

GCT634/AI613: Musical Applications of Machine Learning

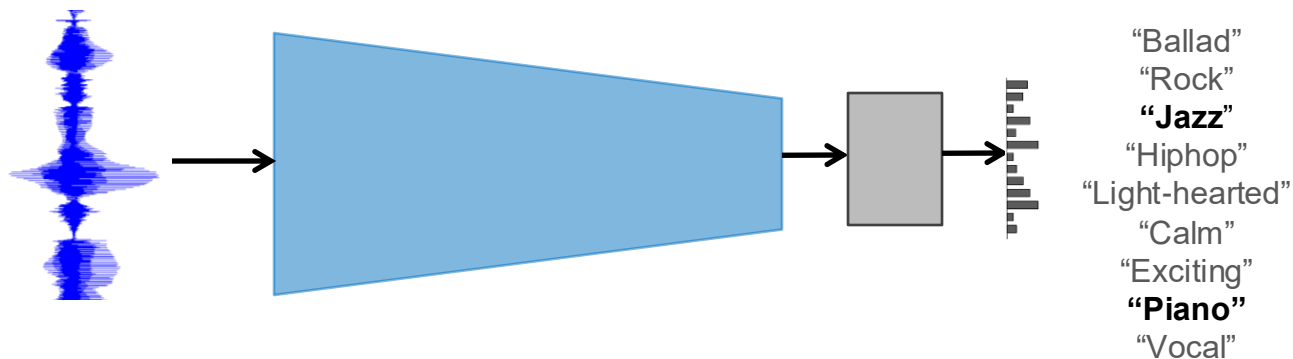
# Music Representation Learning



**Juhan Nam**

# Introduction

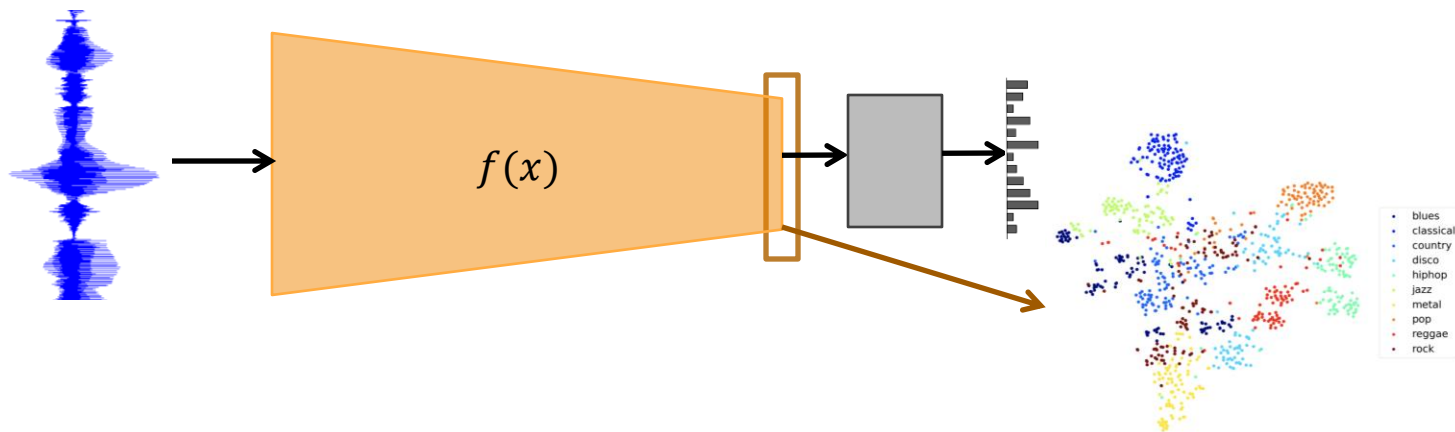
- Problems in music classification (or any supervised learning)
  - Require a **large-scale labeled dataset** to achieve high performance
  - Require more human efforts, time and cost during the data creation process



How can we use deep neural network and achieve high performance with a small labeled dataset?

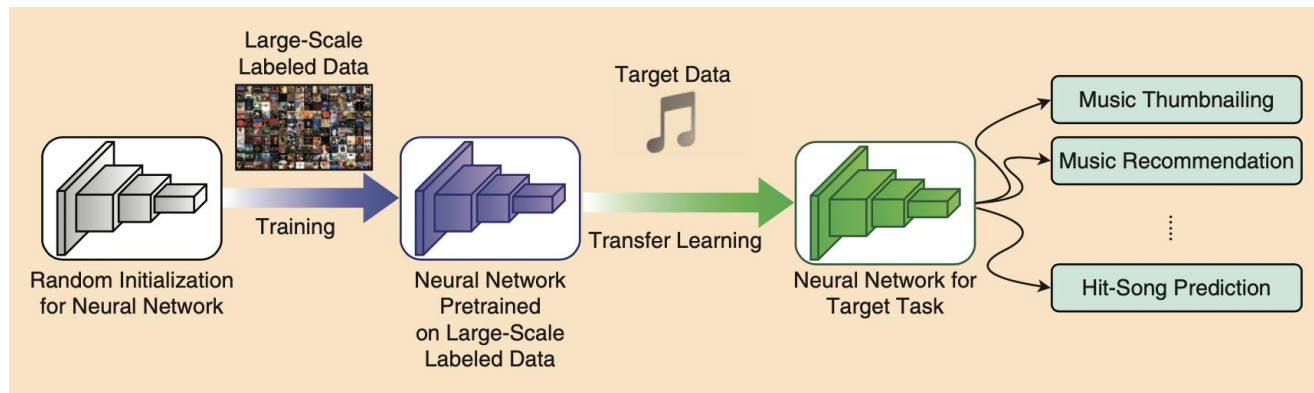
# Introduction

- Use the embedding of **pre-trained neural networks** instead of training the network from scratch
  - Works as a feature extractor (or encoder) that transforms unorganized high-dimensional input to low-dimensional organized embedding vectors
  - The pre-trained model is assumed to be trained with large-scale data and thus it generalizes to unseen data well



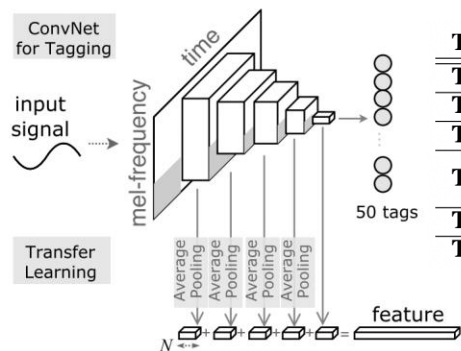
# Transfer Learning

- Transfer knowledge between two tasks
  - Source task: train a neural network with a large-scale dataset and provide a generalized embedding
  - Target task: relevant tasks that re-use the pre-trained network as a feature extractor

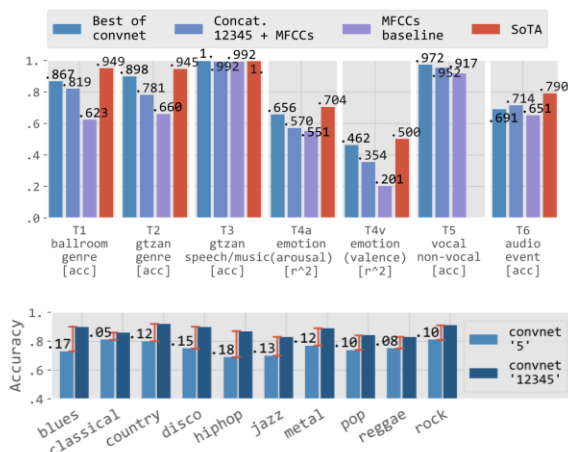


# Transfer Learning for Music Classification

- Source task: music tagging
  - A 2D CNN model is trained using MSD with 250k audio tracks and 50 tags
- Target tasks: 6 different tasks with different data sizes and
  - Extract features from all layers and concatenate the average-pooled outputs
  - Use an SVM for classification or regression
  - Better performance than MFCC



Task	Dataset name	#clips
<b>T1. Ballroom dance genre classification</b>	Extended ballroom [32]	4,180
<b>T2. Genre classification</b>	Gtzan genre [53]	1,000
<b>T3. Speech/music classification</b>	Gtzan speech/music [52]	128
<b>T4. Emotion prediction</b>	EmoMusic (45-second) [46]	744
<b>T5. Vocal/non-vocal classification</b>	Jamendo [41]	4,086
<b>T6. Audio event classification</b>	Urbansound8K [42]	8,732



# Issues

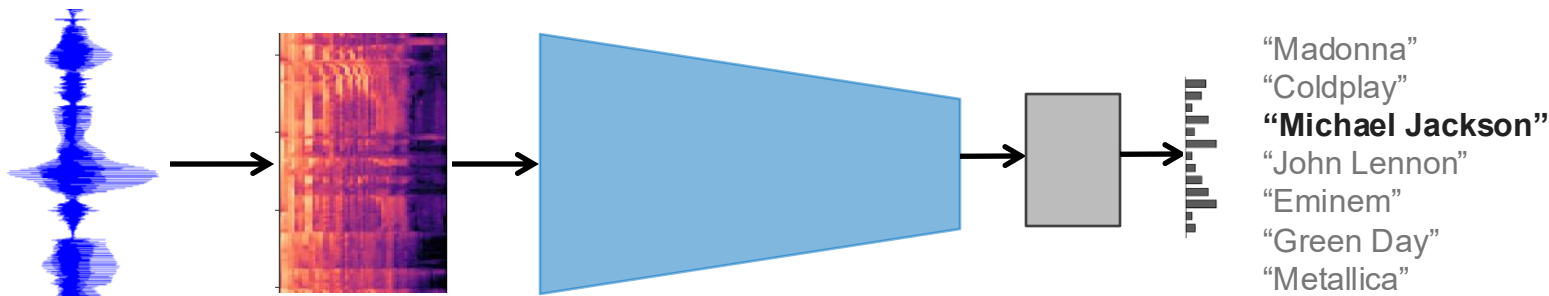
- The neural networks pre-trained with human-labeled datasets are limited
  - Biased to the target task and labels
  - The labels are often noisy and subjective
  - Not easy to extend the data scale
- Solutions
  - Using weak supervision data
  - Self-supervised learning

# Representation Learning with Weak Supervision

- Source of weak supervision: **meta data (artist, album, track), playlist**
  - Easily obtainable from the music catalogue or streaming service
  - Scalable
  - Meta data is factual (or objective)
  
- How can we harness the weak supervision data to train a DNN?

# Representation Learning with Weak Supervision

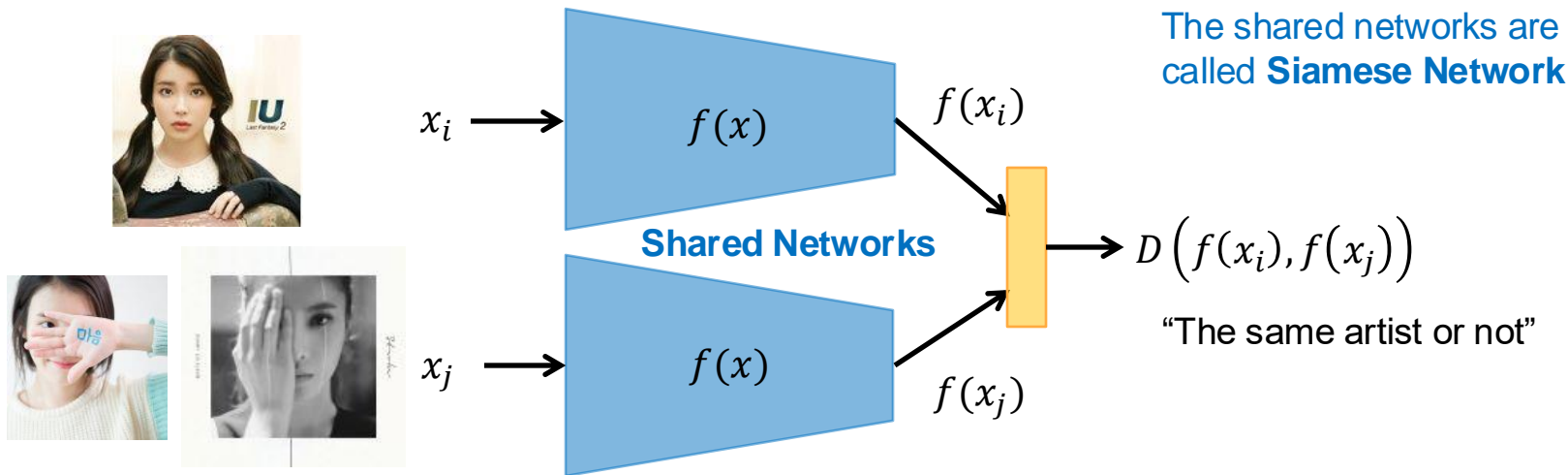
- Let's first take a classification approach with **artist labels**



- Problems
  - The output layer can be excessively large
  - Whenever new artists are added, the model must be trained again

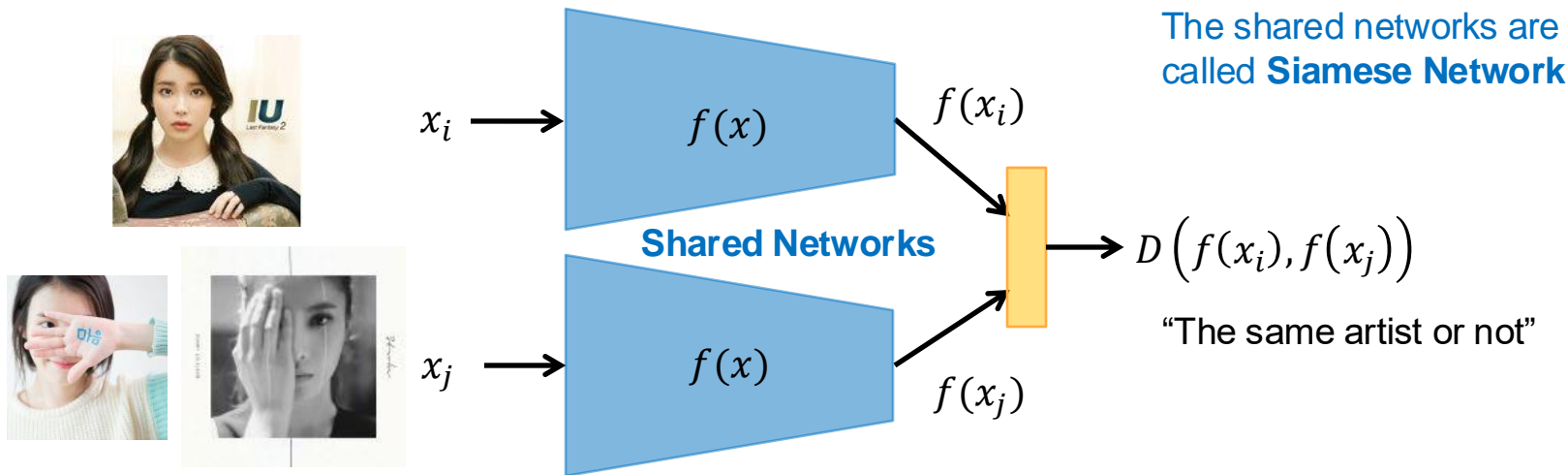
# Metric Learning

- Use **similarity** of input examples to supervise the learning model
  - Similarity is measured by the distance on the embedding space
    - If  $x_i$  and  $x_j$  are from the same class,  $D(f(x_i), f(x_j))$  is small
    - If  $x_i$  and  $x_j$  are from different classes,  $D(f(x_i), f(x_j))$  is large
  - Widely used distance: L2 (Euclidean) or cosine



# Metric Learning

- As a result, we learn the embedding space via the projection function  $f(x)$ 
  - This is also called **contrastive representation learning**



# Metric Learning Loss

- Contrastive loss
- Triplet loss
- NCE / InfoNCE
- Lifted Structured Loss
- N-pair Loss
- Soft-Nearest Neighbors Loss

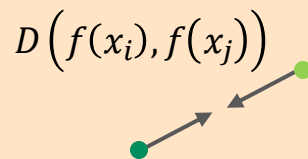
See the nice blog article: <https://lilianweng.github.io/posts/2021-05-31-contrastive/>

# Contrastive loss

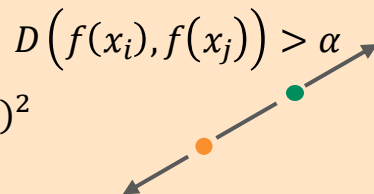
- A binary classification loss

$$L_f(x_i, x_j, y) = (1 - y) \cdot \frac{1}{2} D(f(x_i), f(x_j))^2 + y \cdot \frac{1}{2} (\max(0, \alpha - D(f(x_i), f(x_j))))^2 \quad \alpha: \text{margin}$$

If  $x_i$  and  $x_j$  are from the same class ( $y = 0$ ),  $L_f = \frac{1}{2} D(f(x_i), f(x_j))^2$



If  $x_i$  and  $x_j$  are from different classes ( $y = 1$ ),  $L_f = \frac{1}{2} (\max(0, \alpha - D(f(x_i), f(x_j))))^2$

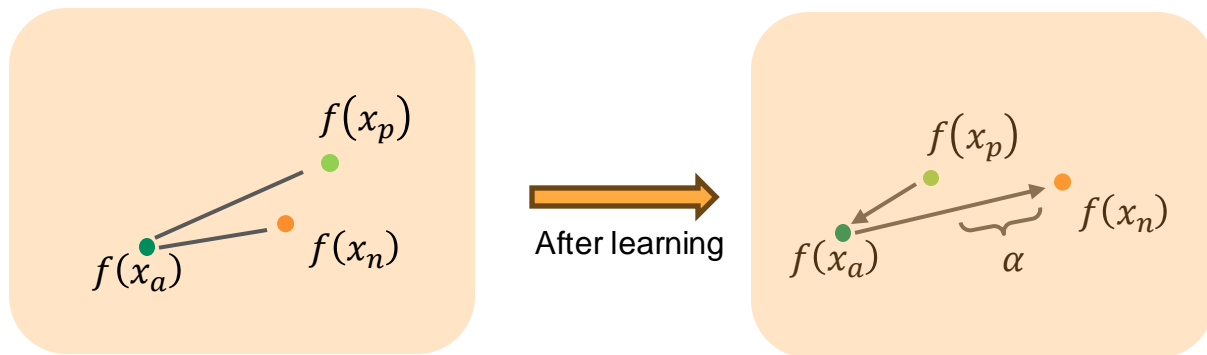


# Triplet loss

- A ranking loss

$$L_f(x_a, x_p, x_n) = \max(0, D(f(x_a), f(x_p)) - D(f(x_a), f(x_n)) + \alpha) \quad \alpha: \text{margin}$$

- Multiple negative pairs are possible
- Also called maximum margin hinge loss



# InfoNCE loss

- Use a single positive pair and multiple negative pairs in the single batch
- Similar to multi-class logistic regression
  - Form a multinomial distribution using the distances of a single positive pair and multiple negative pairs

$$P(x_p | X, c) = \frac{e^{-D(f(x_a), f(x_p)) / \tau}}{\sum_i e^{-D(f(x_a), f(x_i)) / \tau}}$$

$\tau$ : margin

$X$ : all pairs in a batch

$c$ : context about the positive and negative

$-D(f(x_a), f(x_p)) / \tau$

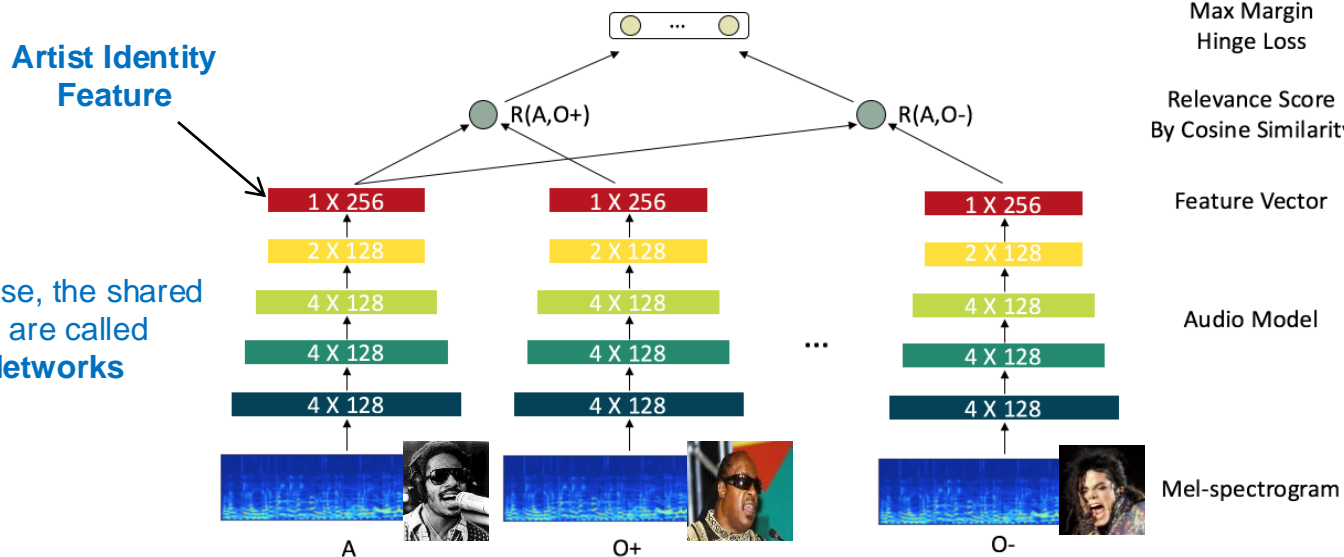
: score function

- The loss is the negative log likelihood

$$L_f(X) = - \sum \log \frac{e^{-D(f(x_a), f(x_p)) / \tau}}{\sum_i e^{-D(f(x_a), f(x_i)) / \tau}}$$

# Representation Learning with Artist labels

- Use the triplet loss
  - The positive pair is from the same artist and the negative pair is from two different artists



Max Margin  
Hinge Loss

- 4 negative samples  
- margin= 0.4

Relevance Score  
By Cosine Similarity

$$\cos(y_A, y_O) = \frac{y_A^T y_O}{|y_A| |y_O|}$$

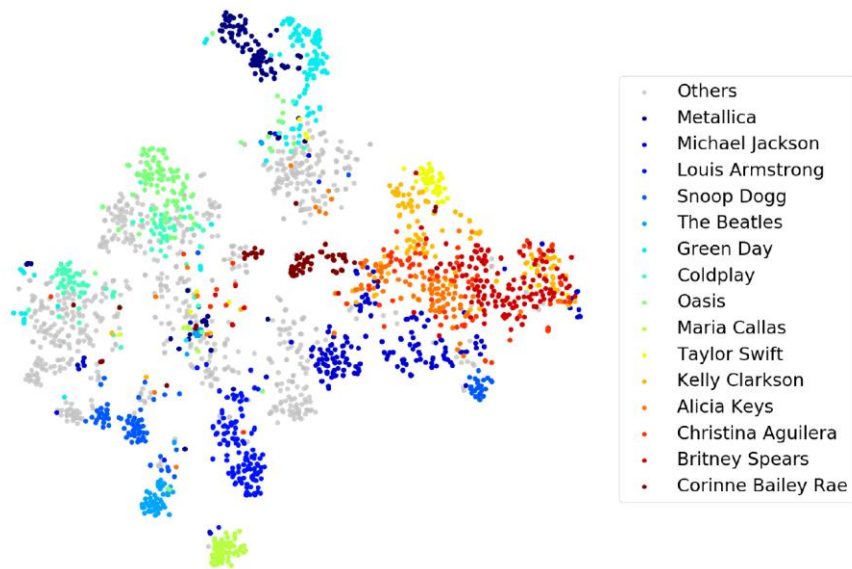
Feature Vector

Audio Model

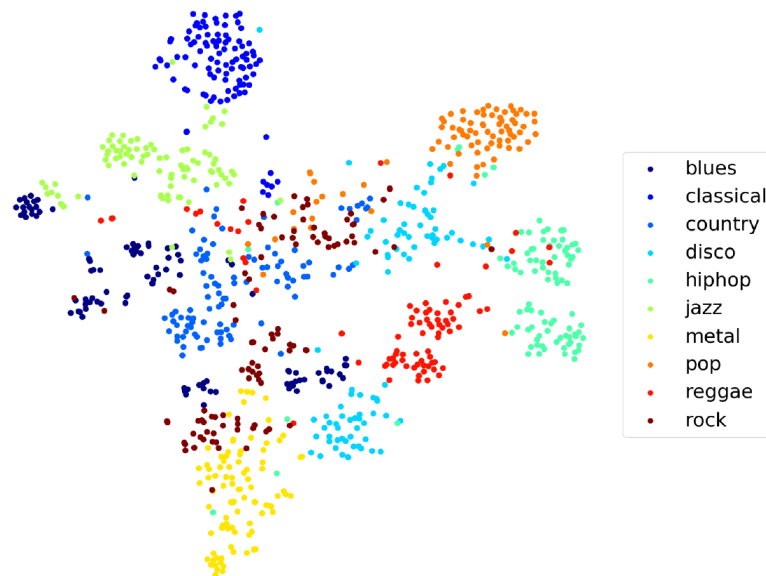
Mel-spectrogram

# Representation Learning with Artist labels

- 2D plots of the embedding space by t-SNE



Artist Distribution



Genre distribution (Transfer Learning)

# Artist-Level Music Retrieval

## Query

0 (1.0) Eminem

HIP-HOP

- 1 (0.9182) OutKast
- 2 (0.9027) Spezializtz
- 3 (0.9012) Dino MC 47
- 4 (0.8878) Bone Thugs-N-Harmony
- 5 (0.8840) Method Man
- 6 (0.8814) Big Punisher
- 7 (0.8761) Cor Veleno
- 8 (0.8642) Lil' Wayne
- 9 (0.845) Obie Trice
- 10 (0.8301) mcenroe & Birdapres

0 (1.0) Green Day

Rock Band

- 1 (0.8728) Unwritten Law
- 2 (0.8692) P.O.D.
- 3 (0.8237) Tokyo Rose
- 4 (0.7903) Linkin Park
- 5 (0.7532) All Star United
- 6 (0.7532) Hinder
- 7 (0.7420) The All-American Rejects
- 8 (0.7384) Jimmy Eat World
- 9 (0.7206) Third Day
- 10 (0.7000) Bon Jovi

0 (1.0) John Lennon

Pop rock

- 1 (0.8355) Cliff Richard
- 2 (0.8199) The Who
- 3 (0.7959) Status Quo
- 4 (0.7836) Nick Cave and the Bad Seeds
- 5 (0.7772) T.Love
- 6 (0.7713) Blake Morgan
- 7 (0.7538) Radney Foster
- 8 (0.7509) Coldplay
- 9 (0.7409) The Beach Boys
- 10 (0.7391) Badly Drawn Boy

## Query

0 (1.0) EXO

Boy group

- 1 (0.9554) 세븐틴
- 2 (0.9496) 2PM
- 3 (0.9484) NCT 127
- 4 (0.9422) 몬스타엑스
- 5 (0.9391) GOT7 (갓세븐)
- 6 (0.9373) 비투비 (BTOB)
- 7 (0.9344) 초신성
- 8 (0.9314) 비스트 (Beast)
- 9 (0.9273) B.A.P
- 10 (0.9241) 샤이니 (SHINee)

0 (1.0) 들국화

80's

- 1 (0.9199) 동물원
- 2 (0.9153) 조하문
- 3 (0.8833) 푸른하늘
- 4 (0.8816) 김현식
- 5 (0.8607) 해바라기
- 6 (0.8492) 유재하
- 7 (0.8468) 김현철
- 8 (0.8458) 김민우
- 9 (0.8427) 피노키오
- 10 (0.8394) 한동준

0 (1.0) 다이내믹 듀오(Dynamic Duo)

HIP-HOP

- 1 (0.9600) 드렁큰타이거
- 2 (0.9456) 소울 다이브(Soul Dive)
- 3 (0.9421) 개코
- 4 (0.9334) 언터처블
- 5 (0.9310) 허클베리피(Huckleberry P)
- 6 (0.9291) 스윙스(Swings)
- 7 (0.9242) 개리
- 8 (0.9218) 방탄소년단
- 9 (0.9201) Simon Dominic
- 10 (0.9190) 빈지노(Beenzino)

# Song-Level Music Retrieval

Query

0 (1.0) bob marley and the wailers - three little birds



1 (0.7639) dennis brown - tribulation



2 (0.6474) junior murvin - police and thieves



3 (0.5599) specials - gangsters

4 (0.4997) shuggie otis - sweet thang

5 (0.4782) james brown - give it up or turnit a loose

Query

0 (1.0) norah jones - dont know why



1 (0.7092) dionne warwick - walk on by



2 (0.6981) jewel - enter from the east



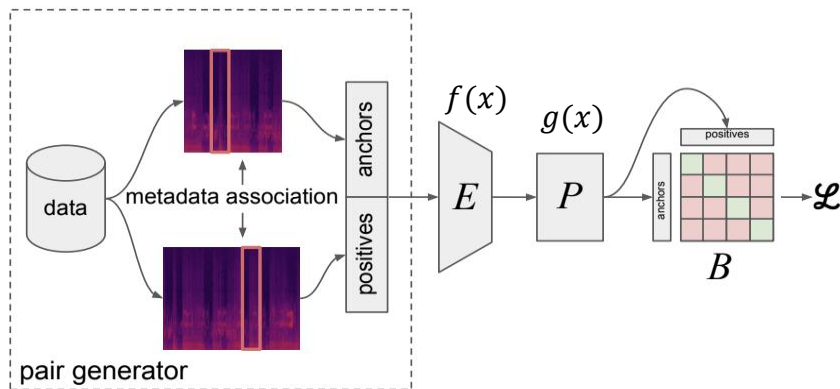
3 (0.6853) shakira - the one

4 (0.6700) mamas and the papas - words of love

5 (0.6602) andrews sisters - boogie woogie bugle boy

# Representation Learning with Track, Album, and Artist Labels

- “Encoder + projector” architecture and bilinear similarity from COLA
  - Encoder: used for downstream tasks
  - Projector: used for measuring similarity in pre-training
  - No explicit anchor/negative pairs → use non-associated pairs in the batch
  - Use meta data from **Discogs** ([www.discogs.com](http://www.discogs.com))



Bilinear similarity:

$$s(x, x') = g(f(x))^T W g(f(x'))$$

$$\mathcal{L} = -\log \frac{\exp(s(x, x^+))}{\sum_{x^- \in \mathcal{X}^-(x) \cup \{x^+\}} \exp(s(x, x^-))}$$

Loss:

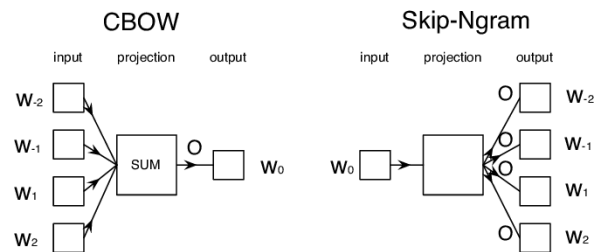
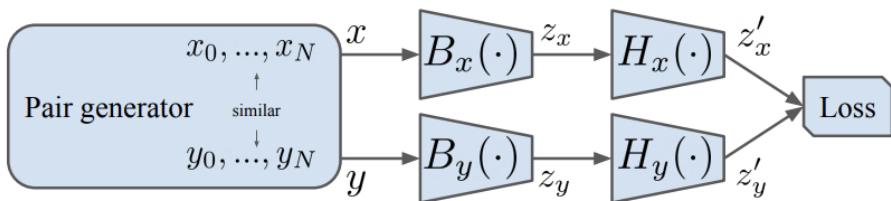
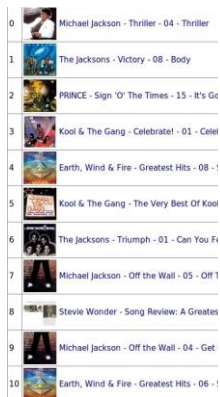
# Representation Learning with Track, Album, and Artist Labels

- Downstream task results

		MTG-Jamendo										
		Genre		Instrument		Mood		Top50		MTAT		FMA
		ROC	PR	ROC	PR	ROC	PR	ROC	PR	ROC	PR	Acc.
Audio-Text Joint Embedding	Lileonardo	-	-	-	-	<b>77.5</b>	<b>15.1</b>	-	-	-	-	-
	Harmoic CNN	-	-	-	-	-	-	83.2	29.8	*91.3	*45.9	-
	MusiCNN	-	-	-	-	-	-	-	-	90.7	38.4	-
	MuLaP	85.9	-	76.8	-	76.1	-	82.8	-	*89.3	*40.2	<b>61.1</b>
	CALM	-	-	-	-	-	-	-	-	<b>91.5</b>	<b>41.4</b>	-
Jukebox Encoder (VQ-VAE)	Random weights	50.7	3.1	49.9	6.4	50.4	3.4	48.3	6.5	50.0	5.3	12.5
	Style tags	87.7	19.9	77.6	19.8	75.6	13.6	83.1	29.7	90.2	37.4	59.1
	VGGish	86.3	17.2	<b>77.8</b>	<b>20.2</b>	76.3	14.1	83.2	28.2	90.2	37.2	53.0
	Track associations	86.3	18.0	69.9	16.7	74.0	12.8	82.9	29.4	89.7	36.4	58.9
	Release associations	86.9	18.9	71.9	17.2	72.8	11.7	83.2	29.8	90.3	37.1	60.9
Artist associations	<b>87.7</b>	<b>20.3</b>	69.7	16.9	76.3	14.3	<b>83.6</b>	<b>30.6</b>	90.7	38.0	59.1	
Label associations	87.0	19.4	75.0	18.2	74.8	12.8	83.1	29.9	88.7	34.2	59.5	
Stack	86.9	19.4	74.7	18.8	74.3	13.0	83.4	30.0	90.8	38.6	59.8	
Multi-task	87.2	19.9	70.5	17.2	76.1	14.4	83.5	30.3	90.8	37.8	60.0	

# Representation Learning with Playlists

- Use co-occurrences between two tracks in music playlists
  - Simple co-occurrences:  $N/2$  pairs ( $N$ : the number of tracks in a playlist)
  - Top-10 co-occurrences: use 10 most occurring tracks only across playlists
  - Word2Vec: regard **a playlist** as a **sentence**
    - Used the affinity within a context window: Continuous Bag of Word (CBOW)



# Representation Learning with Playlists

- Downstream task results

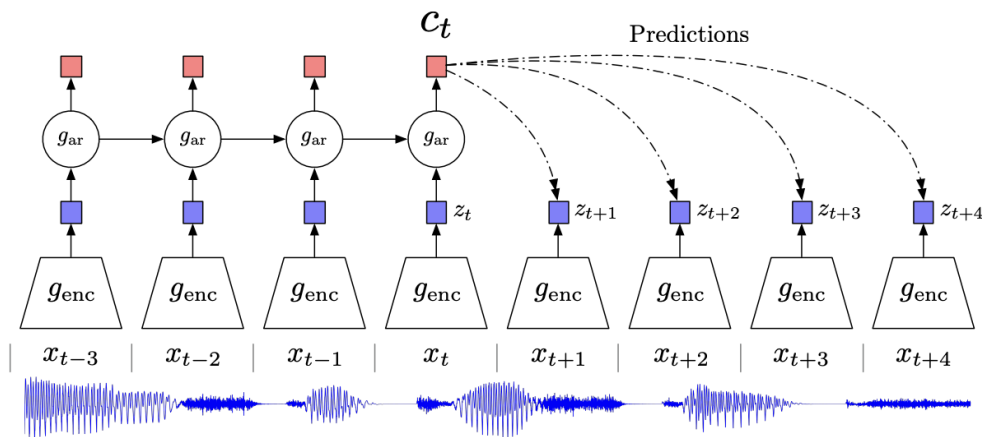
Dataset	Genre		Instrument		Mood		MTAT		Genre Internal	
	AP	ROC	AP	ROC	AP	ROC	AP	ROC	AP	ROC
VGGish										
<i>VGGish FT</i>	15.8±0.3	84.9±0.5	18.1±0.6	74.2±1.2	12.1±0.9	72.7±0.8	44.4±0.6	90.6±0.1	-	-
<i>From Scratch</i>	13.4±0.2	82.6±0.3	15.5±0.4	72.1±0.5	9.3±0.2	70.6±0.4	40.2±0.7	88.9±0.2	54.3±0.3	96.7±0.0
<i>SimCLR</i>	15.2±0.3	83.6±0.4	16.4±0.3	72.1±0.5	10.7±0.2	70.1±0.2	41.1±0.6	88.9±0.2	61.7±0.1	97.6±0.0
<i>Artist CO</i>	17.3±0.1	85.6±0.1	20.4±0.4	76.7±0.1	13.9±0.2	74.5±0.5	46.2±0.1	91.1±0.1	68.8±0.2	98.3±0.1
<i>Playlist CO</i>	17.0±0.1	85.4±0.2	20.2±0.4	76.1±0.3	13.3±0.8	73.8±0.8	45.9±0.2	90.9±0.0	67.7±0.7	98.2±0.1
<i>Playlist TCO</i>	17.5±0.1	84.9±0.4	20.5±0.3	76.3±0.9	13.8±0.3	73.7±0.7	45.8±0.3	91.0±0.1	70.0±0.4	98.4±0.0
<i>Playlist W2V</i>	17.3±0.2	85.5±0.3	19.8±0.7	75.3±0.2	13.5±0.1	72.8±0.7	45.3±0.6	90.9±0.2	69.8±0.1	98.4±0.0
Resnet50										
<i>From Scratch</i>	14.4±0.2	82.9±0.1	15.6±0.5	71.2±0.6	8.9±0.0	69.2±0.4	40.7±0.3	88.8±0.1	63.3±0.1	97.7±0.1
<i>SimCLR</i>	16.3±0.2	84.7±0.3	17.4±0.1	73.3±0.7	12.1±0.3	73.0±0.3	43.4±0.5	90.1±0.2	67.4±0.3	98.2±0.0
<i>Artist CO</i>	<b>19.0±0.1</b>	85.0±0.1	21.1±0.4	76.5±0.9	14.9±0.3	74.8±0.7	47.0±0.3	<b>91.5±0.2</b>	73.4±0.2	98.6±0.1
<i>Playlist CO</i>	18.7±0.7	<b>85.7±0.4</b>	<b>21.2±0.7</b>	76.7±0.9	14.8±0.5	74.2±0.4	46.8±0.2	91.4±0.0	73.4±0.1	98.6±0.0
<i>Playlist TCO</i>	18.9±0.2	85.1±0.3	20.4±0.7	75.4±1.5	14.3±0.4	73.7±0.7	<b>47.0±0.2</b>	91.3±0.2	72.8±0.2	98.6±0.0
<i>Playlist W2V</i>	<b>19.0±0.1</b>	85.4±0.3	20.7±0.4	<b>77.1±0.4</b>	<b>15.0±0.1</b>	<b>75.1±0.4</b>	46.7±0.4	91.2±0.2	<b>74.1±0.2</b>	<b>98.7±0.1</b>

# Self-Supervised Learning (SSL)

- Learn a “good” representation from unlabeled data via “pretext” tasks and use the learned representations for downstream tasks
  - The pretext tasks requires automatically generated labels or supervision
- Types of automatically generated supervision
  - Temporal proximity
  - Data augmentation with digital audio effects or masking

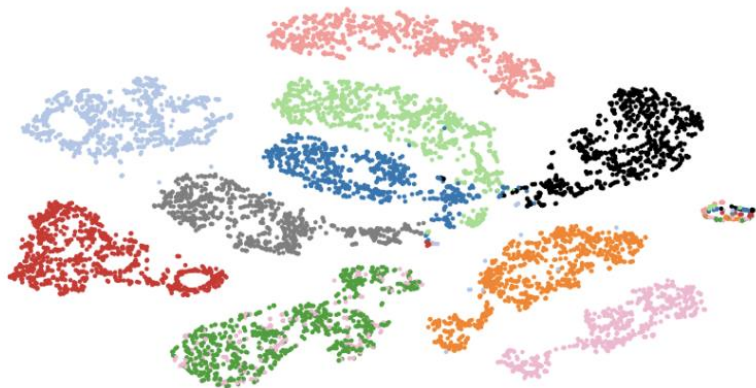
# Contrastive Predictive Coding (CPC)

- Use the temporal proximity between a context vector ( $C_t$ ) and the future encoded vectors ( $z_{t+k}$ ) in the sequence
  - $C_t$  is an autoregressive summary up to the current encoded vectors
  - Use the infoNCE loss with the score function:  $s_k(z_{t+k}, C_t) = z_{t+k}^T W_k C_t$
  - CPC can be applied to images using the spatial affinity



# Contrastive Predictive Coding (CPC)

- Pretask dataset: a 100 hour subset of LibriSpeech
- Learn well-organized feature representations



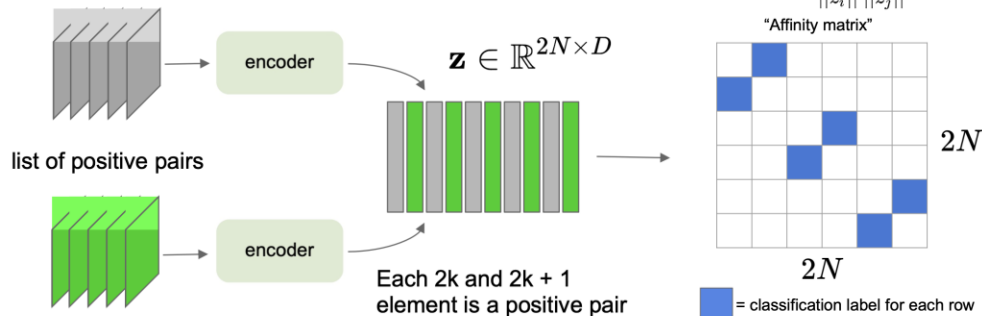
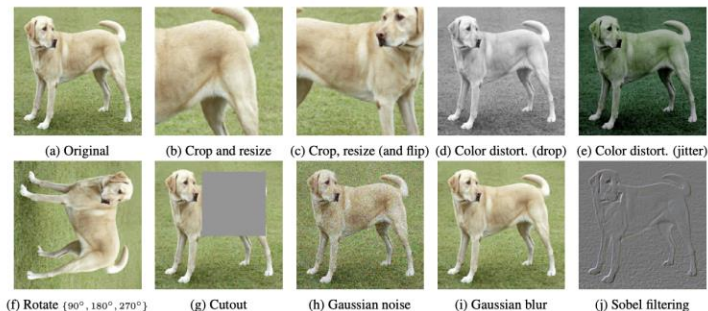
t-SNE visualization of audio embedding  
from 10 speakers

Method	ACC
<b>Phone classification</b>	
Random initialization	27.6
MFCC features	39.7
CPC	64.6
Supervised	74.6
<b>Speaker classification</b>	
Random initialization	1.87
MFCC features	17.6
CPC	97.4
Supervised	98.5

Linear classification on the CPC  
embedding

# SimCLR

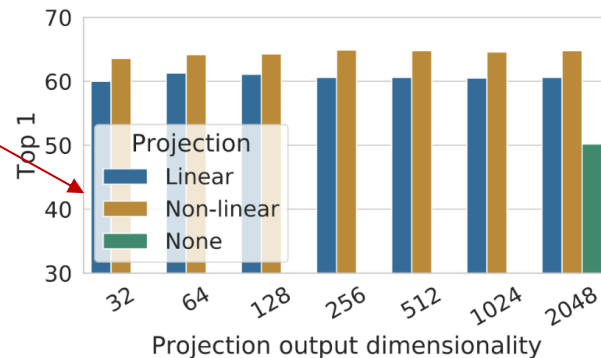
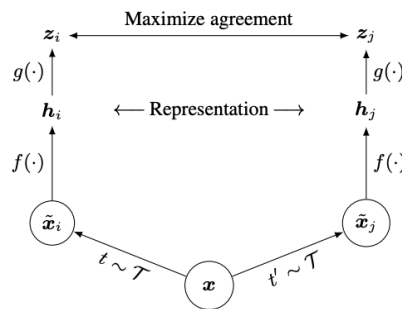
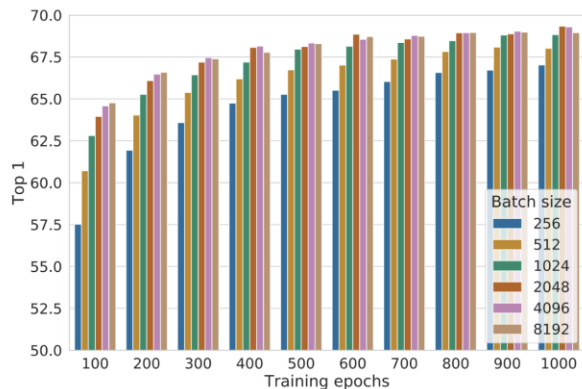
- Use random data augmentation within a minibatch
  - The original and its transformed examples are used as positive pairs
  - Outperforms supervised approaches using only a linear classifier on SimCLR features



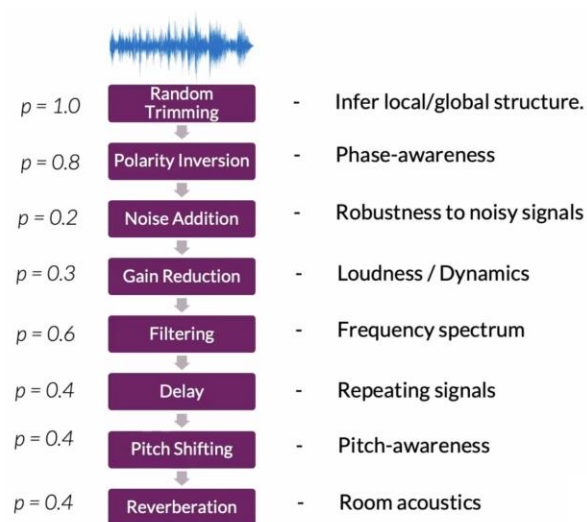
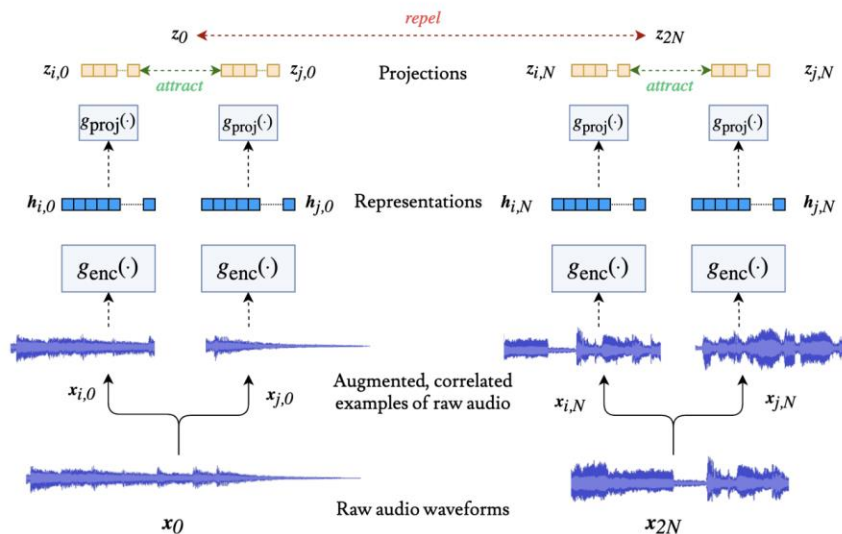
Source: Stanford cs231n

# SimCLR

- Using a large batch size is crucial to achieve high performance
  - But, it uses more memory on GPU
- Linear or non-linear projection improves the model performance
  - $z_j$  is specific to the “pretext” task and may discard useful information for downstream tasks, whereas  $h_j$  preserves more information



- A music version of SimCLR
  - Use a chain of random digital audio effects considering the order
  - Use SampleCNN as a backbone model



- Evaluation results (linear probing)

Model	Dataset	ROC-AUC	PR-AUC
CLMR (ours)	MTAT	88.7 ( <b>89.3</b> )	35.6 ( <b>36.0</b> )
Musicnn [5] <sup>†</sup>	MTAT	89.0	34.9
SampleCNN [26] <sup>†</sup>	MTAT	88.6	34.4
CPC (ours)	MTAT	86.6 (88.0)	31.0 (33.0)
1D CNN [36] <sup>†</sup>	MTAT	85.6	29.6
Transformer [37] <sup>†§</sup>	MSD	<b>89.7</b>	<b>34.8</b>
Musicnn [5] <sup>†</sup>	MSD	88.0	28.7
SampleCNN [26] <sup>†</sup>	MSD	87.9	28.5
CLMR (ours)	MSD	85.7	25.0

Comparison of CLMR, CPC, and Fully-supervised CNN:  
 - CLMR and CPC use linear classification on the fixed encoder.

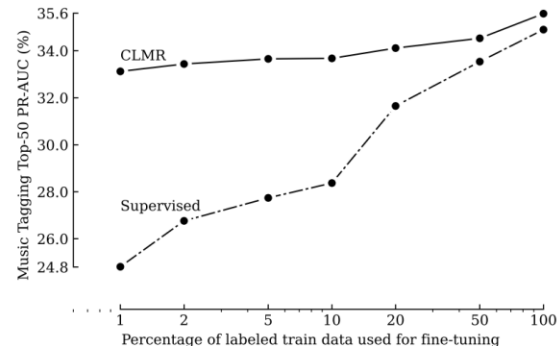
Transform	Tag		Clip	
	ROC-AUC	PR-AUC	ROC-AUC	PR-AUC
Filter	87.6	33.3	92.5	67.9
Reverb	86.5	31.7	91.8	65.8
Polarity	86.3	31.5	91.7	65.7
Noise	86.1	31.5	91.5	65.5
Pitch	86.4	31.5	91.5	65.3
Gain	86.2	31.1	91.5	65.1
Delay	85.8	30.5	91.3	64.9
Crop	85.8	30.5	91.3	64.8

**Table 2:** CLMR music tagging performance using a random crop together with one other audio data augmentation.

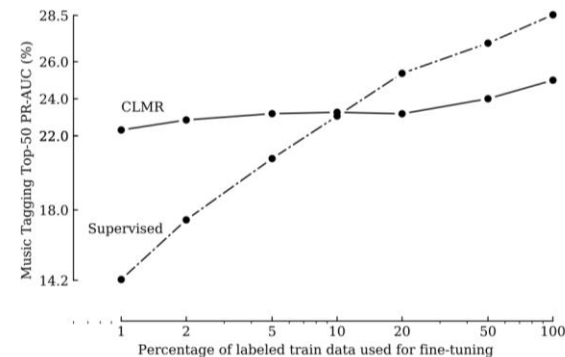
- Data efficient classification experiments

Model	Train Dataset	ROC-AUC <sub>TAG</sub>	PR-AUC <sub>TAG</sub>
CLMR	MSD	<b>87.8</b>	<b>33.1</b>
CPC	FMA	86.3 (87.8)	30.7 (32.5)
CLMR	FMA	86.2 (86.6)	30.6 (31.2)
CPC	Billboard	85.8 (86.3)	29.7 (30.2)
CPC	GTZAN	83.4 (86.0)	26.9 (29.7)
CLMR	Billboard	82.7 (84.2)	26.9 (27.8)
CLMR	GTZAN	81.9 (85.4)	26.2 (29.5)

Unseen music tagging datasets



**Figure 4:** Percentage of labels used for training vs. the achieved PR – AUC<sub>TAG</sub> score on the MTAT dataset.

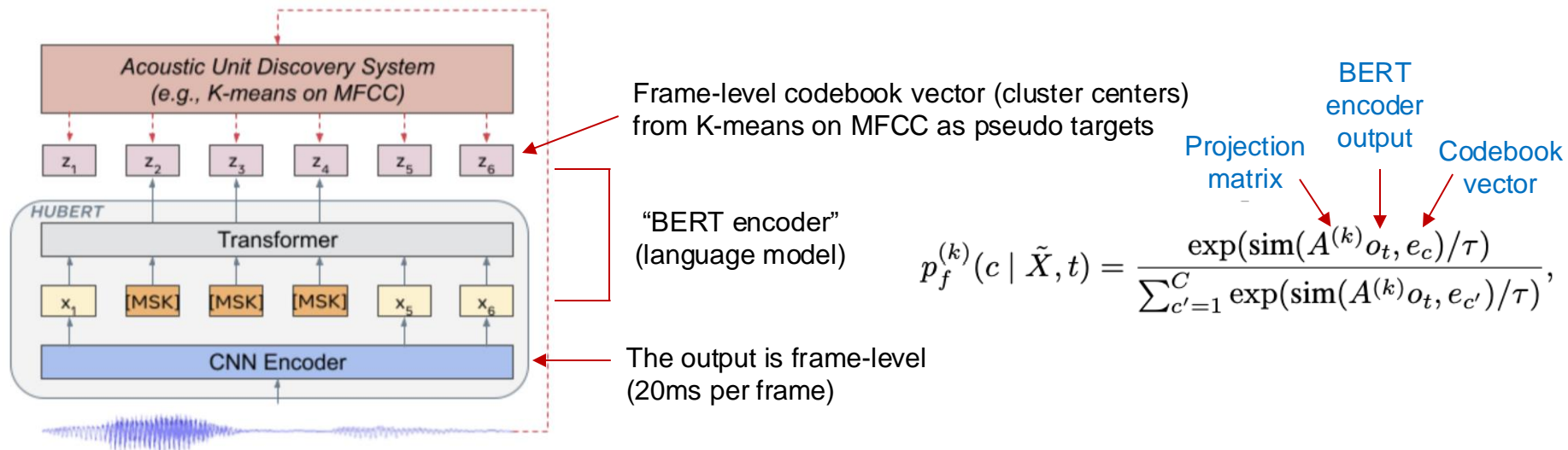


**Figure 5:** Percentage of labels used for training vs. the achieved PR – AUC<sub>TAG</sub> score on the MSD.

- PyTorch Tutorial on CLMR
  - [https://music-classification.github.io/tutorial/part5\\_beyond/self-supervised-learning.html](https://music-classification.github.io/tutorial/part5_beyond/self-supervised-learning.html)

# HuBERT

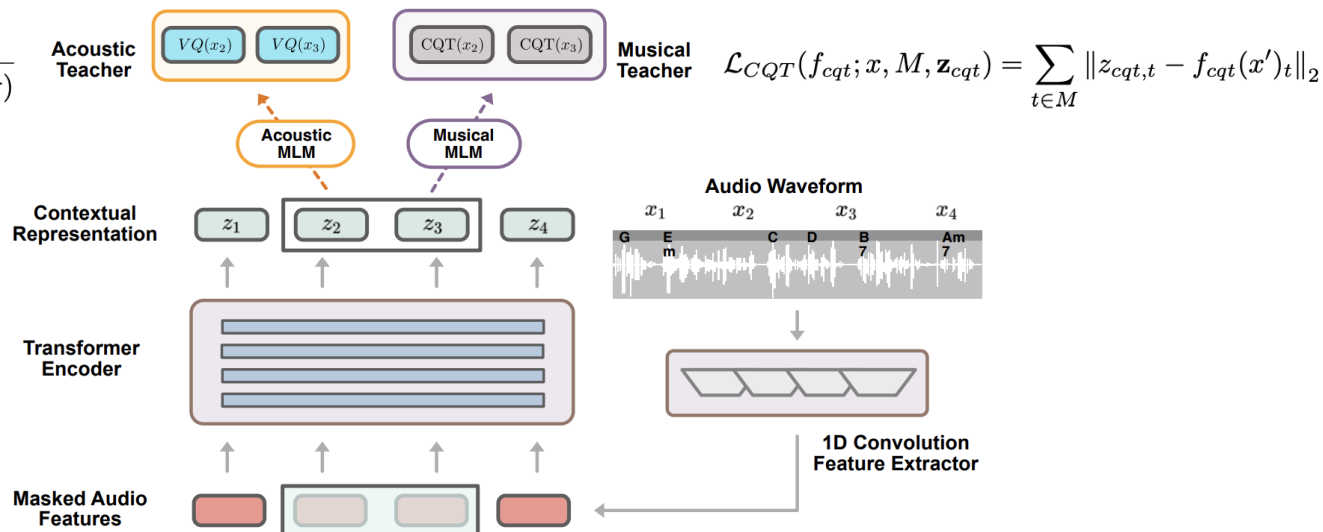
- A masked language model on frame-level CNN audio encoder
  - Learn both acoustic representation and language modeling by predicting frame-level features from masked frames via BERT
  - An ensemble of K-means to cover different granularity of phonetic features



# MERT

- A music version of HuBERT using two teachers
  - Acoustic teacher: EnCodec (VQ-VAE) → InforNCE loss
  - Musical teacher: constant-Q transform → L2 regression loss

$$p_f(c | x', t) = \frac{\exp(\text{sim}(T(o_t), e_c)/\tau)}{\sum_{c'=1}^C \exp(\text{sim}(T(o_t), e_{c'})/\tau)}$$



- Downstream tasks (1/2)

Dataset Task	MTT Tagging		GS Key	GTZAN Genre	GTZAN Rhythm	EMO Emotion		Nsynth		VocalSet Tech	VocalSet Singer
	ROC	AP	Acc <sup>Refined</sup>	Acc	F1 <sup>beat</sup>	R2 <sup>V</sup>	R2 <sup>A</sup>	Acc	Acc	Acc	Acc
MusiCNN [41]	90.6*	38.3*	12.8*	79.0*	-	46.6*	70.3*	72.6	64.1	70.3	57.0
CLMR [48]	89.4*	36.1*	14.9*	68.6*	-	45.8*	67.8*	67.9	47.0	58.1	49.9
Jukebox-5B [15; 57]	91.5*	<b>41.4*</b>	66.7*	79.7*	-	<b>61.7*</b>	72.1*	70.4	91.6	76.7	82.6
MULE [36]	91.4*	40.4*	66.7*	73.5*	-	57.7*	70.0*	74.0*	89.2*	75.5	<b>87.5</b>
HuBERT-base <sup>music</sup> [25]	90.2	37.7	14.7	70.0	<b>88.6</b>	42.1	66.5	69.3	77.4	65.9	75.3
data2vec-base <sup>music</sup> [2]	90.0	36.2	50.6	74.1	68.2	52.1	71.0	69.4	93.1	71.1	81.4
MERT-95M <sup>K-means</sup>	90.6	38.4	65.0	78.6	88.3	52.9	69.9	71.3	92.3	74.6	77.2
MERT-95M-public <sup>K-means</sup>	90.7	38.4	67.3	72.8	88.1	59.7	72.5	70.4	92.3	75.6	78.0
MERT-95M <sup>RVQ-VAE</sup>	91.0	39.3	63.5	78.6	88.3	60.0	<b>76.4</b>	70.7	92.6	74.2	83.7
MERT-330M <sup>RVQ-VAE</sup>	91.3	40.2	65.6	79.3	87.9	61.2	74.7	72.6	<b>94.4</b>	<b>76.9</b>	87.1
(Previous) SOTA	<b>92.0</b> [26]	<b>41.4</b> [15]	<b>74.3</b> [30]	<b>83.5</b> [36]	80.6 [24]	<b>61.7</b>	72.1 [15]	<b>78.2</b> [53]	89.2 [36]	65.6 [55]	80.3 [39]

- Downstream tasks (2/2)

Dataset Task	MTG Instrument		MTG MoodTheme		MTG Genre		MTG Top50		MUSDB Source Separation				Avg.
	ROC	AP	ROC	AP	ROC	AP	ROC	AP	SDR <sup>vocals</sup>	SDR <sup>drums</sup>	SDR <sup>bass</sup>	SDR <sup>other</sup>	
MusiCNN [41]	74.0	17.2	74.0	12.6	86.0	17.5	82.0	27.5	-	-	-	-	-
CLMR [48]	73.5	17.0	73.5	12.6	84.6	16.2	81.3	26.4	-	-	-	-	-
Jukebox-5B [15; 57]	-	-	-	-	-	-	-	-	5.1*	4.9*	4.1*	2.7*	-
MULE [36]	76.6	19.2	78.0	15.4	<b>88.0</b>	<b>20.4</b>	83.7	30.6	-	-	-	-	-
HuBERT-base <sup>music</sup> [25]	75.5	17.8	76.0	13.9	86.5	18.0	82.4	28.1	4.7	3.7	1.8	2.1	55.8
data2vec-base <sup>music</sup> [2]	76.1	19.2	76.7	14.3	87.1	18.8	83.0	29.2	5.5	5.5	4.1	3.0	59.9
MERT-95M <sup>K-means</sup>	77.2	19.6	75.9	13.7	87.0	18.6	82.8	29.4	5.6	5.6	4.0	3.0	62.9
MERT-95M-public <sup>K-means</sup>	77.5	19.6	76.2	13.3	87.2	18.8	83.0	28.9	5.5	5.5	3.7	3.0	63.0
MERT-95M <sup>RVQ-VAE</sup>	77.5	19.4	76.4	13.4	87.1	18.8	83.0	28.9	5.5	5.5	3.8	3.1	63.7
MERT-330M <sup>RVQ-VAE</sup>	78.1	19.8	76.5	14.0	86.7	18.6	83.4	29.9	5.3	5.6	3.6	3.0	<b>64.7</b>
(Previous) SOTA	<b>78.8</b>	<b>20.2</b> [1]	<b>78.6</b>	<b>16.1</b> [36]	87.7	20.3 [1]	<b>84.3</b>	<b>32.1</b> [36]	<b>9.3</b>	<b>10.8</b>	<b>10.4</b>	<b>6.4</b> [44]	64.5

- Comparison with different acoustic teachers

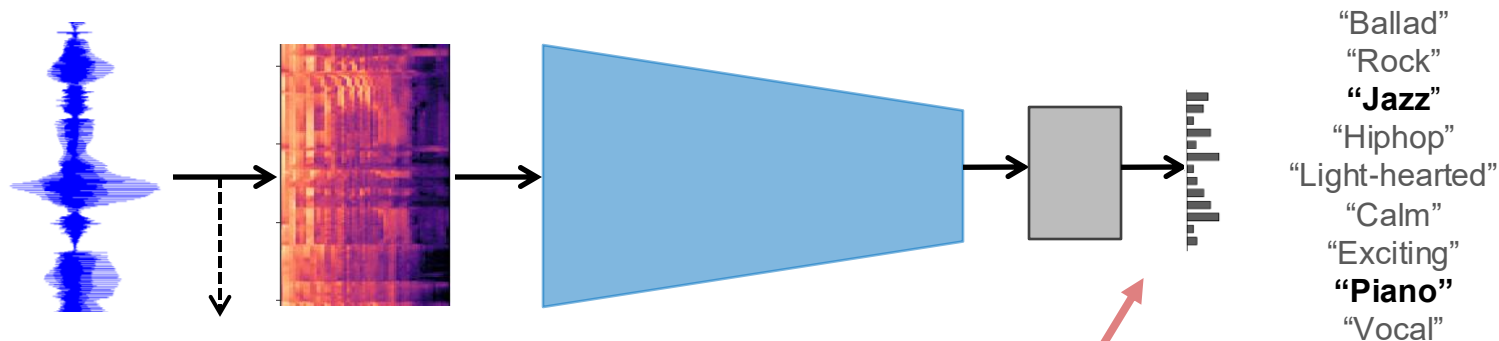
Acoustic Teacher	Acoustic Target Class	Musical Teacher	MTT Tagging		GS Key	GTZAN Genre	EMO Emotion		Avg.
			ROC	AP	Acc <sup>Refined</sup>	Acc	R2 <sup>V</sup>	R2 <sup>A</sup>	
K-means <sup>MFCC</sup>	100	N/A	89.8	36.3	15.1	66.2	39.6	67	49.4
K-means <sup>MFCC</sup>	500		90.3	38	17	70	40.6	67.5	51.3
K-means <sup>MFCC</sup>	2000 <sup>Δ1</sup>		90.2	37.6	15.6	70	44.3	67.6	51.4
K-means <sup>Logmel+Chroma</sup>	300 + 200 <sup>▲1</sup>		90.5	37.6	55.1	75.2	40.1	68.2	62.1
K-means <sup>MFCC</sup>	2000 <sup>Δ2</sup>		90.4	37.5	16.1	68.3	43.9	67.7	51.0
K-means <sup>Logmel+Chroma</sup>	500 <sup>▲2</sup>		90.4	37.7	49.2	72.8	46.5	66.9	60.7
K-means <sup>MFCC+CQT</sup>	300+200		89.4	35.3	53.2	69.0	45.8	66.8	60.2
K-means <sup>Logmel+Chroma</sup>	300 + 200		CQT	90.6	38.4	65.0	<b>78.6</b>	53.1	68.7
RVQ-VAE	1024 × 8 <sup>all codebook</sup>	N/A	<b>90.7</b>	<b>38.7</b>	60.5	72.8	<b>55.3</b>	69.0	65.0
	1024 × 8 <sup>all codebook</sup>		90.5	38.4	63.2	77.2	53.2	<b>72.3</b>	66.9
	1024 <sup>codebook7</sup>	CQT	88.6	34.4	63.5	62.1	33.3	53.2	57.6
	1024 <sup>codebook0</sup>		90	36.7	59.4	67.2	39.7	64.5	60.5
	1024 × 8 <sup>random codebook</sup>		90.6	38.1	<b>66.8</b>	73.8	48.1	68.6	65.8

# More SSL models

- Speech SSL models
  - Wav2Vec: <https://arxiv.org/abs/1904.05862>
  - Wav2Vec 2.0: <https://arxiv.org/abs/2006.11477>
  - COLA: <https://arxiv.org/abs/2010.10915>
  - BYOL-A: <https://arxiv.org/abs/2103.06695>
  
- Music SSL models
  - MULE: <https://arxiv.org/abs/2210.03799>
  - MusicFM: <https://arxiv.org/abs/2311.03318>

# Labels in Classification Models

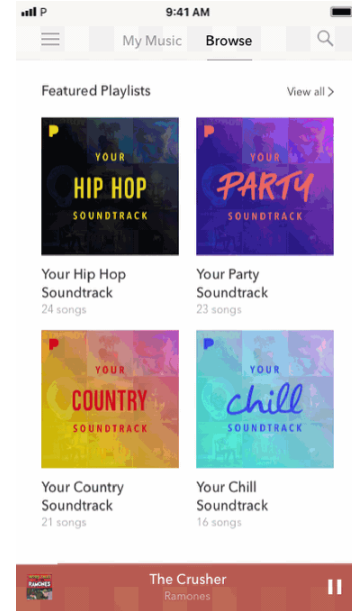
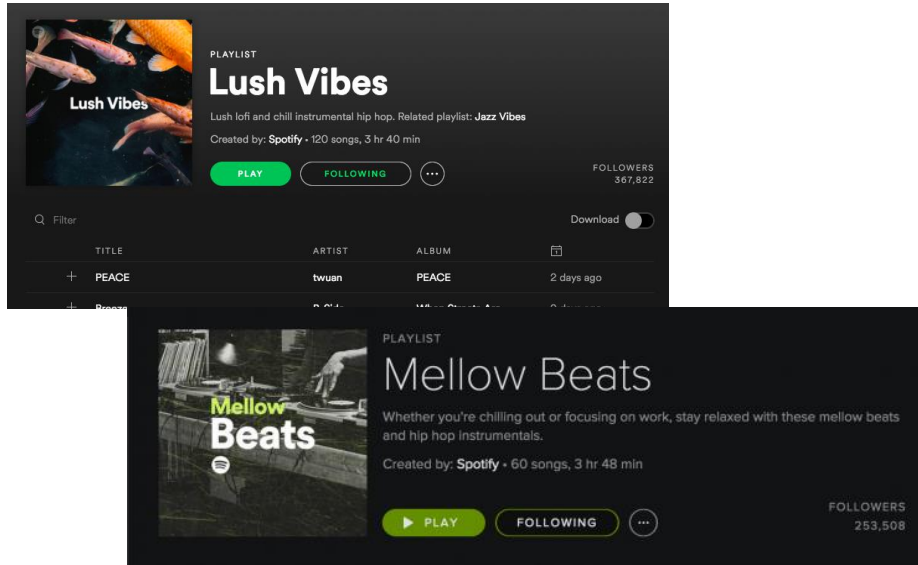
- A fixed number of tags (e.g. top 50 tags) are used in practice



## Top50 tags in MagnaTagATune

guitar, classical, slow, techno, strings, drums, electronic, rock, fast, piano, ambient, beat, violin, vocal, synth, female, indian, opera, male, singing, vocals, no vocals, harpsichord, loud, quiet, flute, woman, male vocal, no vocal, pop, soft, sitar, solo, man, classic, choir, voice, new age, dance, male voice, female vocal, beats, harp, cello, no voice, weird, country, metal, female voice, choral

# Words in Music Services



[Image sources]

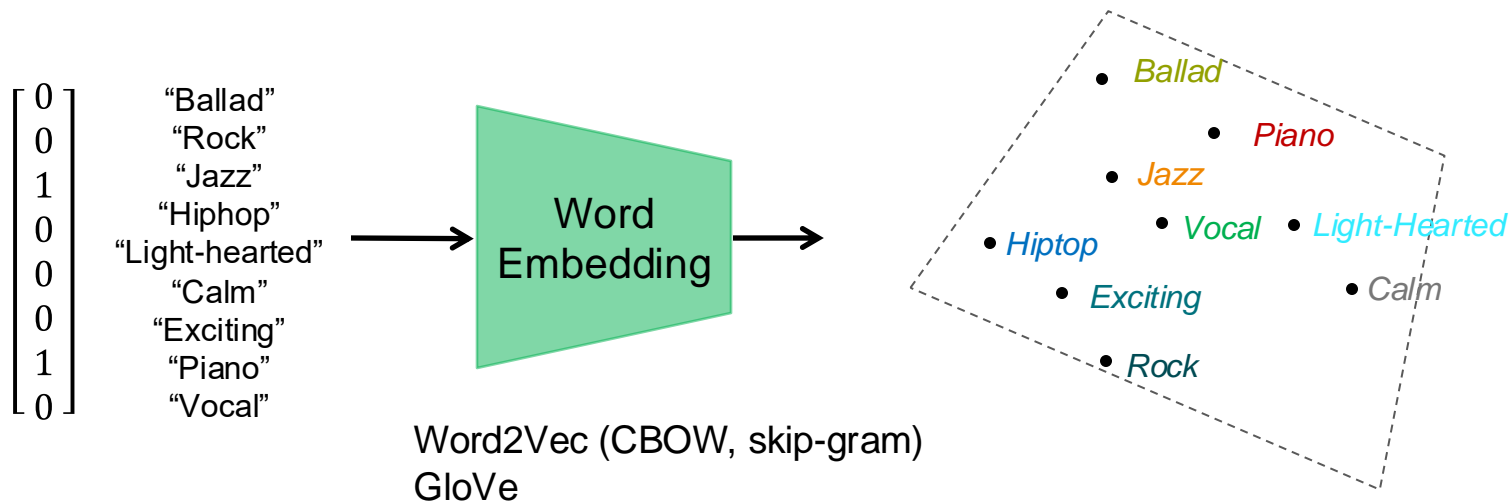
- <https://medium.com/@clintchoi/on-lofi-hip-hop-how-spotifys-mood-based-music-curation-pushes-this-subgenre-2b06c364affc>
- <https://rockcontent.com/blog/5-things-spotify-can-teach-content-curation/>
- <https://techcrunch.com/2018/03/28/pandora-takes-on-spotify-with-dozens-of-personalized-playlists-built-using-its-music-genome/>

# Questions

- [Annotation] can we train the model to **predict a wide variety of words beyond the fixed tag labels?**
- [Retrieval] can we train the model to **search music with arbitrary words?**

# Word Embedding

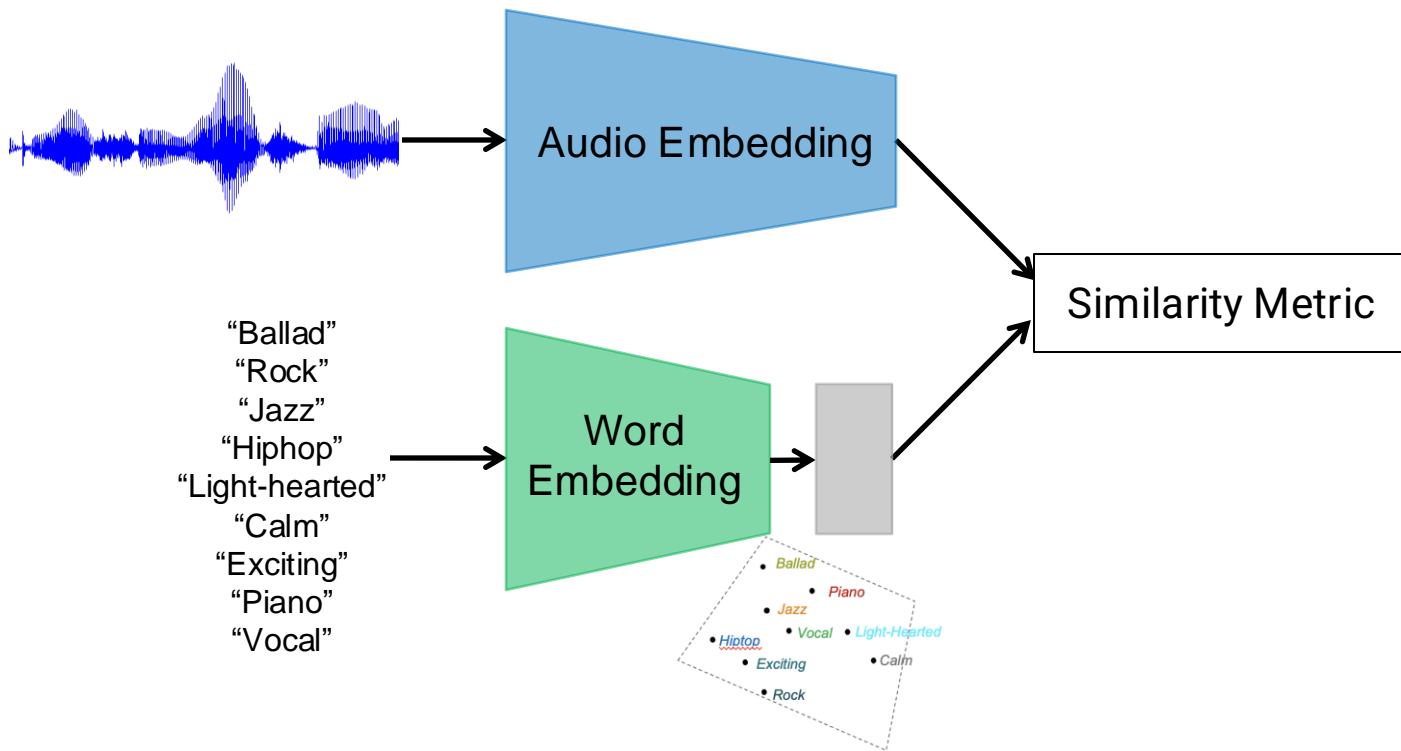
- Mapping tag labels (one-hot style vectors) to a distributed vector space



**Pretrained word embedding models are available**

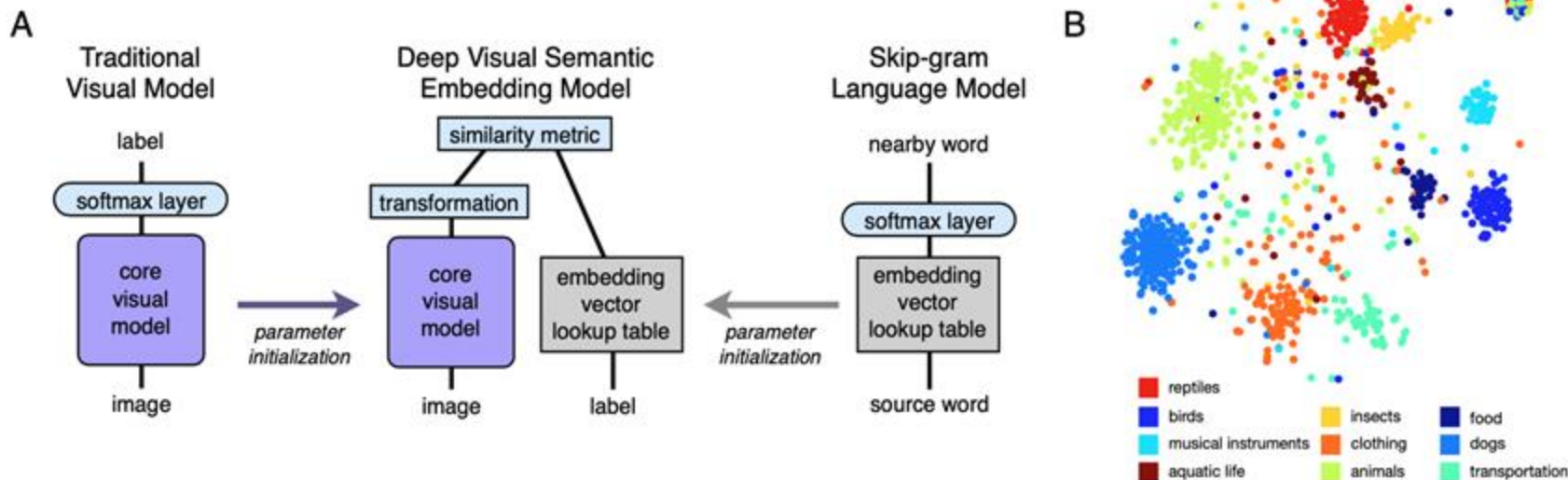
# Cross-modal Metric Learning

- Learning the **joint embedding space** between audio and word



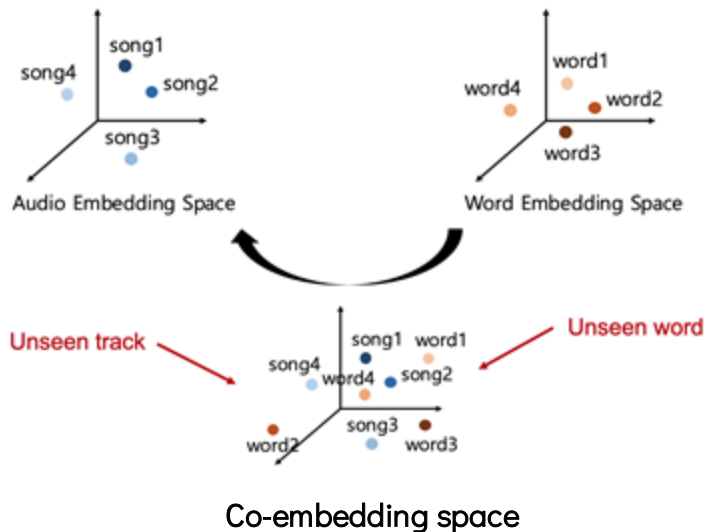
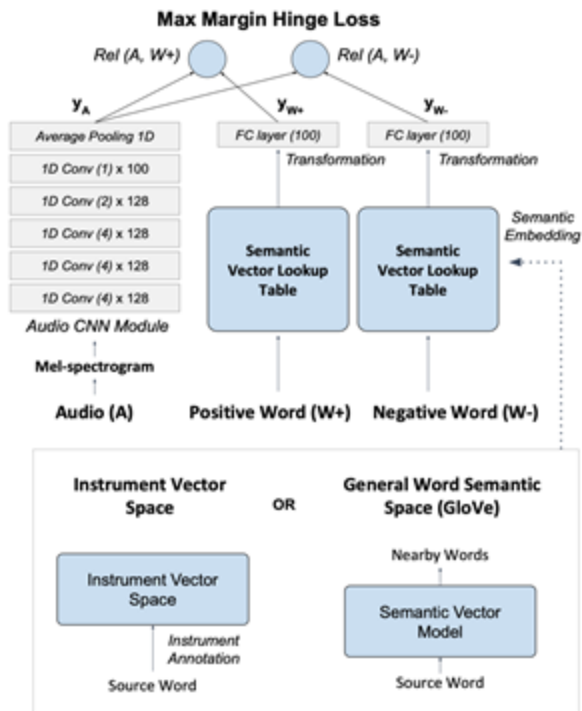
# DeViSE: a Deep Visual-Semantic Embedding Model

- Pretrain each subnetwork individually before cross-modal metric learning



# Cross-modal Metric Learning

- Learn co-embedding between **music** and **tags** using a triplet loss



# Zero-Shot Learning

- Annotation and retrieval results

Smells like teen spirit Nirvana	Superstition Stevie Wonder	Theme to Grace / Lament George Winston
classicrock (unseen)	funk (unseen)	folk
punk	soul	instrumental
rock (unseen)	pop (unseen)	jazz
80s (unseen)	jazz	piano (unseen)
alternative	80s (unseen)	singersongwriter
punkrock	blues (unseen)	chillout
90s	classicrock	acoustic
metal	90s (unseen)	blues
vintage	disco	mellow
alternativerock	dance	chill

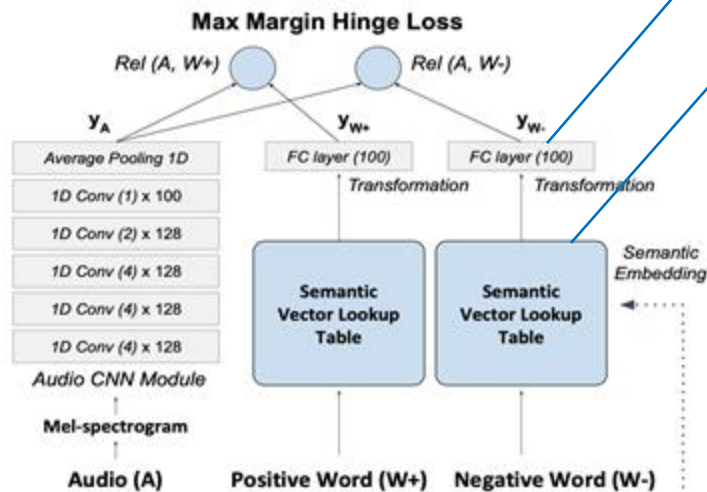
Top 10 auto-tagging results for examples of well-known songs including unseen tags during training

Query	Top 5 Retrieved Tracks (Title / Artist)	Original Last.fm Annotation
guitar	Iron Acton / Beak	psychedelic, experimental, krautrock, english, bass
	Drink Whiskey And Shut up / Brian Setzer	rock
	Thar She Blows / The Halibuts	party, surf, surfrock
	All Quiet On 34th Street / Eric Burdon And The New Animals	rock, rocknroll, hardrock, screamo, 00s
	Gimme Some Lovin' / Traffic	60s, british, classicrock, rock
lovely	Eddie My Love / The Chordettes	50s, doowop, oldies, pop, vocal
	Vaya Con Dios / Les Paul & Mary Ford	50s, jazz, oldies
	(I Can't Help You) I'm Falling Too / Skeeter Davis	country, oldies
	Mr. Blue / The Fleetwoods	oldies, 50s, pop, doowop, ballad
	I'm Blue Again / Patsy Cline	blue, country

Top 5 retrieved tracks for a query word from unseen tag subset ('guitar') and an arbitrary word ('lovely')

# Zero-Shot Learning

## “Acoustically-Informed” Word Embedding

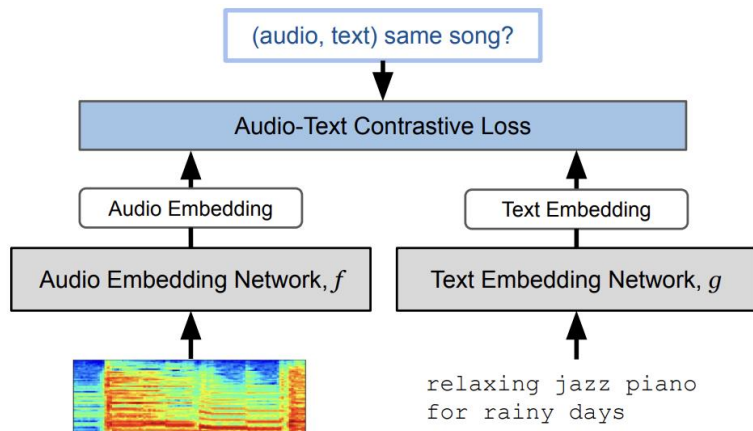


Word Embedding (Glove)

Query	General Semantic Space	Zero-shot Embedding Space
guitar	bass, acoustic, piano, vocals, violin, percussion, strings, vocal, music, jazz	instrumental, minimal, rock, acidrock, progressiverock, alternative, psychedelic, folkrock, classicrock, band
lovely	awesome, love, cool, romantic, relaxing, summer, christmas, holiday, vintage, soft	relaxing, relax, lovesongs, easylisting, baby, country, romantic, easy, americana, ballad

Top 10 nearest word vectors (out of 1126 tags)

- A large-scale audio-text joint embedding (Similar to CLIP)
  - 44 million music recordings from YouTube
  - Audio encoder: audio spectrogram transformer / Resnet50
  - Text encoder: finetune a pre-trained BERT
  - Use CLS token for both audio and text embeddings



**Table 1.** Text annotation examples.

Type	Examples
Short-form (SF)	tags like genre, mood, instrument, artist name, song title, album name
Long-form (LF)	'Hip-hop features rap with an electronic backing.' 'The melody is so nostalgic and unforgettable.'
Playlist (PL)	'Feel-good mandopop indie', 'Latin workout' 'Salsa for broken hearts', 'Piano for study'

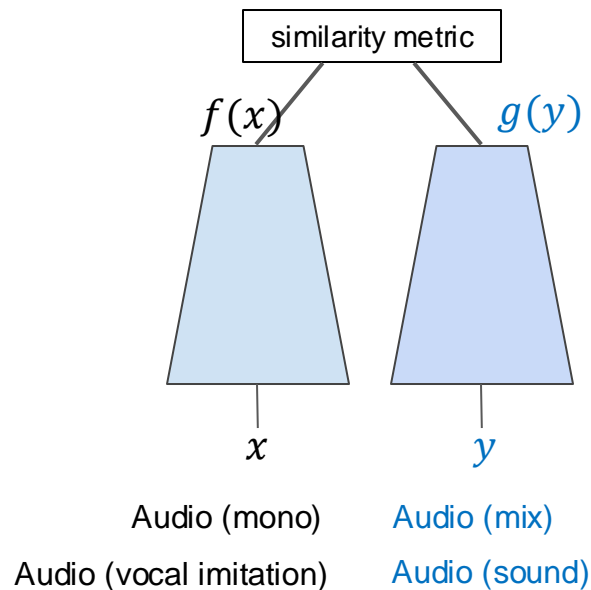
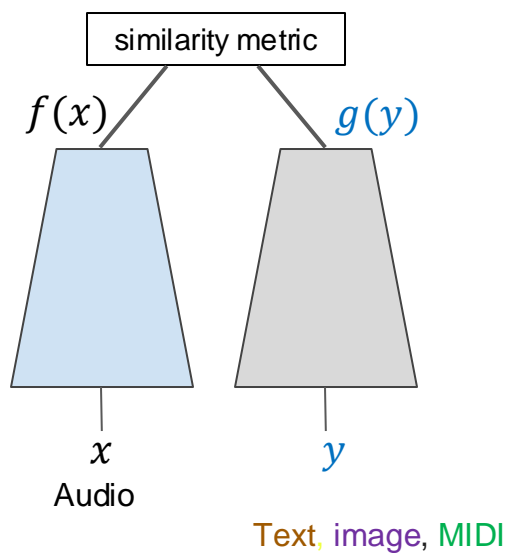
Free form text strings are filtered using a music text classifier

# More Text-to-Music Joint Embedding

- MuLaP: <https://arxiv.org/abs/2112.04214>
- TTMR: <https://arxiv.org/abs/2211.14558>

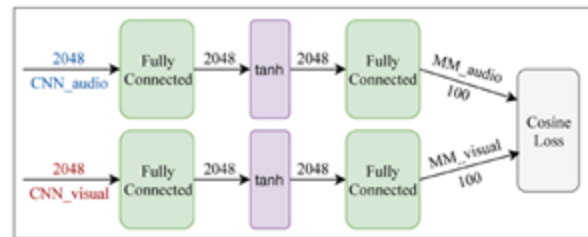
# More Cross-modal Representation Learning

- Learning correspondence between two different modalities of inputs
  - Each modality has its own subnetwork



# Audio and Album Cover Image

- Learning joint embedding between **audio** and **image**
  - Audio: constant-Q transform (30s)
  - Image: album cover
- Audio and vision subnetworks are individually trained to classify genres
  - The vision subnetwork is initialized using a pretrained network with the ImageNet dataset
- The pretrained subnetworks are co-embedded using metric learning (triplet loss)
  - Visually or aurally-informed features are extracted for multi-label genre classification



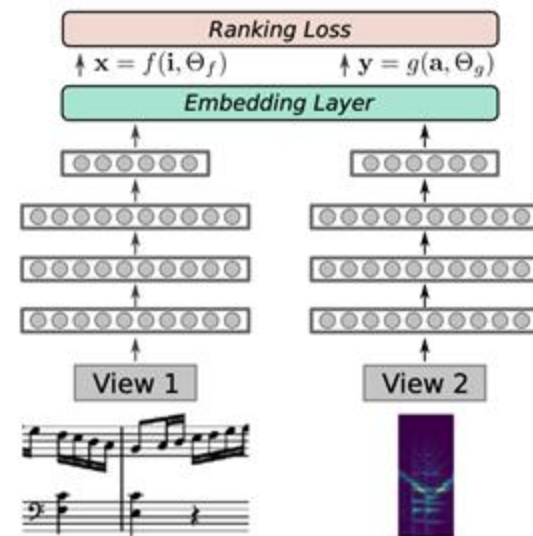
# Audio and Album Cover Image

- Results

Input	Model	P	R	F1	
Audio	CNN_AUDIO	$0.385 \pm 0.006$	$0.341 \pm 0.001$	$0.336 \pm 0.002$	
	MM_AUDIO	$0.406 \pm 0.001$	$0.342 \pm 0.003$	$0.334 \pm 0.003$	
	<b>CNN_AUDIO + MM_AUDIO</b>	$0.389 \pm 0.005$	$0.350 \pm 0.002$	<b><math>0.346 \pm 0.002</math></b>	← Visually-informed audio feature improves accuracy
Video	<b>CNN_VISUAL</b>	$0.291 \pm 0.016$	$0.260 \pm 0.006$	<b><math>0.255 \pm 0.003</math></b>	
	MM_VISUAL	$0.264 \pm 0.005$	$0.241 \pm 0.002$	$0.239 \pm 0.002$	
	CNN_VISUAL + MM_VISUAL	$0.271 \pm 0.001$	$0.248 \pm 0.003$	$0.245 \pm 0.003$	← Aurally-informed image feature does not help
A + V	CNN_AUDIO + CNN_VISUAL	$0.485 \pm 0.005$	$0.413 \pm 0.005$	$0.425 \pm 0.005$	
	MM_AUDIO + MM_VISUAL	$0.467 \pm 0.007$	$0.393 \pm 0.003$	$0.400 \pm 0.004$	
	<b>ALL</b>	$0.477 \pm 0.010$	$0.413 \pm 0.002$	<b><math>0.427 \pm 0.000</math></b>	← Combining all together improve accuracy further

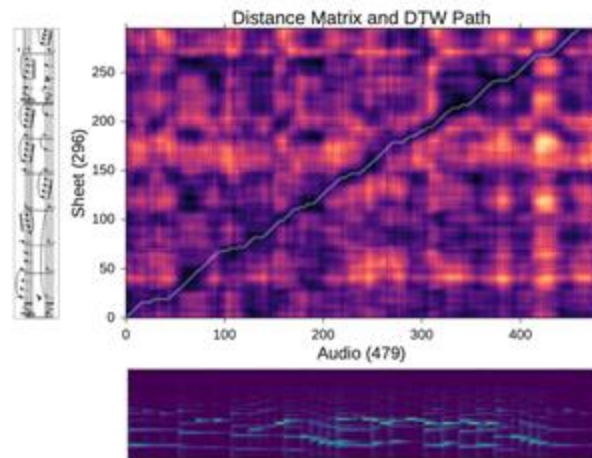
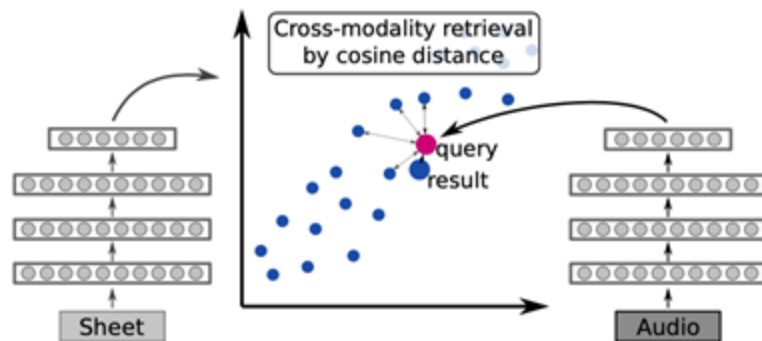
# Audio and Sheet Music Image

- Learning joint embedding between audio and sheet music image
  - Audio excerpt: 92bins x 42 frames (log-freq. spectrogram)
  - Sheet music snippet: 180 x 200 pixels
- Data augmentation
  - Audio: synthesized with different samples and tempo change
  - Sheet music: scaling, translation
- Use the triplet hinge loss  $\mathcal{L}_{rank} = \sum_{\mathbf{x}} \sum_k \max\{0, \alpha - s(\mathbf{x}, \mathbf{y}) + s(\mathbf{x}, \mathbf{y}_k)\}$ 
  - Cosine similarity
  - Anchor: sheet music embedding
  - Positive: matching audio
  - Negative: non-matching audio



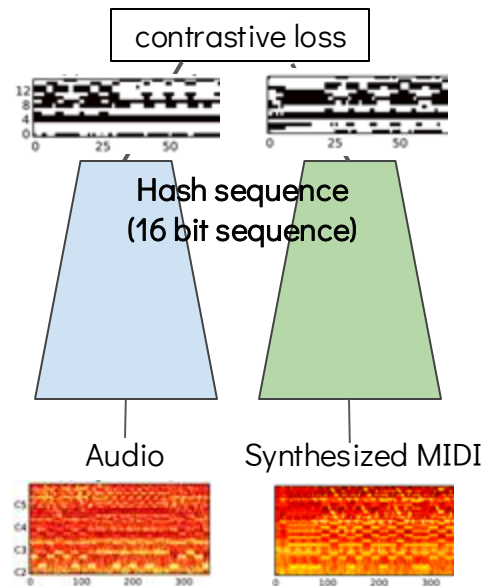
# Audio and Sheet Music Image

- Task 1: sheet music identification from audio queries: voting of retrieved results from all audio excerpts within a whole audio recording
- Task 2: offline alignment of a given audio with the sheet music image using the distance matrix where each point is computed from the two embeddings



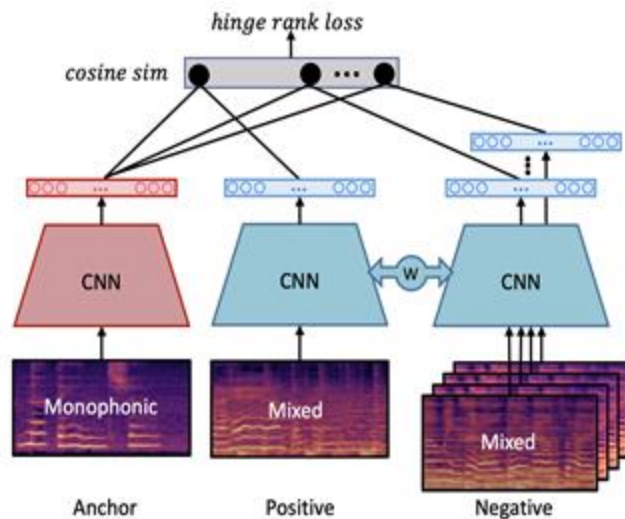
# Audio and MIDI

- Large-scale content-based MIDI-to-Audio retrieval
  - Find matching between 140,910 MIDI files and 994,950 audio files
  - The traditional method (DTW over CQT/chroma) is too slow!
  - Computing the distance matrix is the bottleneck
- Speed-up by learning “binary vectors” from audio and synthesized MIDI
  - The exclusive-or operation (hamming distance) between the hash sequences is 400 times faster than the inner product between the CQT/Chroma
  - The DTW score with the hamming distance is 9 times faster



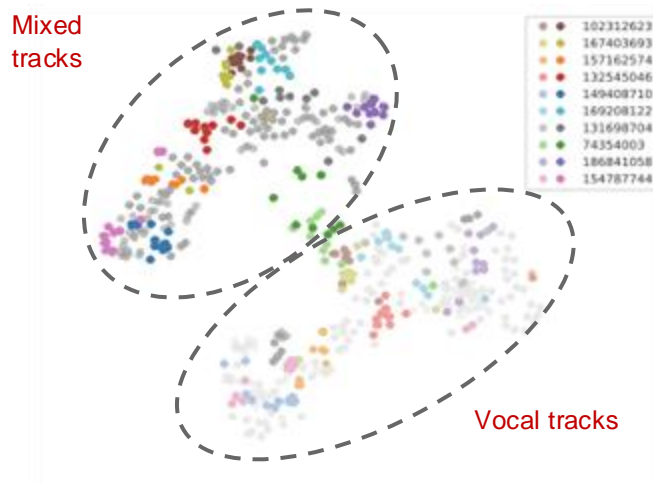
# Mono Audio and Mixed Audio

- Learn a joint embedding space between monophonic vocal and its mix with background music
- Data generation
  - Vocal track: karaoke singing (DAMP)
  - Mixed track: auto-mashup between the vocal and background tracks (musdb18)

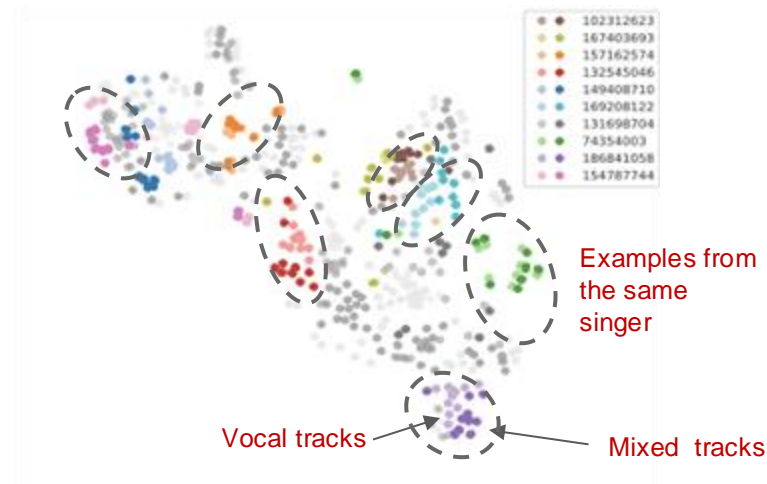


# Mono Audio and Mixed Audio

- Vocal tracks and their mixed tracks are projected closely: “implicit vocal source separation”



Before cross-modality metric learning  
(the model is trained with mixed tracks only)



After cross-modality metric learning  
(the model is jointly trained with mono and mixed tracks)