Spectral Transition-Based Playlist Prediction

Chris Billovits, Rahul Prabala, Nipun Agarwala

Departments of Electrical Engineering and Computer Science, Stanford University

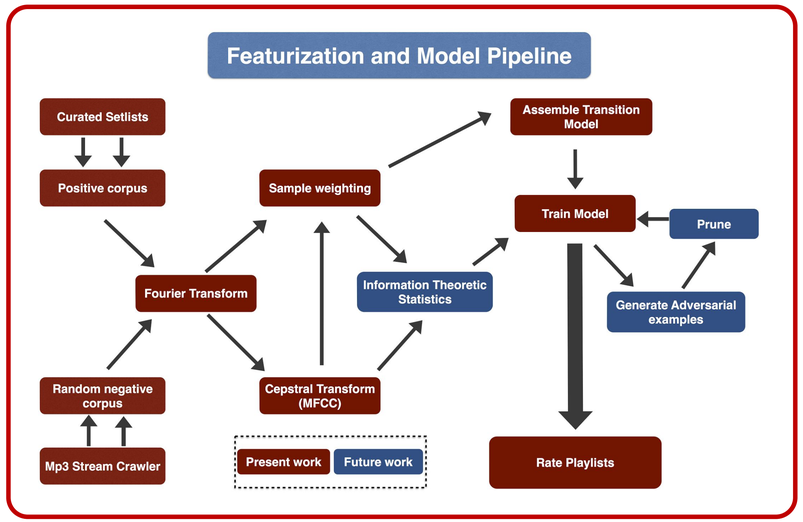
Introduction

Playlist curation is a lucrative field amassing greater interest.

Goal: Predict idealized playlist orderings.

Hypotheses:

1) Spectral methods can leverage vast quantities of data to produce dense musical representations
2) Transitions between songs convey sufficient information for playlist ordering.

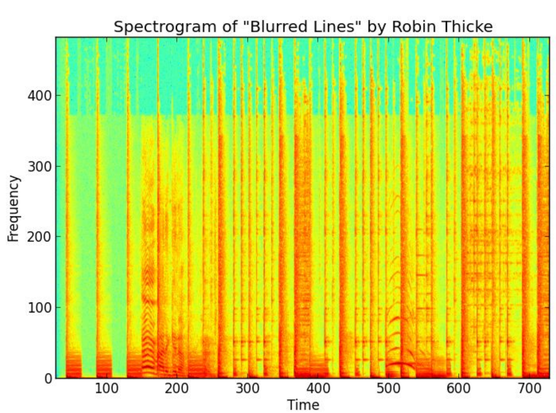


Methodology

- Focus on EDM and House genres, where transitions are vital (**Hypothesis 2)**
- Perform a Fast-Fourier Transform (**FFT**) on windowed samples
  - Vary window length and quantity for different feature sets
- Use Mel-frequency Cephstral Coefficients (**MFCC**s) and Delta coefficients **(Hypothesis 1)**
  - More closely approximate human perception of audio
  - Delta coefficients measure change in MFCC over time
- **Feature Selection**: Choose k FFT/MFCC samples from end of previous song and p samples from the start of the next song
- **Training Set**: Arbitrarily combine randomized adversarial examples with curated playlist transitions
- **Test set**: Predict curated playlists using learned parameters
- **Evaluation metric:** Positive-example F1:
  - Intuition: Good transitions are rare - want to separate most wheat from chaff
  - Less support for good playlist transitions

Challenges

- Different purposes for playlists make prediction noisy
- **Solution:** Use DJ setlists for positive training data
- Unseen negative playlist transitions are not necessarily bad
- **Solution:** Bootstrap off of adversarial examples and minimize regret to reduce model sensitivity



Algorithms

- Naive Logistic Regression:
  - $\arg\min_{\theta} \|y - h_\theta(x)\|^2$
- Ridge Regression to bias towards smaller FFT values:
  - $\arg\min_{\theta} \|y - h_\theta(x)\|^2 + \lambda\|\theta\|^2$
- Hinge Loss, for a max-margin classifier:
  - $\arg\min_{w} \sum_{i=1}^{m} (1 - y_i(w^T x_i)) + \lambda\|w\|^2$
- Shallow Neural Network with $\tan(h)$ activation, least squares error
  - 10-50 hidden layers, randomly initialized bias
  - Biased predictor:

$$h_{W,U}(x) = 1\left[tanh\left(U \cdot tanh\left(Wx + b^{(1)}\right) + b^{(2)}\right) > c\right]; -1 \le c \le 1$$

- Optimization with ADAGRAD - adaptive subgradient method providing per-feature learning rate

Wave Embeddings of Artist-Stratified Playlist Transitions

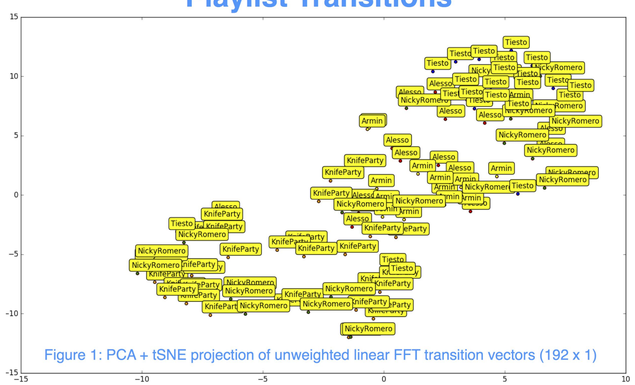


Figure 1: PCA + tSNE projection of unweighted linear FFT transition vectors (192 x 1)

Future Work

- Use GloVe on discretized FFT sequences to find out interesting low-dimensional feature representations
- Investigate LSTM Recurrent Neural Networks on the distance metric for stronger transition-feature correlation and more transition history utilization
- Explore diverse genres and datasets for general applications and results

Findings

Algorithm\Results	Training F1	Test F1
Logistic Regression (Reduced Features)[Baseline]	0.65	0.62
Logistic Regression (Full Features) [Baseline]	0.85	0.82
Hinge Loss (Reduced Features)	0.91	0.8
Hinge Loss (All Features)	0.99	0.87
Shallow Neural Network (Reduced Features)	0.88	0.89
L-BFGS Neural Network (Reduced Features)	**0.94**	**0.94**

- Initial feature size of 200 96-bin FFT samples (0.1s), equally divided across each transition.
- Reduced to ~11 96-bin FFT samples (0.1s), equally divided across transition
- Hinge loss overfits and NN give best results
- NN provides best performance with 1/10th as many samples (!)

Contact: rprabala@stanford.edu
cjbillov@stanford.edu
nipuna1@stanford.edu