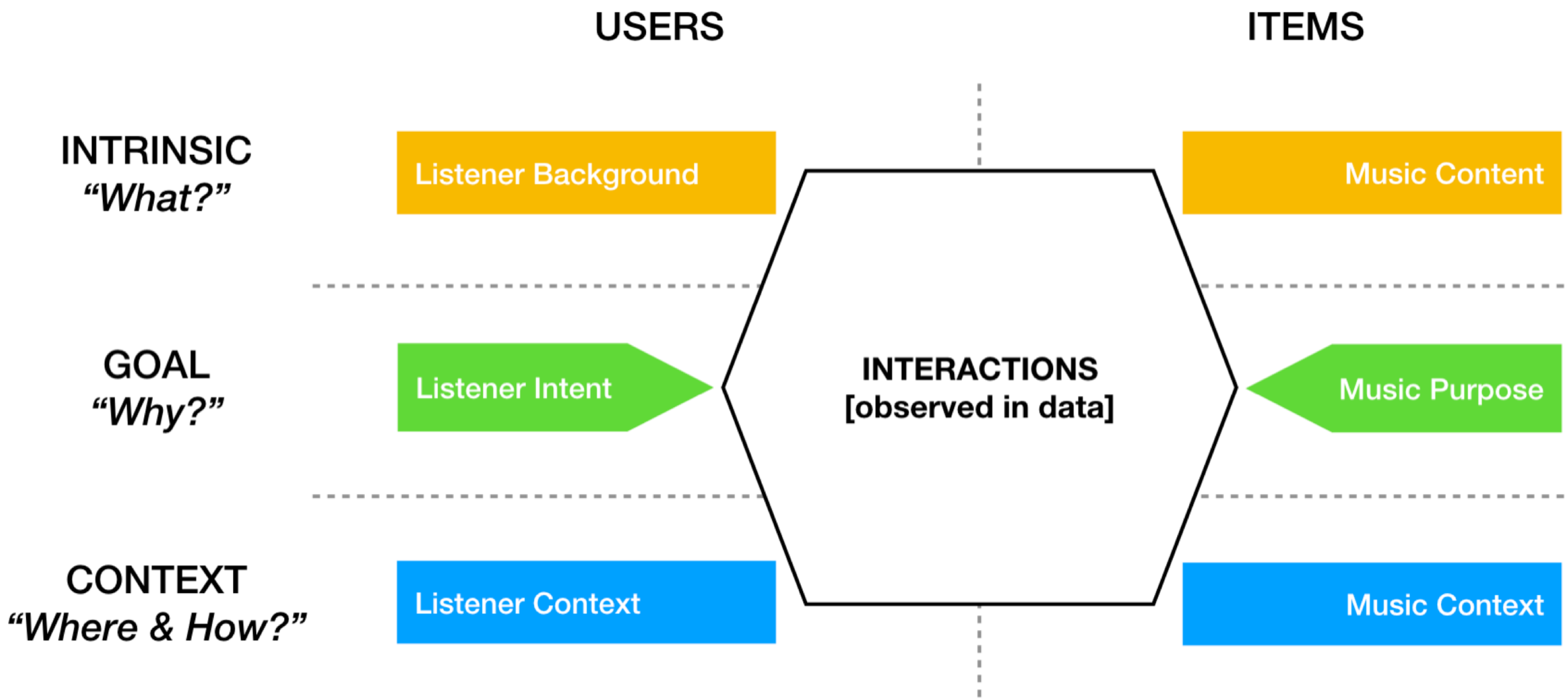
Where does sequence-awareness fit?



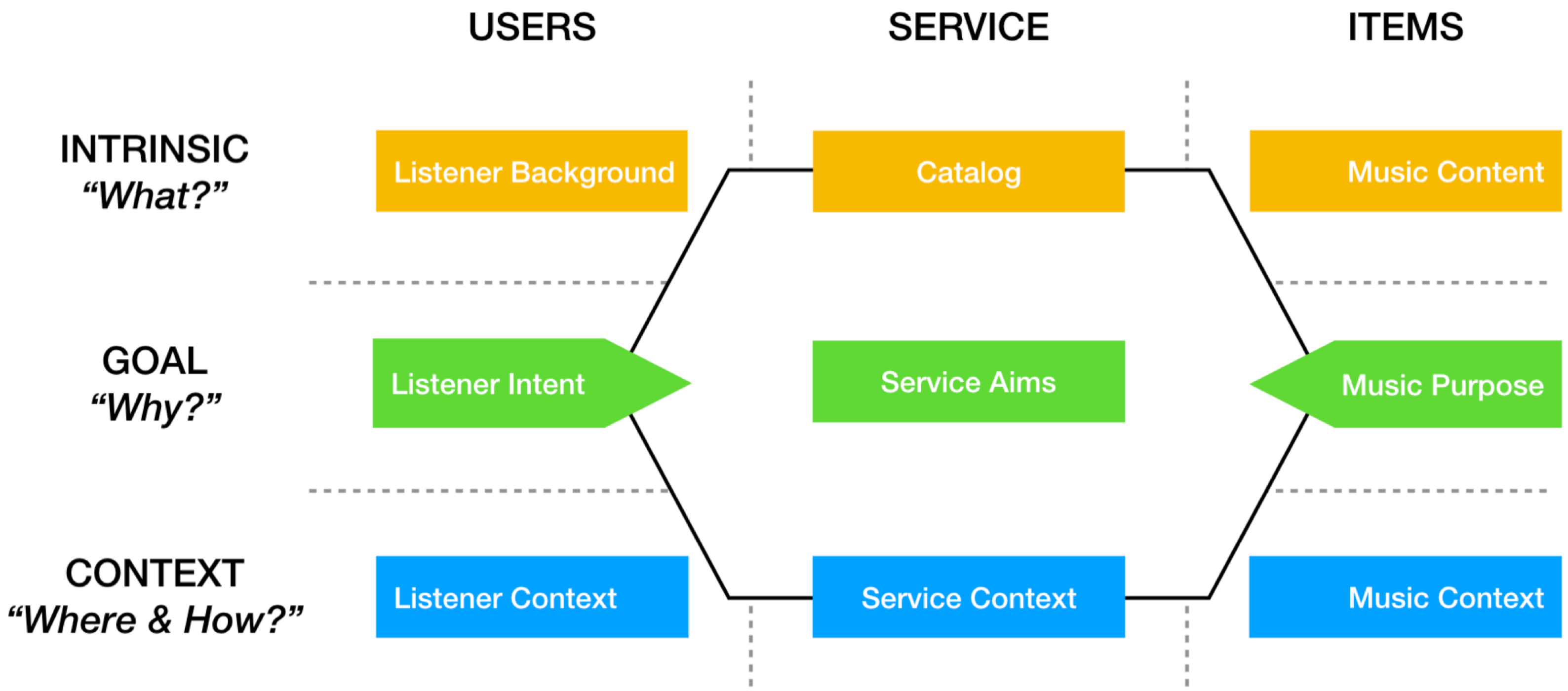
Where does sequence-awareness fit?



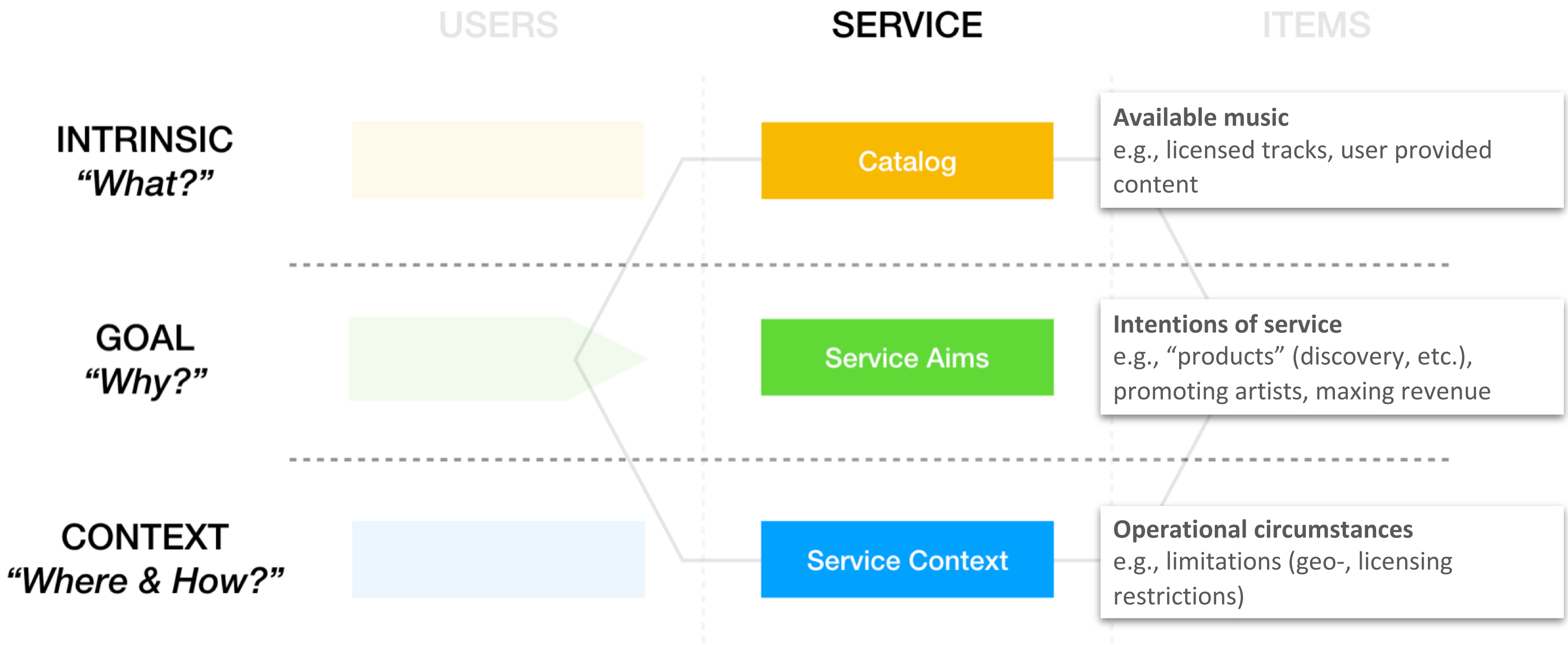
One more thing...



Factoring the Service into the Picture



Factors Hidden in the Data



Looking into Service in More Detail

Recommendations (+collected data!) depend on **factors other than users or items**

Catalog

- Which content is provided/recommended?
- e.g. Soundcloud recommends different content than Spotify

Service Aims

- Why is this service in place? What is the purpose/identified market niche?
- What are the identified use cases? (Discovery? Radio? Exclusives? Quality?)
- Do they push their own content (cf. Netflix)?

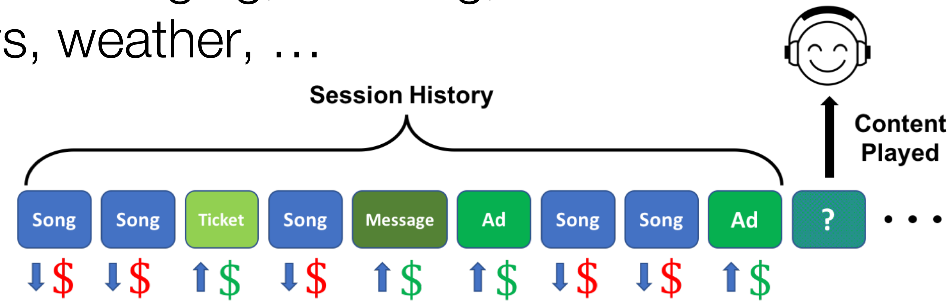
Service Context

- How do catalog and service aims depend on context?
- Are there licensing issues/restrictions in particular countries?
- Is the service context-aware? (e.g. app vs desktop/browser)

Service Aims

Why are recommendations made the way they are?

- **Multistakeholder situation**
  platform, artists, labels, brands, …
- Platform **pays for streaming music,**
  but **gets revenues** from ads, artist messaging, ticketing, etc.
  Other content for UX: sports, news, weather, …
- Exploit vs Explore:
  High short-term satisfaction vs
  user profile development.
  Discovery function!



[Jannach & Adomavicius, 2016] *Recommendations with a purpose*, RecSys.
[Abdollahpouri & Essinger, 2017] *Multiple Stakeholders in Music Recommender Systems*, WS on Value-Aware and Multistakeholder Rec.

Maybe we need to talk more about service biases

- Data from one service not generalizable to others

  SOUNDCLOUD ≠ bandcamp ≠ Spotify ≠ DEEZER ≠ pandora ≠ …

- Particularly for niche market segments

  ≠ IDAGIO ≠ pono ≠ qobuz ≠ …

- And different listening patterns (+content) in different parts of the world

  ≠ KKbox ≠ SUPERPLAYER ≠ simfy africa ≠ …

- Service influences listening behavior; it's different to listening "in the wild"
- Focused service with clear customer base vs addressing all (market new products to underrepresented demographics)

New and related: [Porcaro & Gomez, 2019] *20 years of Playlists: a Statistical Analysis on Popularity and Diversity",* ISMIR.

Data Biases

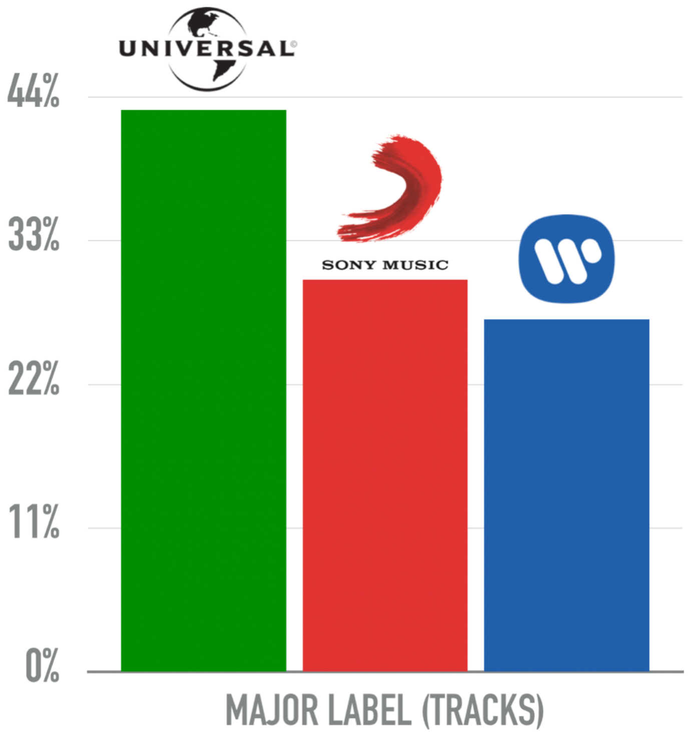
- ”Service biases” directly affect the data collected and therefore research datasets and experimentation
- Other biases in MIR datasets as well
    - Popularity biases (+feedback loops!)
    - Selection biases (no “alternate realities”)
    - Sampling biases (are included events representative?)
    - Cultural and community biases
    - Historical biases (symbolic, Classical music; licensing: royalty free)
- Impacts generalization of findings

Practical: Datasets

- Million Song Dataset: https://labrosa.ee.columbia.edu/millionsong
- Million Musical Tweets Dataset: http://www.cp.jku.at/datasets/mmtd
- #nowplaying Spotify playlists dataset: http://dbis-nowplaying.uibk.ac.at
- LFM-1b: http://www.cp.jku.at/datasets/LFM-1b
- Celma's Last.fm datasets: http://www.dtic.upf.edu/~ocelma/MusicRecommendationDataset/index.html
- Yahoo! Music: http://proceedings.mlr.press/v18/dror12a.html
- Art of the Mix (AotM-2011) playlists: https://bmcfee.github.io/data/aotm2011.html
- Spotify Million Playlist Dataset (RecSys Challenge 2018): ???

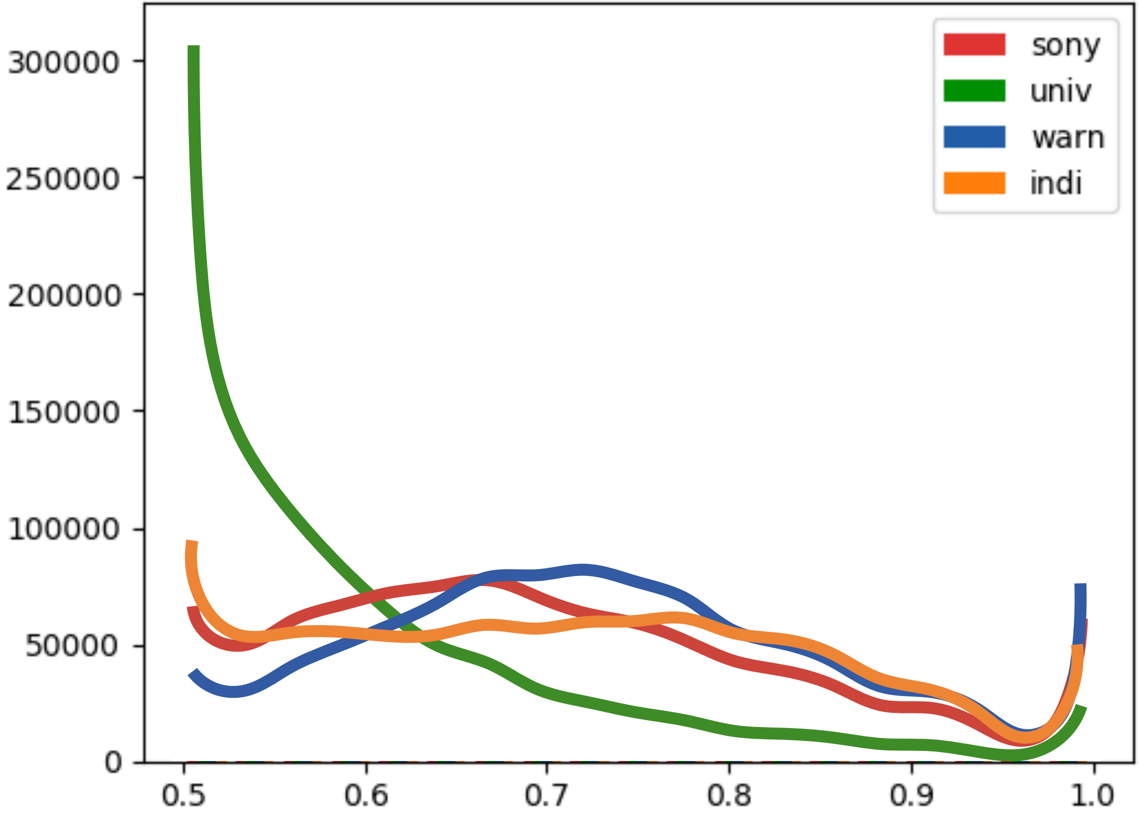
Investigating Datasets

- Analysis of Spotify playlist dataset (MPD)
- 1 million US playlists
- Webcrawler to identify record label of tracks
- Information for about 50% of tracks



Investigating Datasets

- Investigating playlist diversity wrt. major record labels
- NB: work in progress
- First attempt: Entropy-based
- Left: pure
  Right: diverse



Conclusions and Outlook

Many more use cases in music recommendation

- Live Music Business, e.g.
  - Recommending upcoming concerts to listeners
  - Recommending artists to e.g. music festivals
- Recommendations for artist management, e.g.
  - Help agents find best opportunities for artists
- Recommendations to artists
  - Recommending artists where to play
  - Help artists grow their careers, with insights based on data
  - Help artists communication with their fanbase

Many more use cases in music recommendation

- Data Science for record labels, e.g.
  - Assist A&R in finding new talents
  - An artist is launching an album, which track(s) to promote?
  - Make the best use / better monetization of back-catalogue
  - General assistance in business decisions
  - Marketing (where, to whom, how)
  - etc.
- Alternative audio content to music, e.g.
  - Ads (where a lot of $$$ is)
  - News, Podcasts
  - Artist messages

NB: Interesting explore/exploit trade-off

Further opportunities

- Voice-driven interaction with music
    - Dedicated hardware (for home or car) vs. usual interfaces (e.g. phone)
    - Smart speaker growth
    - Today: “command-and-fetch”, e.g. “Play God’s Plan by Drake”
    - Tomorrow: More casual interactions, ambiguous queries, conversations
    - Calls for: Metadata, Personalization
    - Competes with terrestrial radio (more passive listening)

[Dredge; 2018] *Everybody’s talkin’: Smart speakers and their impact on music consumption*, Music Ally Report fo BPI and ERA.

Ethics

- Business-related recommendations (e.g. promotional content) vs. what the user actually wants/needs
- Impact on popular culture (shaping what makes popular culture)
    - Responsibility to counteract algorithmic biases and business-only metrics
    - “Filter bubble”
- Impact on “how” people listen to music (e.g. influence on curiosity)
- Impact on artists, on what’s successful, on the type of music composed
- Privacy

[Holzapfel et al., 2018] *Ethical dimensions of music information retrieval technology*, TISMIR.
[Knijnenburg, Berkovsky, 2017] *Privacy for Recommender Systems*, Tutorial RecSys 2017
[Werthner et al., 2019] Vienna Manifesto on Digital Humanism. https://www.informatik.tuwien.ac.at/dighum/

Research Challenges

- Understanding **listening behavior** and **listener intent** in context

  → Insights from social psychology, cf. Laplante [2015], but not much impact on actual music recommenders

- Improving managing a listener's plurality of tastes
- **Listener Background**: Gain deeper understanding of influence of emotion, culture, and personality on music preferences (also general vs. individual patterns)
- Music Purpose: somewhat less relevant, but still missing in the picture
- Blending social interactions in music streaming
- Blending human-curated recommendations with algorithmic ones

[Laplante, 2015] *Improving Music Recommender Systems: What Can We Learn From Research On Music Tastes?*, ISMIR.

[Motajcsek et al. 2016] *Algorithms Aside: Recommendations as the Lens of Life*, RecSys 2016

A Word on Research Methodology

- **Qualitative methods** are also used in music rec. (investigating e.g., listening and music seeking behavior)
- E.g., in **Music Creation**, recommenders are seen critical

  "I am happy for it to make suggestions, especially if I can ignore them"

- Artistic originality in jeopardy
- Imitation is not the goal: opposition is the challenge

  "I'd like it to do the opposite actually, because the point is to get a possibility, I mean I can already make it sound like me, it's easy." (TOK001)

[Andersen, Knees; 2016] *Conversations with Expert Users in Music Retrieval and Research Challenges for Creative MIR*. ISMIR.

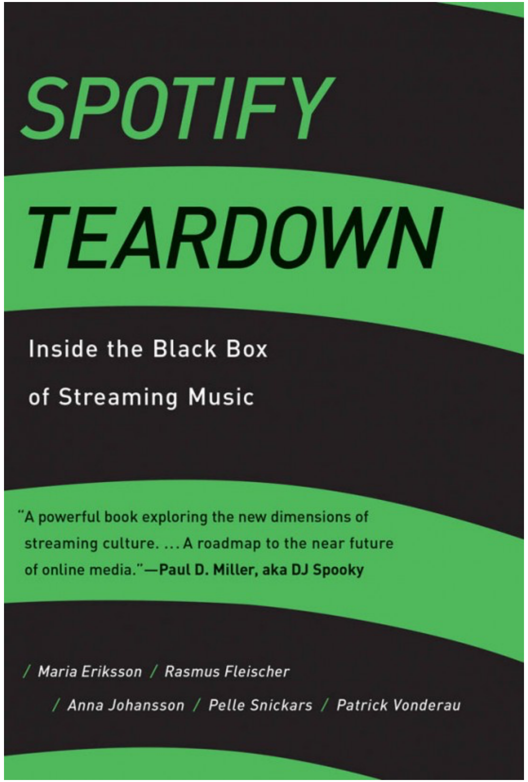
[Ekstrand, Willemsen; 2016] *Behaviorism is Not Enough: Better Recommendations through Listening to Users*. RecSys.

[Knees, Schedl, Ferwerda, and Laplante, 2019] *User Awareness in Music Recommender Systems*. Personalized Human-Computer Interaction, Augstein et al. (Eds.), (expected Sept. 2019)

Take-Away Messages

- Dramatic changes in music consumption (growth, ownership → access) imply great challenges and impact for recommender systems
- Music is not “just another item”, many different representations and sources of data for manifold recommendation techniques
- Recommender have potential to be disruptive in many parts of the music industry (not just end-user consumption)
- Creating truly personalized music RecSys and evaluating user satisfaction is still challenging
- Beware of biases in “real-world data”

Recommended Reading



Spotify Teardown:
Inside the Black Box of Streaming Music,

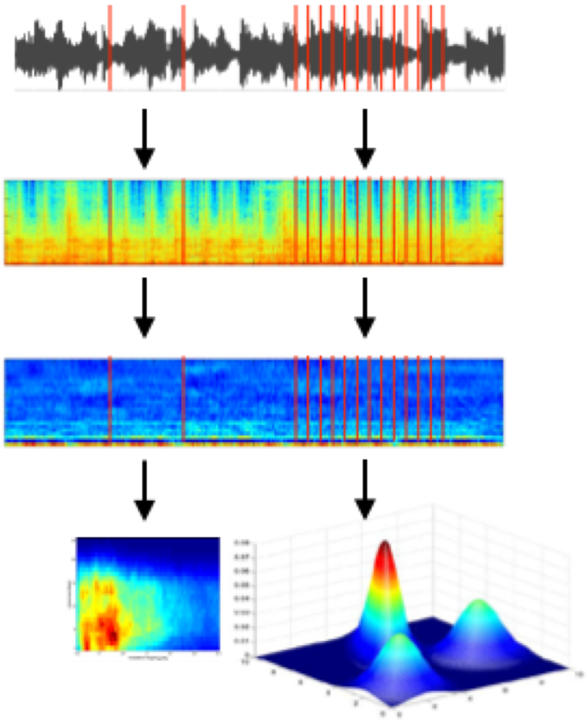
Maria Eriksson, Rasmus Fleischer,
Anna Johansson, Pelle Snickars, and
Patrick Vonderau.

MIT Press, 2019.

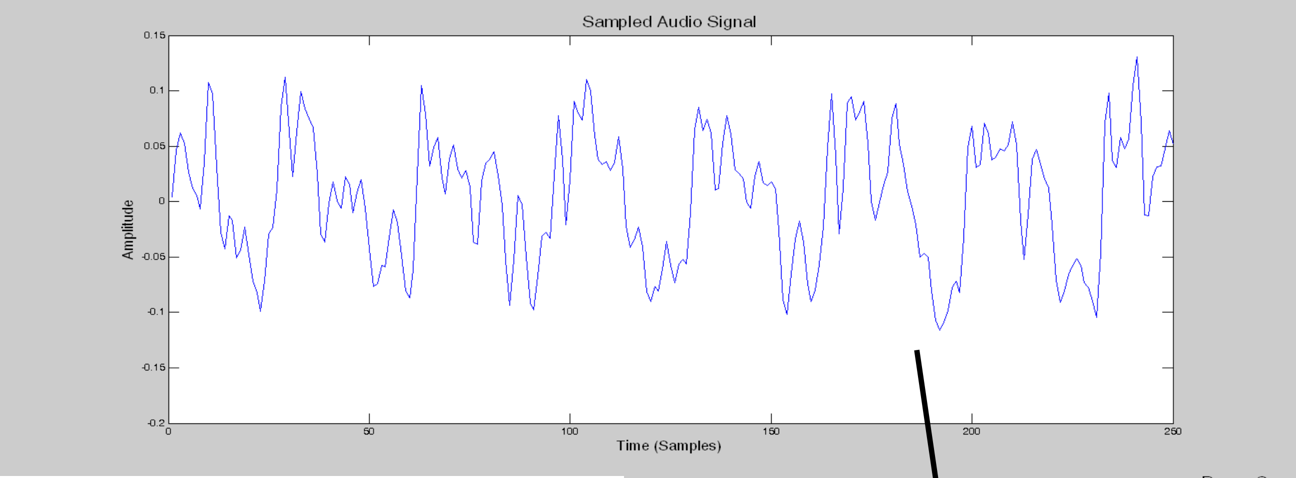
Extended Version
Music Content Analysis

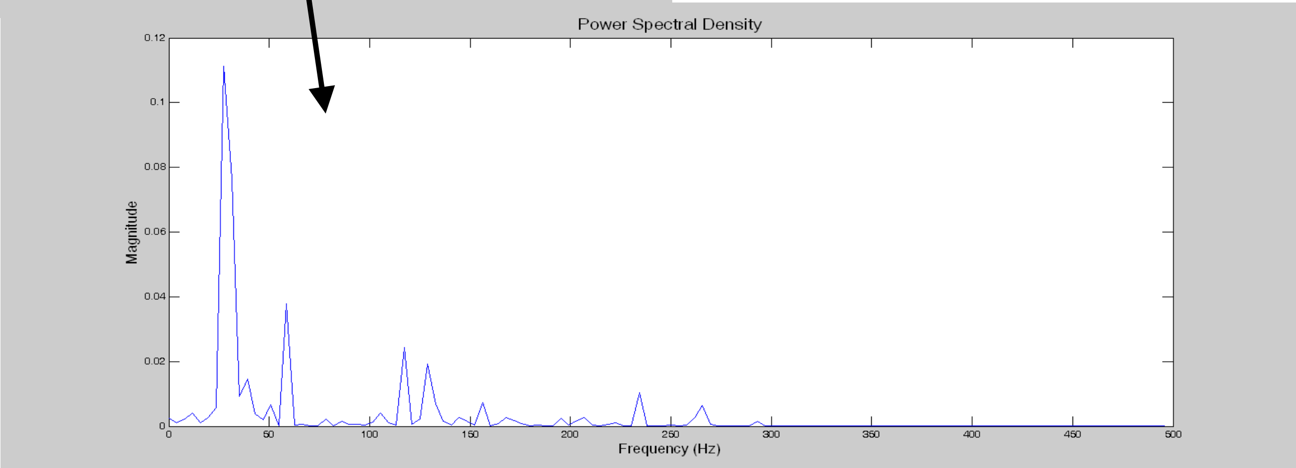
Audio Features: Basic Processing Steps

- Convert signal from time domain to *frequency domain*, e.g., using a Fast Fourier Transform (FFT)
- *Psychoacoustic transformation* (Mel-scale, Bark-scale, Cent-scale, ...): mimics human listening process (not linear, but logarithmic!), removes aspects not perceived by humans, emphasizes low frequencies
- Extract features
  - *Block-level* (large time windows, e.g., 6 sec)
  - *Frame-level* (short time windows, e.g., 25 ms) needs model distribution of frames
- Calculate similarities between feature vectors/models

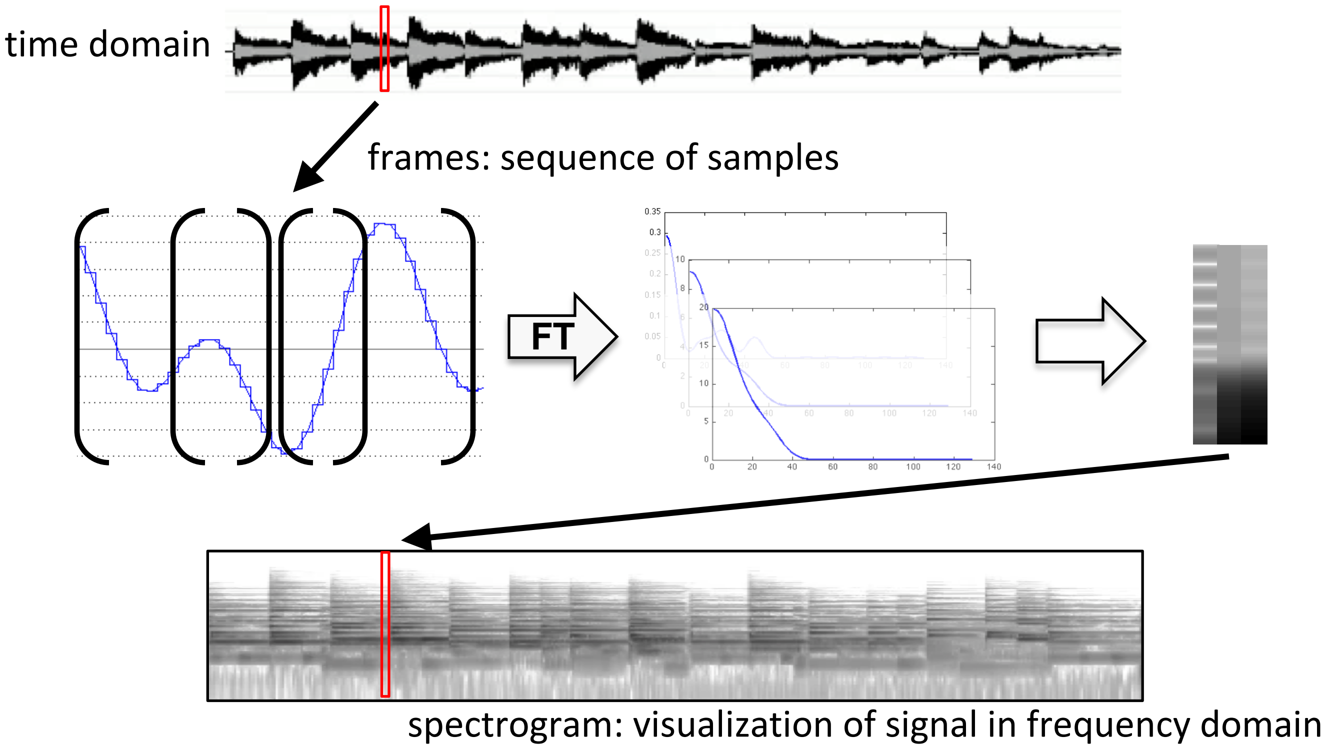


From Time to Frequency Domain (1 Frame)





Fourier Transform (FT) / Spectrogram

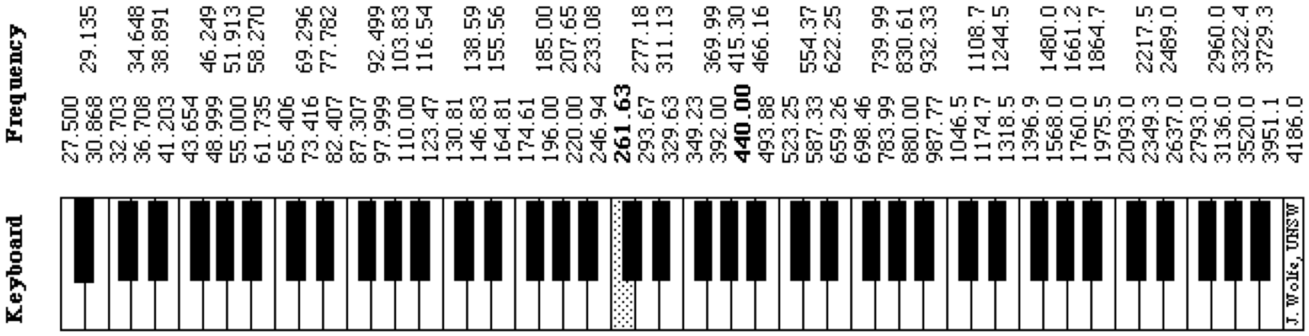


Pitch Class Profiles (aka *chroma vectors*)

- Transforming the frequency activations into well known musical system/representation/notation

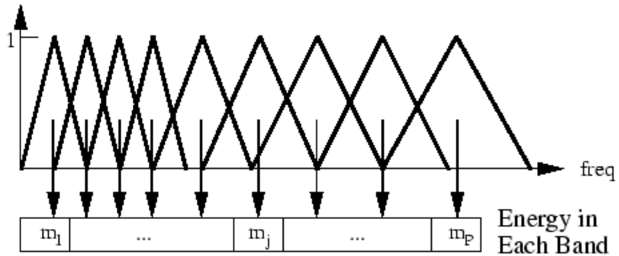
(Fujishima; 1999)

- Mapping to the equal-tempered scale (each semitone equal to one twelfth of an octave)
- For each frame, get intensity of each of the 12 semitone (pitch) classes

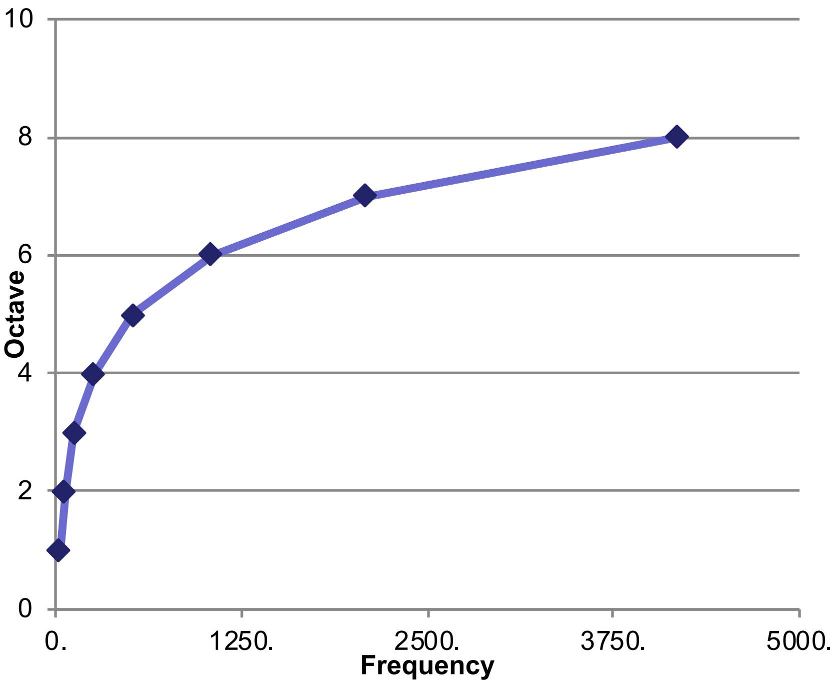


Semitone Scale

- Map data to semitone scale to represent (western) music
- Frequency doubles for each octave
  - e.g. pitch of A3 is 220 Hz, A4 440 Hz
- Mapping, e.g., using triangular filter bank
  - centered on pitches
  - width given by neighboring pitches
  - normalized by area under filter

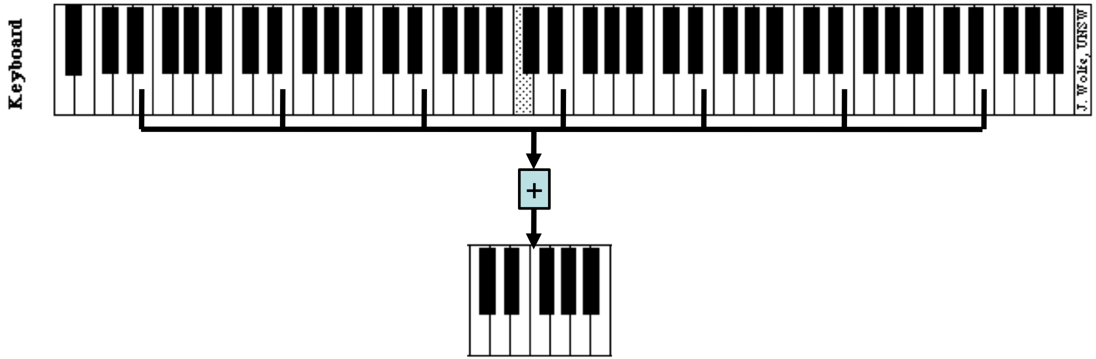


The note C in different octaves vs. frequency



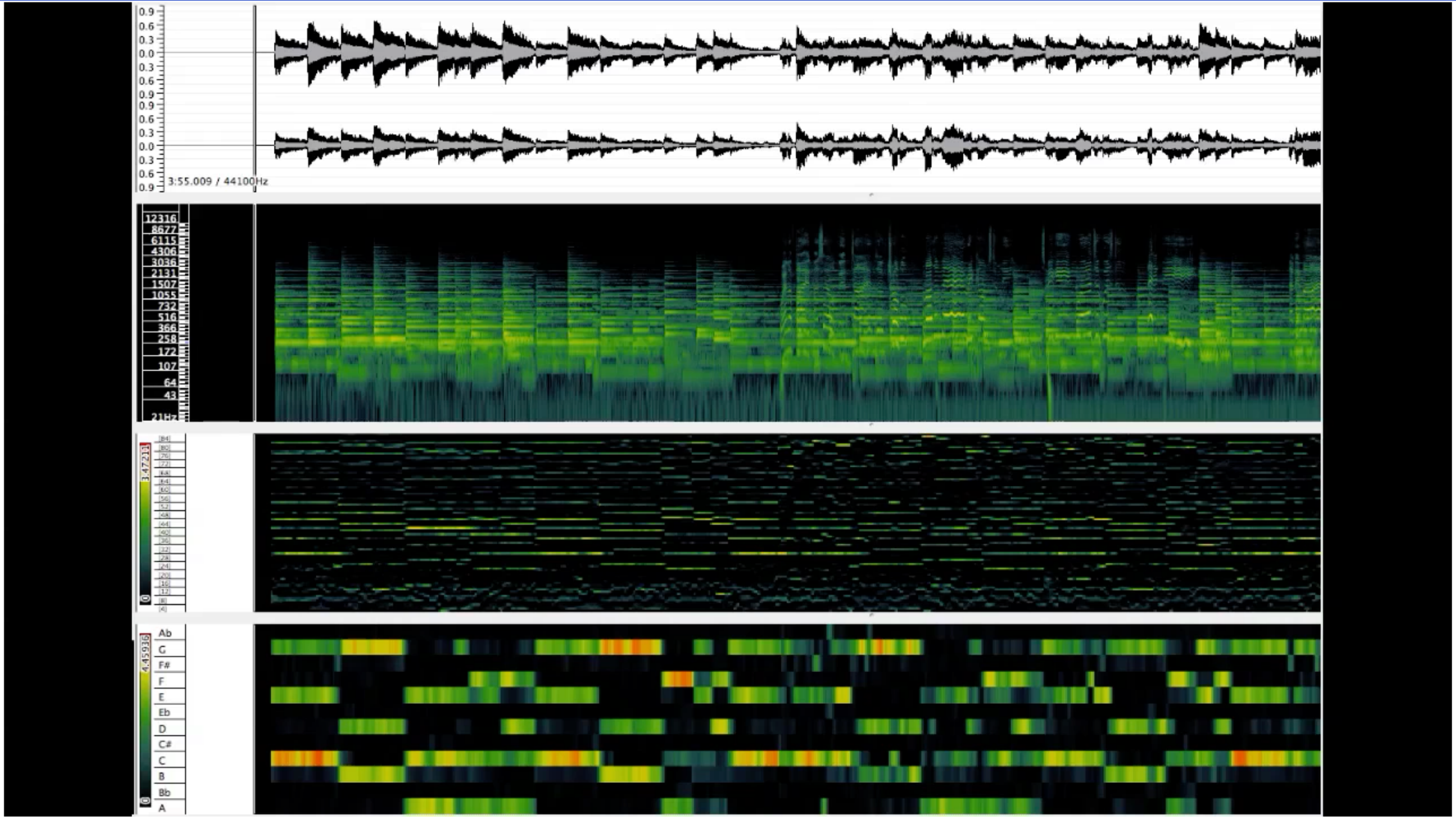
Pitch Class Features

- Sum up activations that belong to the **same class of pitch** (e.g., all A, all C, all F#)



- Results in a 12-dimensional feature vector for each frame
- PCP feature vectors describe tonality
  - Robust to noise (including percussive sounds)
  - Independent of timbre (~ played instruments)
  - Independent of loudness

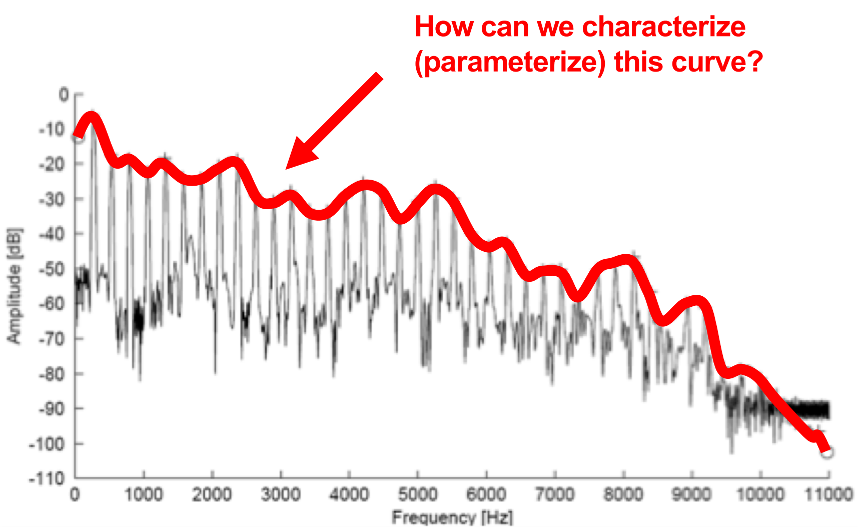
Pitch Class Profiles in Action



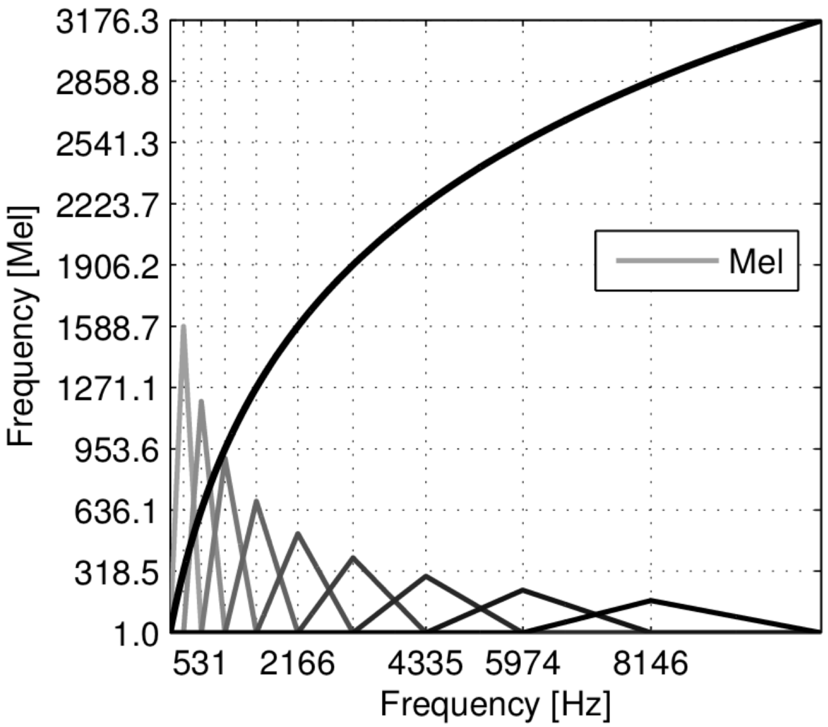
Sonic Visualizer by QMUL, C4DM; http://www.sonicvisualiser.org

MFCCs

- Mel Frequency Cepstral Coefficients (MFCCs) have their roots in speech recognition and are a way to represent the envelope of the power spectrum of an audio frame
  - the spectral envelope captures perceptually important information about the corresponding sound excerpt (*timbral aspects*)
  - sounds with similar spectral envelopes are generally perceived as "sounding similar"



The Mel Scale



- Perceptual scale of pitches judged by listeners to be equal in distance from one another
- Given Frequency f in Hertz, the corresponding pitch in Mel can be computed by

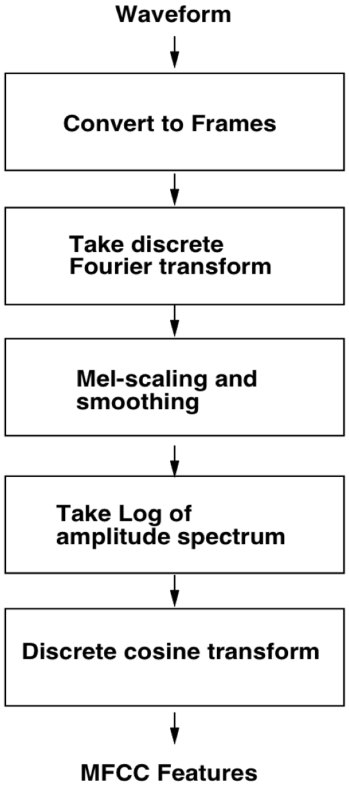
$$m = 2595 \log_{10} \left(1 + \frac{f}{700}\right)$$

- Normally around 40 bins equally spaced on the Mel scale are used

MFCCs

MFCCs are computed per frame

1. Framing
2. DFT: discrete Fourier transform on windowed signal
3. Mapping of spectrum to the Mel scale (melspectrogram, “melgram”), quantization (into e.g., 40 bins)
4. Logarithm of Mel-scaled amplitude (motivated by the way humans perceive loudness)

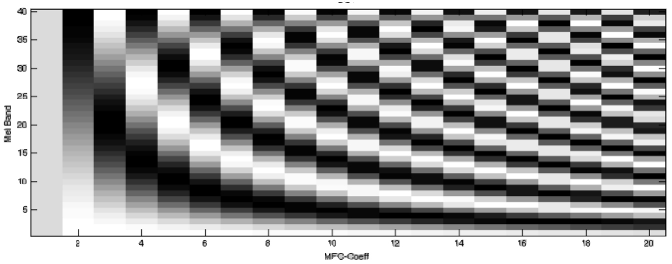


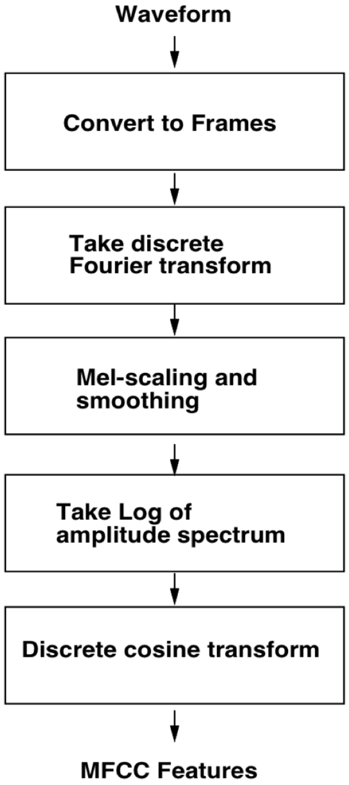
MFCCs

5. perform Discrete Cosine Transform (DCT) to de-correlate the Mel-spectral vectors
   - similar to FFT; only real-valued components
   - describes a sequence of finitely many data points as sum of cosine functions oscillating at different frequencies

$$X_k = \sum_{n=0}^{N-1} x_n \cdot \cos\left(\frac{\pi}{N} \cdot \left(n + \frac{1}{2}\right) \cdot k\right) \qquad k = 0, \ldots, N-1$$

   - results in n coefficients (e.g., n = 20)

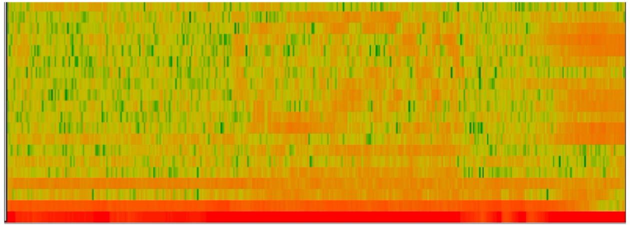




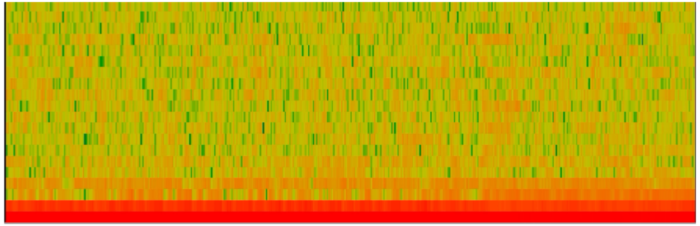
NB: performing (inverse) FT or similar on log representation of spectrum:“cepstrum” (anagram!)

MFCC Examples

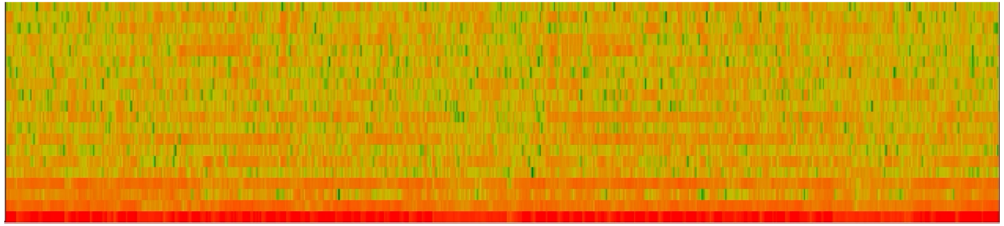
- Beethoven



- Shostakovich

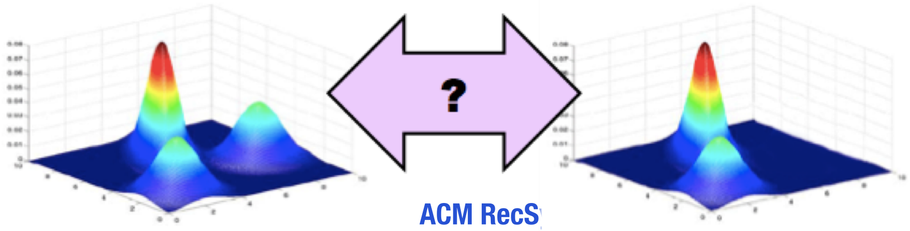


- Black Sabbath



“Bag-of-frames” Modeling

- Full music piece is now a set of MFCC vectors
  - Variable amount of *n*-dim features vectors per piece (*n*… number of MFCCs)
  - Number of frames depends on length of piece
- Need summary/aggregation/modeling of this set
  - Average over all frames? Sum?
- Comparing two songs = comparing their feature distributions
- Implication: loss of temporal information



“Bag-of-frames” Modeling

- Practical solution: describe distribution of all these local features via statistics such as mean, var, cov
- “Quick-and-dirty” approach: compare these values directly
- Better: calculate distance of distributions, e.g. via Earth Mover’s Distance or Kullback-Leibler divergence
- For two distributions, $p(x)$ and $q(x)$, the KL divergence is defined as:

$$KL(p \,||\, q) \equiv \int p(x) \log \frac{p(x)}{q(x)} \, dx$$

- Expectation of the log difference between the probability of data in one distribution (p) and the probability of data in another distribution (q)

MFCCs for Genre Classification

- For multivariate Gaussian distributions, a closed form of the KL-divergence exists

$$KL_{(P||Q)} = \frac{1}{2}\left[\log\frac{|\Sigma_P|}{|\Sigma_Q|} + Tr\left(\Sigma_P^{-1}\Sigma_Q\right) + \left(\mu_P - \mu_Q\right)^{\intercal}\Sigma_P^{-1}\left(\mu_Q - \mu_P\right) - d\right]$$

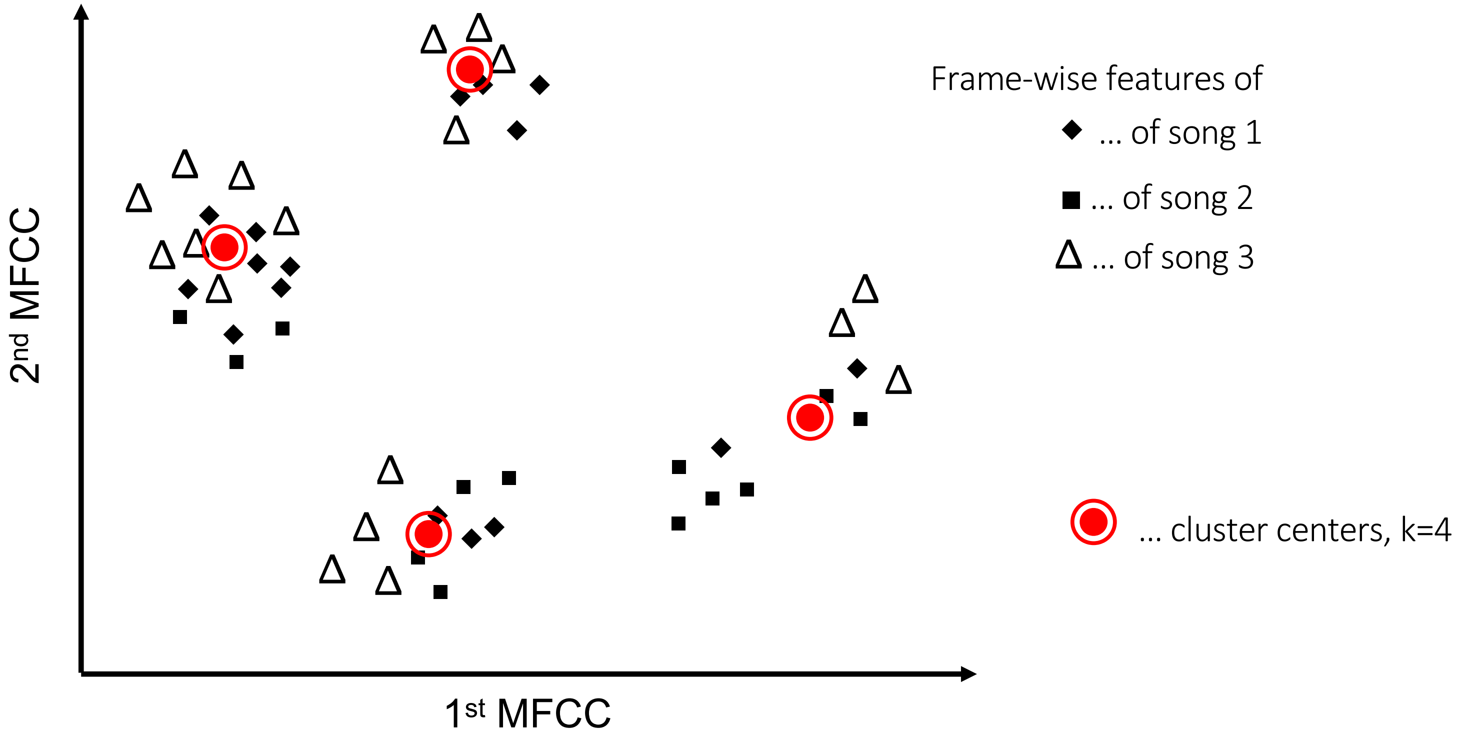
  - μ ... mean, Σ ... cov. mat., Tr ... trace, d ... dimensionality
  - asymmetric, symmetrize by averaging: $d_{KL}(P,Q) = \frac{1}{2}\left(KL_{(P||Q)} + KL_{(Q||P)}\right)$
  - not a metric!

- Use KL divergence on Gaussian model of MFCC “bag-of-frames” as kernel (gram matrix) for Support Vector Machines (SVMs) [Mandel and Ellis, 2005]

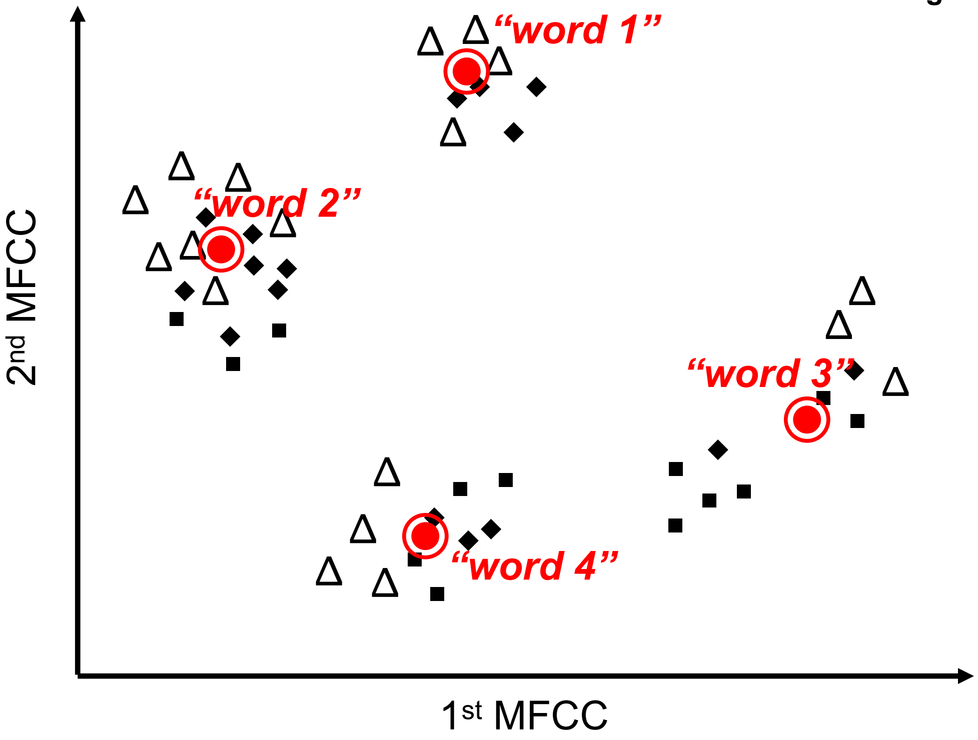
Alternative: Codebook Approach

1. Extract features (e.g., MFCCs from all frames) from all songs in training collection
2. Try to describe the resulting feature distribution/space by finding clusters
   → **clustering** step (e.g., k-means clustering)
3. Cluster centers are the **codebook entries** or "**words**" (cf. "bag-of-words")
   → choice of *k* defines the dimensionality of the new(!) feature vector space
4. For each song (new or in training set), find closest cluster center for each extracted frame feature vector and **create histogram** of how often each cluster center (word) is mapped
5. Normalize histogram
6. Histogram is ***k*-dim global feature vector** of song
7. Compare songs by comparing histogram feature vectors

Codebook Approach (2D Example)



Codebook Approach (2D Example)



counting "word" occurrences:

◆ ... [4, 7, 2, 3]

■ ... [0, 3, 6, 4]

Δ ... [4, 7, 3, 4]

normalize:

◆ ... [0.25, 0.44, 0.13, 0.19]

■ ... [0.00, 0.23, 0.46, 0.31]

Δ ... [0.22, 0.39, 0.17, 0.22]

= song feature vectors

vector space:

- simple similarity (Eucl., cos)
- efficient indexing
- ...

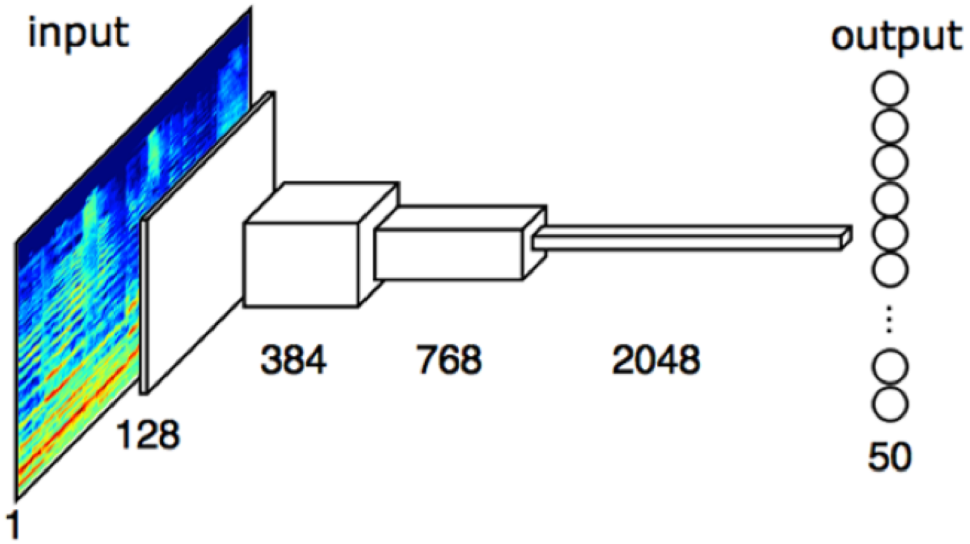
Limitations of “Bag-of-Frames”

- Loss of Temporal Information:
  - temporal ordering of the MFCC vectors is completely lost because of the distribution model (bag-of-frames)
  - possible approach: calculate delta-MFCCs to preserve difference between subsequent frames
- Hub Problem (“Always Similar Problem”)
  - depending on the used features and similarity measure, some songs will yield high similarities with many other songs without actually sounding similar (requires post-processing to prevent, e.g., recommendation for too many songs)
  - general problem in high-dimensional feature spaces!

A More General Approach

- Automatically learn the features from signal → deep learning architecture
- “End-to-End Learning”
- Input: spectrogram or Mel-spectrogram
- CNN architecture (or CRNN)
- Output: Single (e.g., genre) or multi-class labels (e.g., tags)
- Still: carefully design architecture of network
    - What is the task? (e.g., percussive vs harmonic or both)
    - Which properties are desired? (e.g. pitch invariances)

End-to-End Learning for Tags



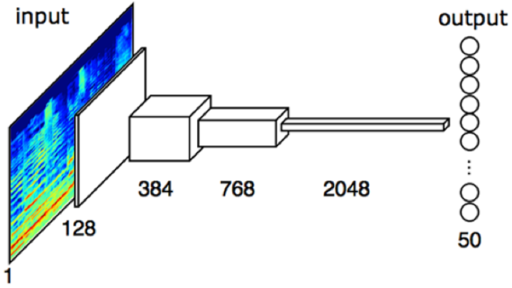
[Choi et al., 2016]

- Automatic learning of audio features for tagging with CNN
- CNN properties:
  - translation, distortion, and locality invariance
  - → musical features/events relevant to tags can appear at any time or frequency range

Architecture

- Input: 29.1 sec audio clips (MagnaTagATune clip length)
- 12 kHz downsampling, 256 samples hop size → 1,366 frames per clip
- Log amplitude Mel-spectrogram with 96 Mel bands
- ReLUs in conv. layers
- Batch normalization, dropout, ADAM optimization
- Output: 50 tags

Mel-spectrogram *(input: 96×1366×1)*
Conv 3×3×128
MP (2, 4) *(output: 48×341×128)*
Conv 3×3×384
MP (4, 5) *(output: 24×85×384)*
Conv 3×3×768
MP (3, 8) *(output: 12×21×768)*
Conv 3×3×2048
MP (4, 8) *(output: 1×1×2048)*
Output 50×1 (sigmoid)

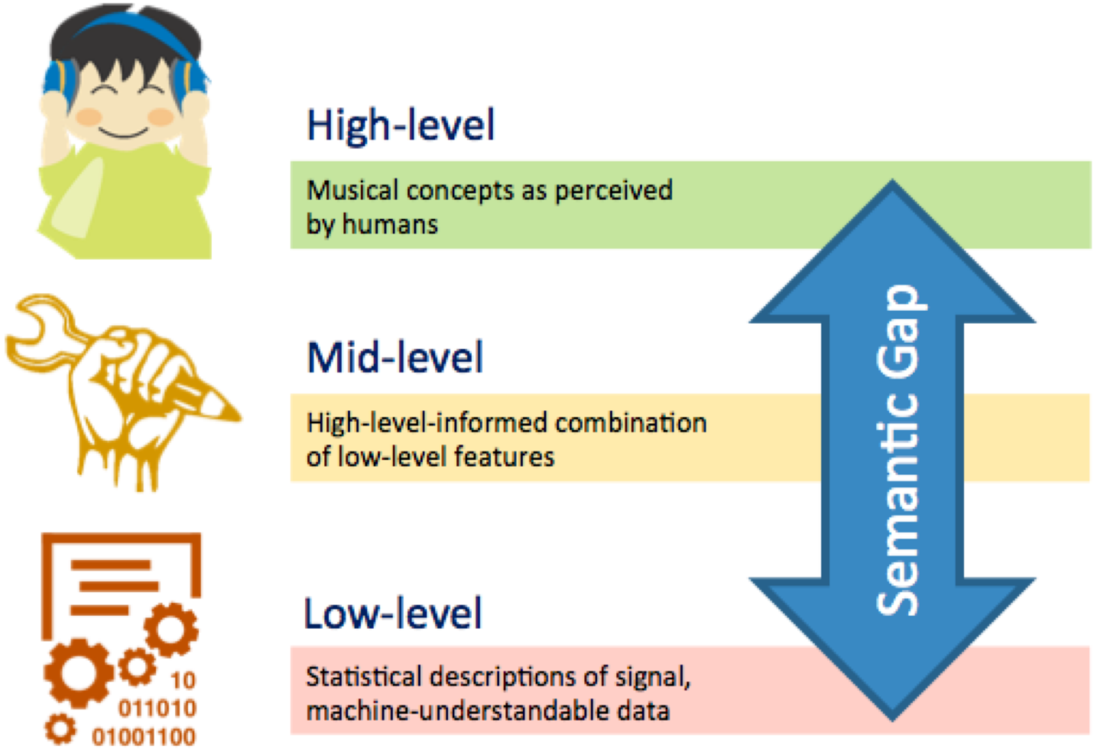


So, great … why is this difficult then?

- “Objective” similarity measure
- Describes the output of the applied transformation
- Works well for genre and mood classification

- The resulting numbers represent a very narrow aspect of acoustic properties, describe no *musical* qualities (structure, development, time dependency, etc.)
- Which sound properties are important to whom and in which context?
- Lack of any personal preferences or experiences
- No consideration of multimodality of music perception

Mind the Semantic Gap!

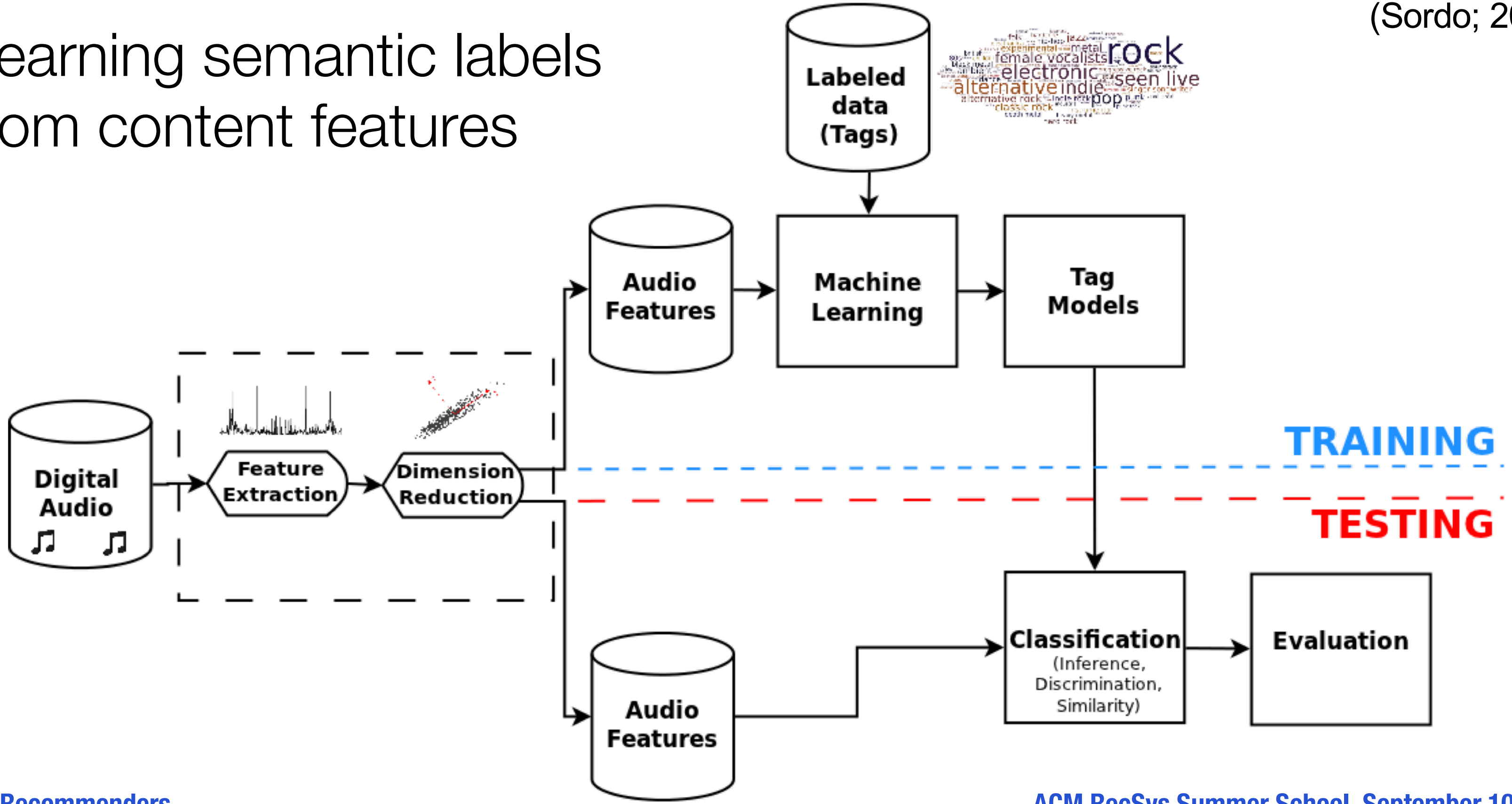


- e.g. melody, themes, motifs + “semantic” categories: genre, time period, mood, etc
- e.g. MFCCs, chroma + (latent) text topics
  typically the level used when estimating similarity!
- e.g. energy, zero-crossing-rate + text: TFIDF

Auto-Tagging

(Sordo; 2012)

Learning semantic labels from content features



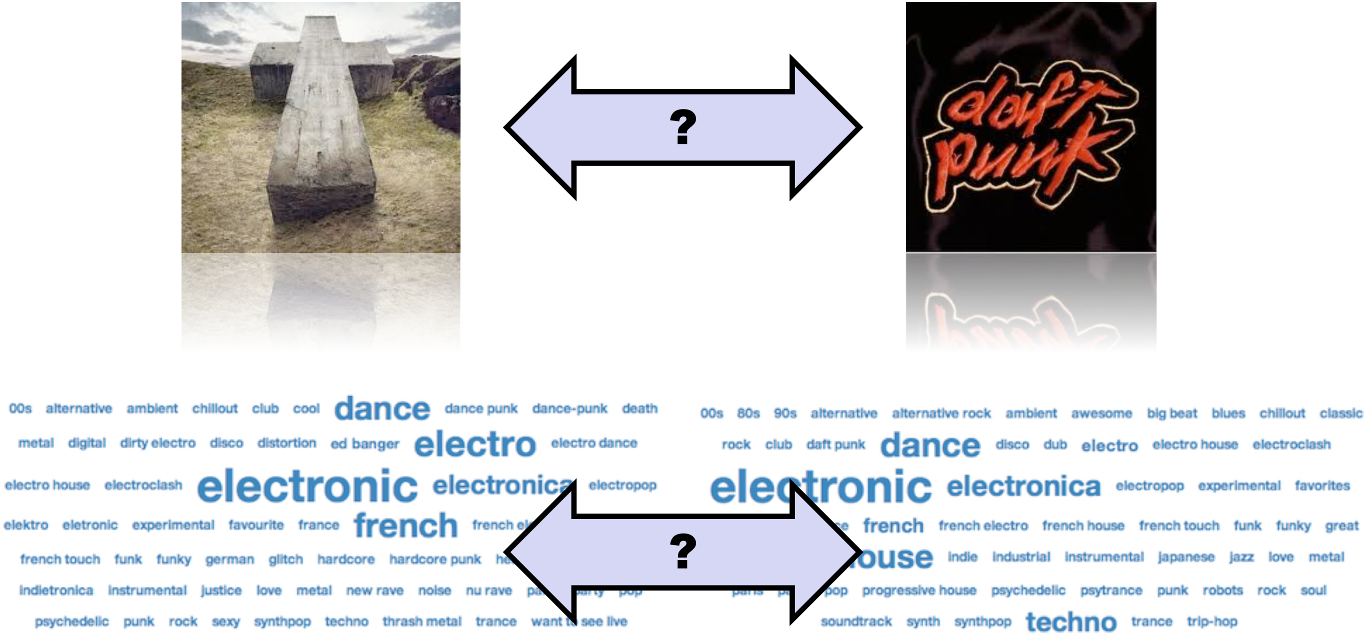
Text Analysis Methods (Basic IR)

- Text-processing of **user-generated content** and **lyrics**
→ captures aspects beyond pure audio signal (→ **Music Context**)
→ no audio file necessary
- Transform the content similarity task into a text similarity task
(cf. "content-based" movie recommendation)
- Allows to use the full armory of text IR methods, e.g.,
  - Bag-of-words, Vector Space Model, TFIDF
  - Topic models (LSI, LDA, …), word2vec
- Example applications: Tag-based similarity, sentiment analysis (e.g., on reviews), mood detection in lyrics

[Knees and Schedl, 2013] *A Survey of Music Similarity and Recommendation from Music Context Data,* Transactions on Multimedia Computing, Communications, and Applications 10(1).

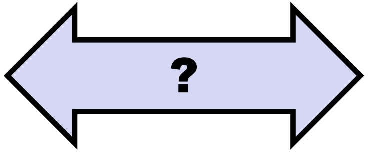
Using Texts for Music Recommendation

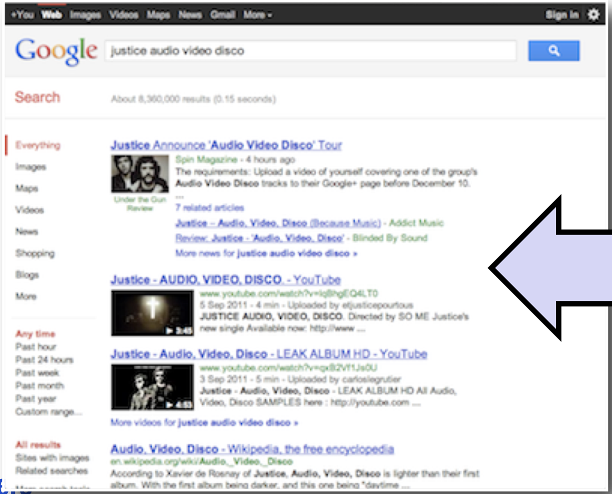
Recommending non-texts based on associated data, e.g., tags

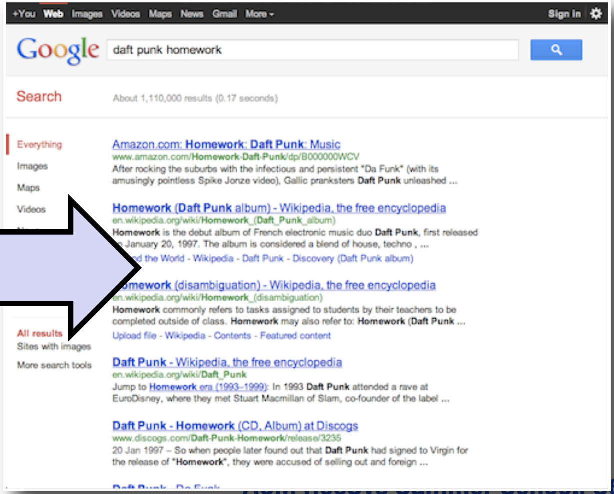


Using Texts for Music Recommendation

Recommending non-texts based on associated data, e.g., web pages

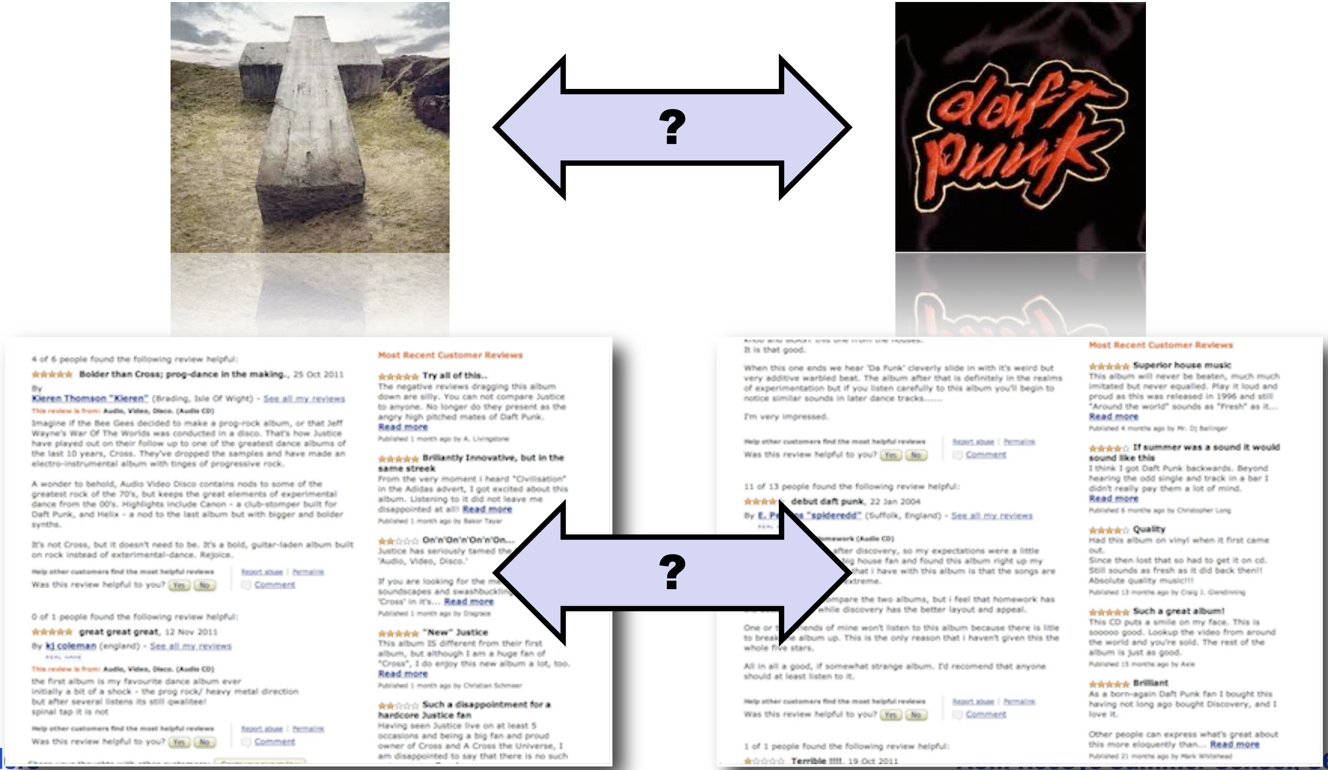






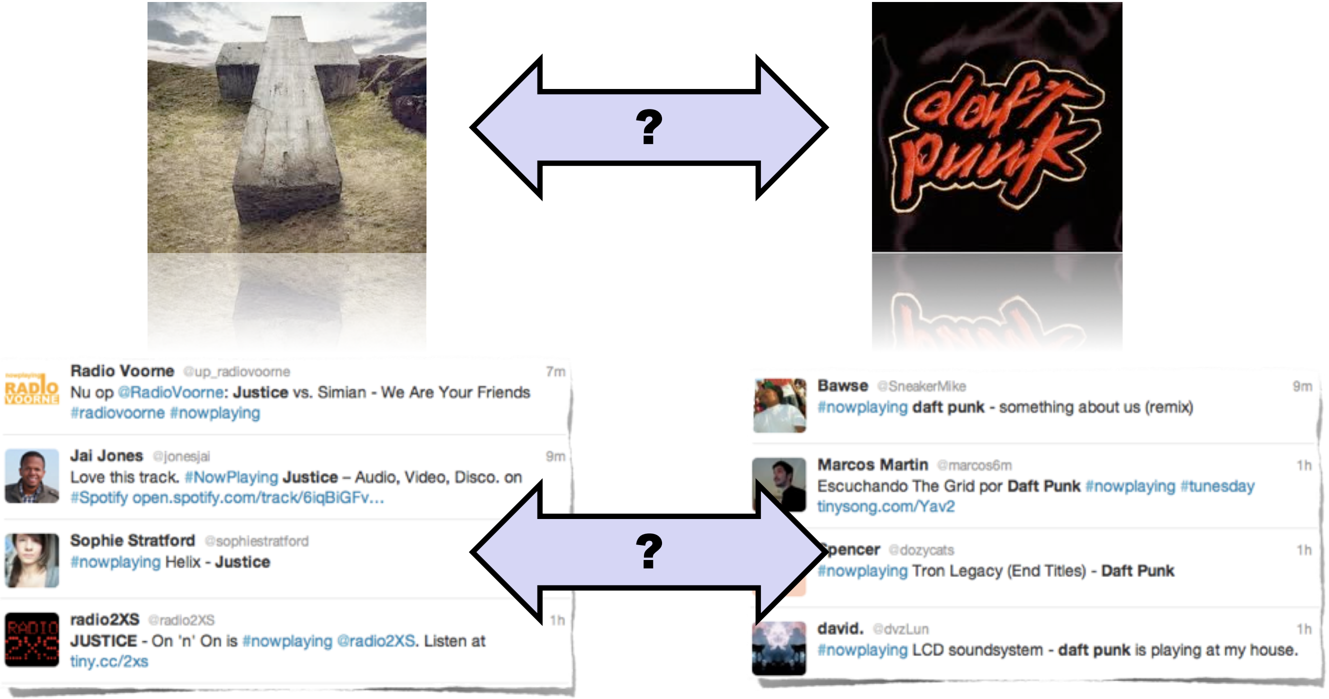
Using Texts for Music Recommendation

Recommending non-texts based on associated data, e.g., reviews



Using Texts for Music Recommendation

Recommending non-texts based on associated data, e.g., tweets



Using Texts for Music Recommendation

Recommending music based on related texts, e.g., lyrics

?

Before day break there was none
And as it broke there was one
The Moon, the sun, it goes on 'n' on
The winter battle was won
The summer children were born
And so the story goes on 'n' on
Come woman if your life beats
Those we buried with the house keys
Smoke and feather where the fields are green
From here to eternity
Come woman on your own time

?

Around the world, around the world
Around the world, around the world
Around the world, around the world

Around the world, around the world
Around the world, around the world
Around the world, around the world

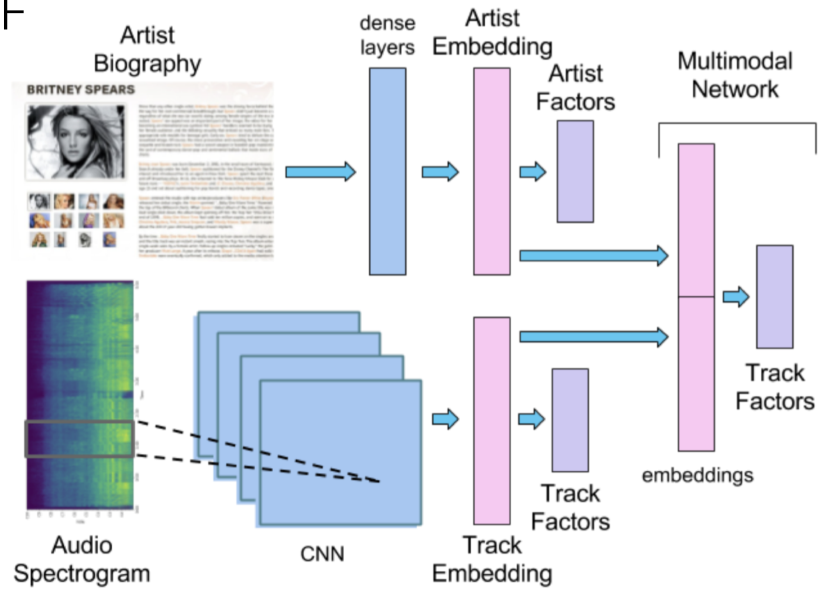
Around the world, around the world
Around the world, around the world
Around the world, around the world

Multimodal Approaches

- Incorporation of different sources / complementary information
- Content to handle cold-start problem in CF
- E.g. combining artist biography text embeddings with CNN-trained track audio embeddings

[Oramas et al., 2017] *A Deep Multimodal Approach for Cold-start Music Recommendation.* RecSys DLRS workshop.

- E.g. fusing deep features from audio and image (album covers) and text



[Oramas et al., 2018] *Multimodal Deep Learning for Music Genre Classification.* TISMIR 1(1).

Toolboxes for Text Analysis

- Natural Language Toolkit nltk (Python): https://www.nltk.org
- Gensim (Python): https://radimrehurek.com/gensim/
- GATE (Java): https://gate.ac.uk
- MeTA (C++): https://meta-toolkit.org
- Apache OpenNLP (Java): http://opennlp.apache.org
- jMIR (Java): http://jmir.sourceforge.net

Challenges when relying on Music Context Data

- Dependence on availability of sources (web pages, tags, playlists, ...)
- Popularity of artists may distort results
- Cold start problem (newly added entities do not have any information associated, e.g. user tags, users' playing behavior)
- Hacking and vandalism (cf. last.fm tag "*brutal death metal*")

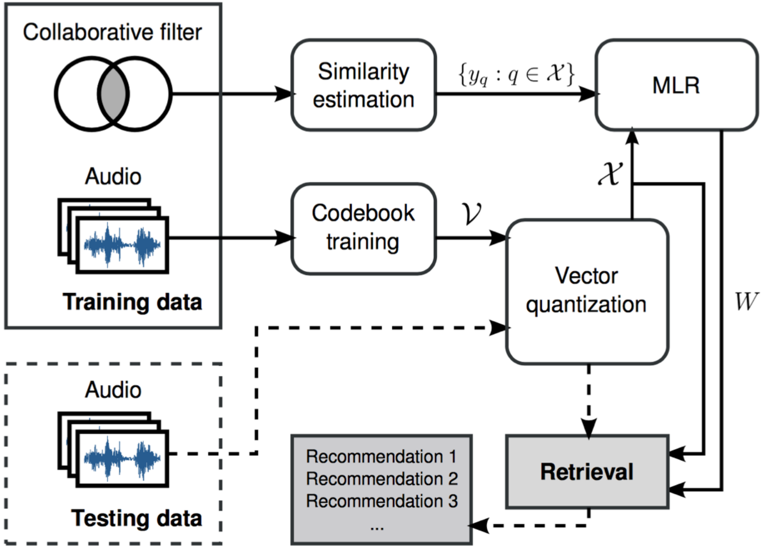
- Bias towards specific user groups (e.g., young, Internet-prone, metal listeners on last.fm)
- (Reliable) data often **only available on artist level for music context**
- Content-based methods do not have these problems (but others)

Feedback-Transformed Content

- CF model as target for learning features from audio
- Dealing with cold-start: predict CF data from audio
- Potentially: personalizing the mixture of content features

- E.g., learning item-based CF similarity function from audio features using metric learning

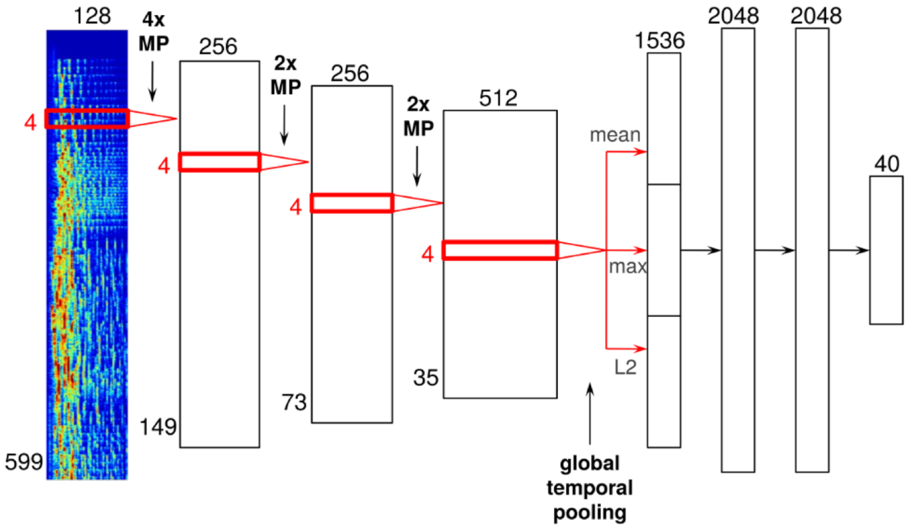
[McFee et al., 2012] *Learning Content Similarity for Music Recommendation.* IEEE TASLP 20(8).



Feedback-Transformed Content

- E.g. learning latent item features using weighted matrix factorization
  - CNN input: mel-spectrogram
  - CNN targets: latent item vectors
  - Visualization of clustering of learned song representations (t-SNE) on next slide

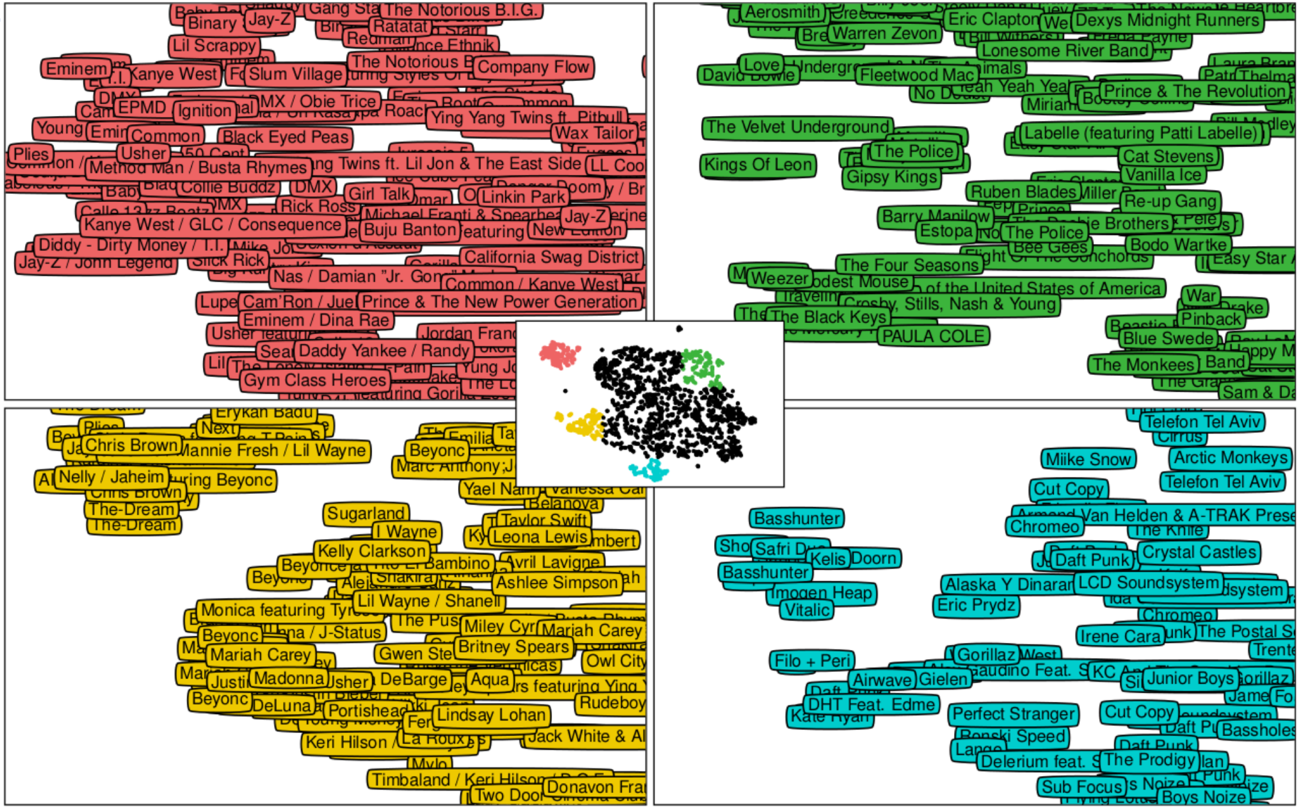
[van den Oord et al., 2013] *Deep Content-Based Music Recommendation*. NIPS workshop.



- E.g. combining matrix factorization with tag-trained neural network to emphasize content in cold-start

[Liang et al., 2015] *Content-Aware Collaborative Music Recommendation Using Pre-Trained Neural Networks*. ISMIR.

Feedback-Transformed Content



[van den Oord et al., 2013] *Deep Content-Based Music Recommendation*. NIPS workshop.