Few-Shot Learning and Zero-Shot Learning for MIR

Yu Wang[1], Hugo Flores García[2], Jeong Choi[3]

[1]New York University
[2]Northwestern University
[3]NAVER Corp.

Tutorial Outline

- Part 1: Introduction & Foundations
- Part 2: Coding Examples
- Part 3: Recent Advances
- Closing Remarks

Tutorial Outline

- Part 1: Introduction & Foundations
- Part 2: Coding Examples
- Part 3: Recent Advances
- Closing Remarks

The Issue of Labeled Data Scarcity in MIR

Deep learning models are data hungry

Annotating music is hard

1. Listening takes time
2. Expert knowledge
3. Overlapped events

Some of the existing solutions

Data

1. Crowdsourcing
2. Augmentation
3. Synthesis

- Significant human effort
- Generalizability issue

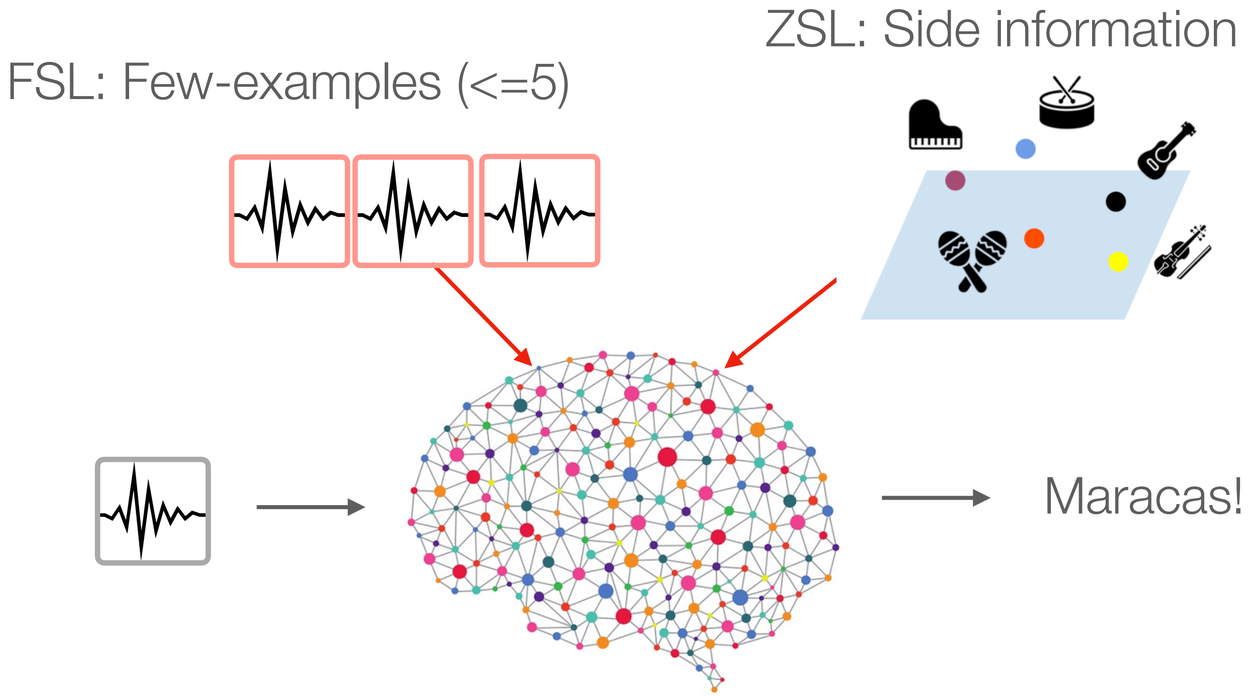
Learning paradigm

1. Transfer learning
2. Representation learning
3. Unsupervised learning
4. Semi-supervised learning

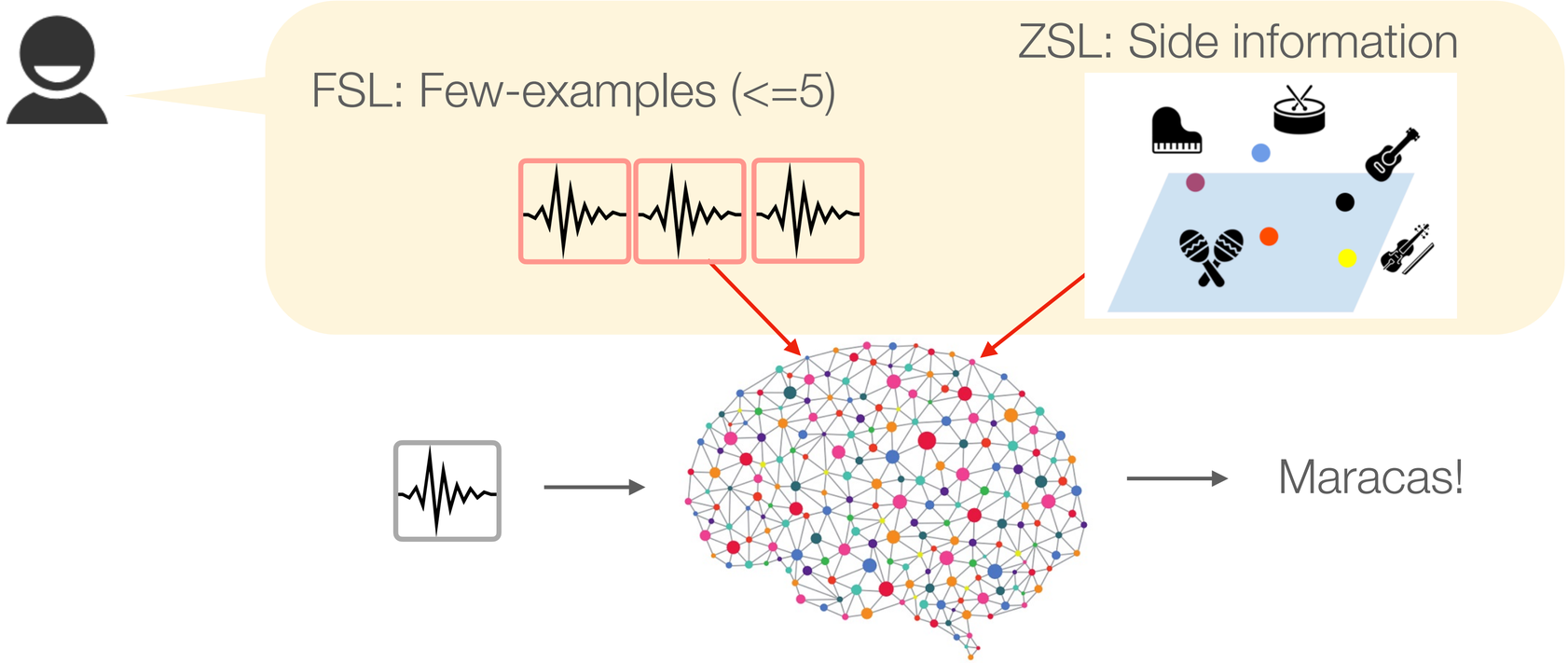
- Often still requires a significant amount of labeled data for the downstream tasks

Few-Shot Learning (FSL) and Zero-Shot Learning (ZSL)

Learning paradigms that aim to learn a model that can learn a new concept (e.g. recognize a new class) quickly in a data efficient way



Advantages of FSL and ZSL



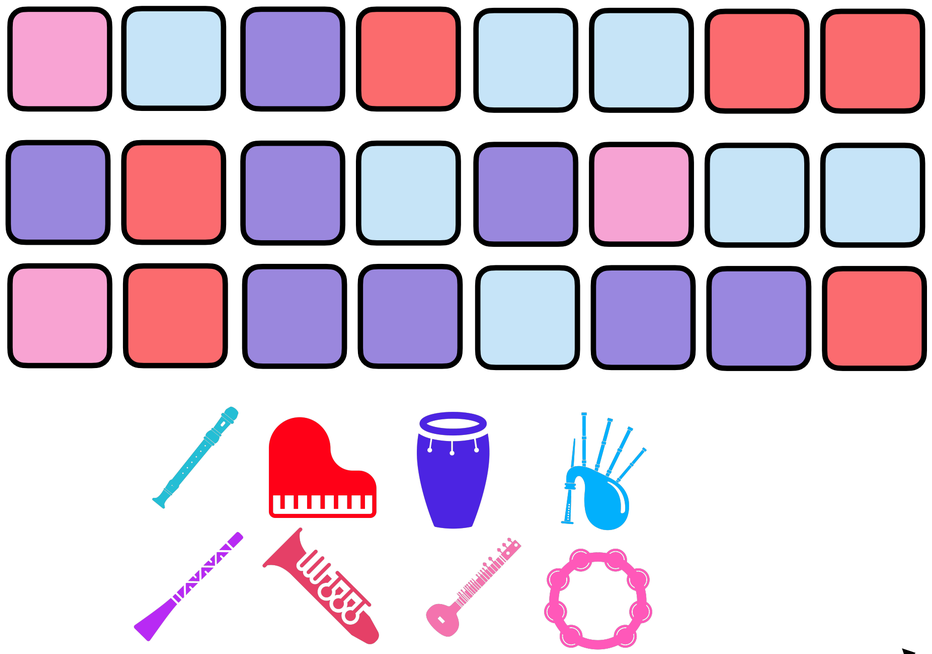
1. Learning rare, fine-grained, new classes
2. Incorporating human input: a model that can be customized in an interactive way

Foundations: FSL

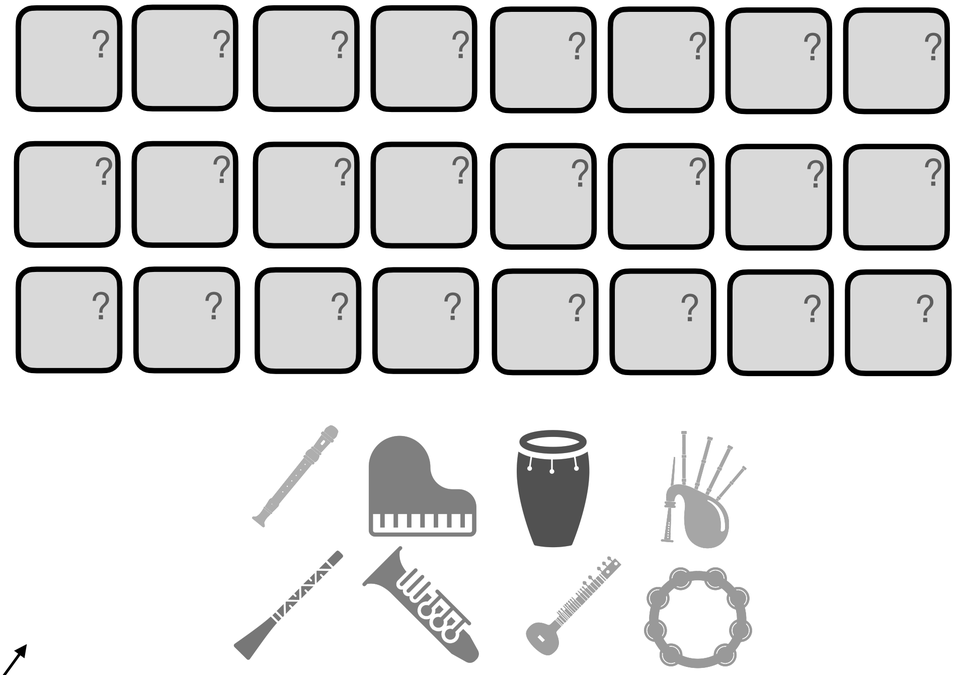
Thinking about Data

(In supervised learning)

Training



Inference



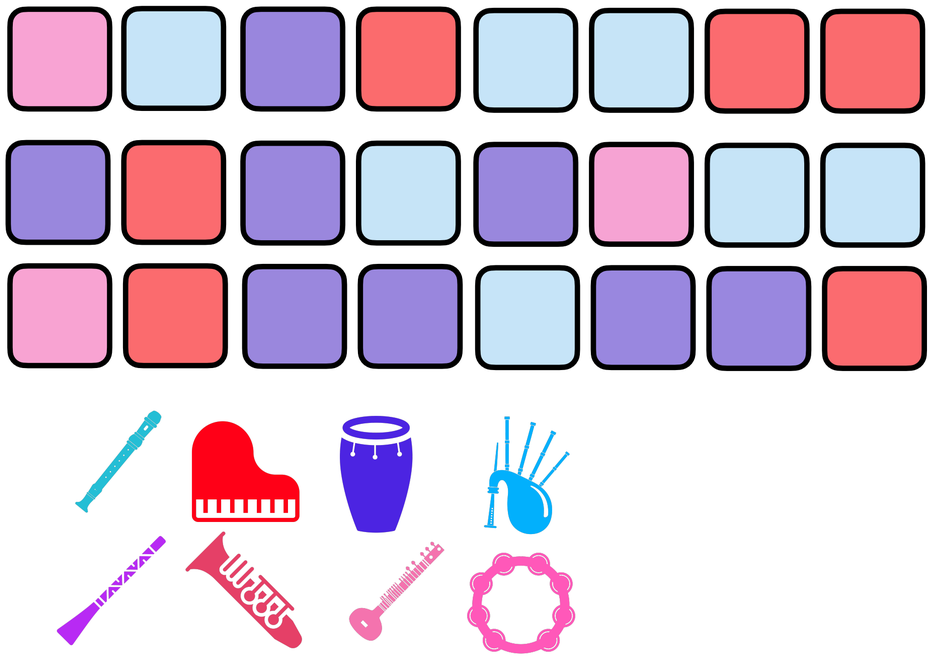
Assumption: same set of classes!

Thinking about Data

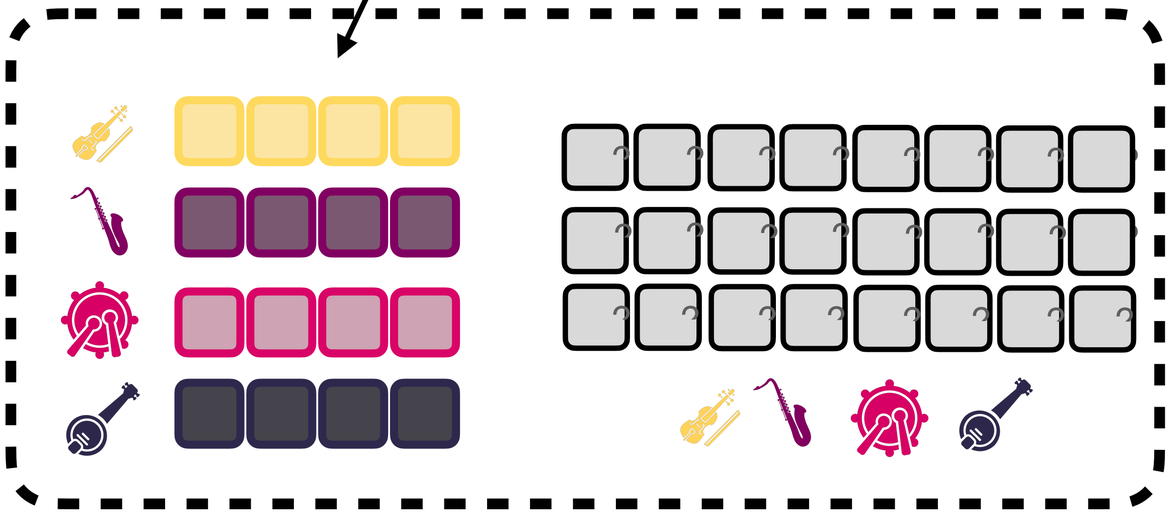
(In few-shot learning)

A few labeled “shots” for each novel class

Training



Inference

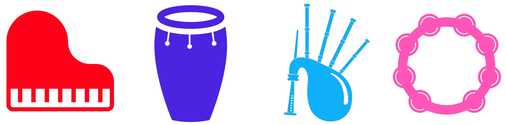


A few-shot learning “task”!

NOT the same set of classes!

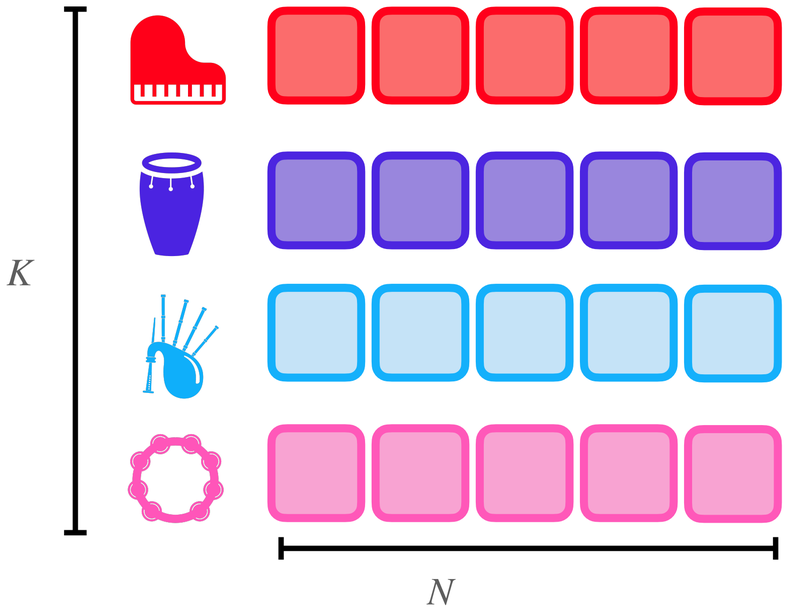
Defining the Problem

We'd like to classify between K classes **unseen** during training.



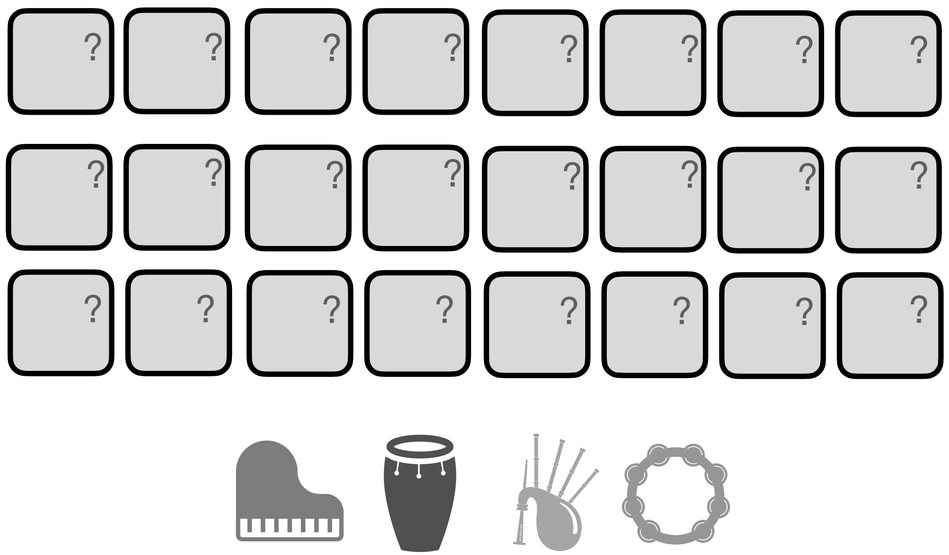
K-way, N-shot task!

for each of these,
we are given N **labeled** examples per class



Support Set

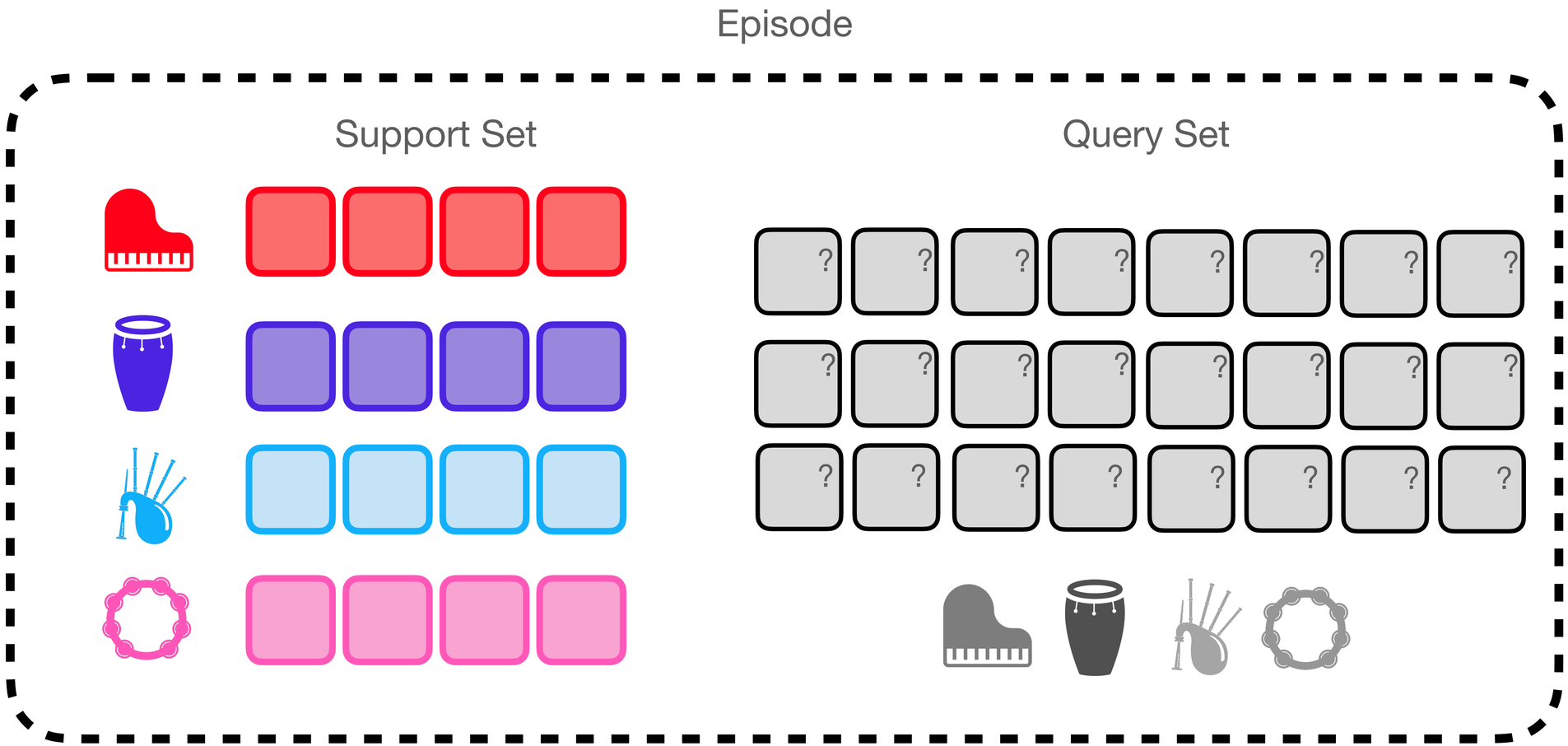
we also have a set of many **unlabeled** examples
that we would like to label



Query Set

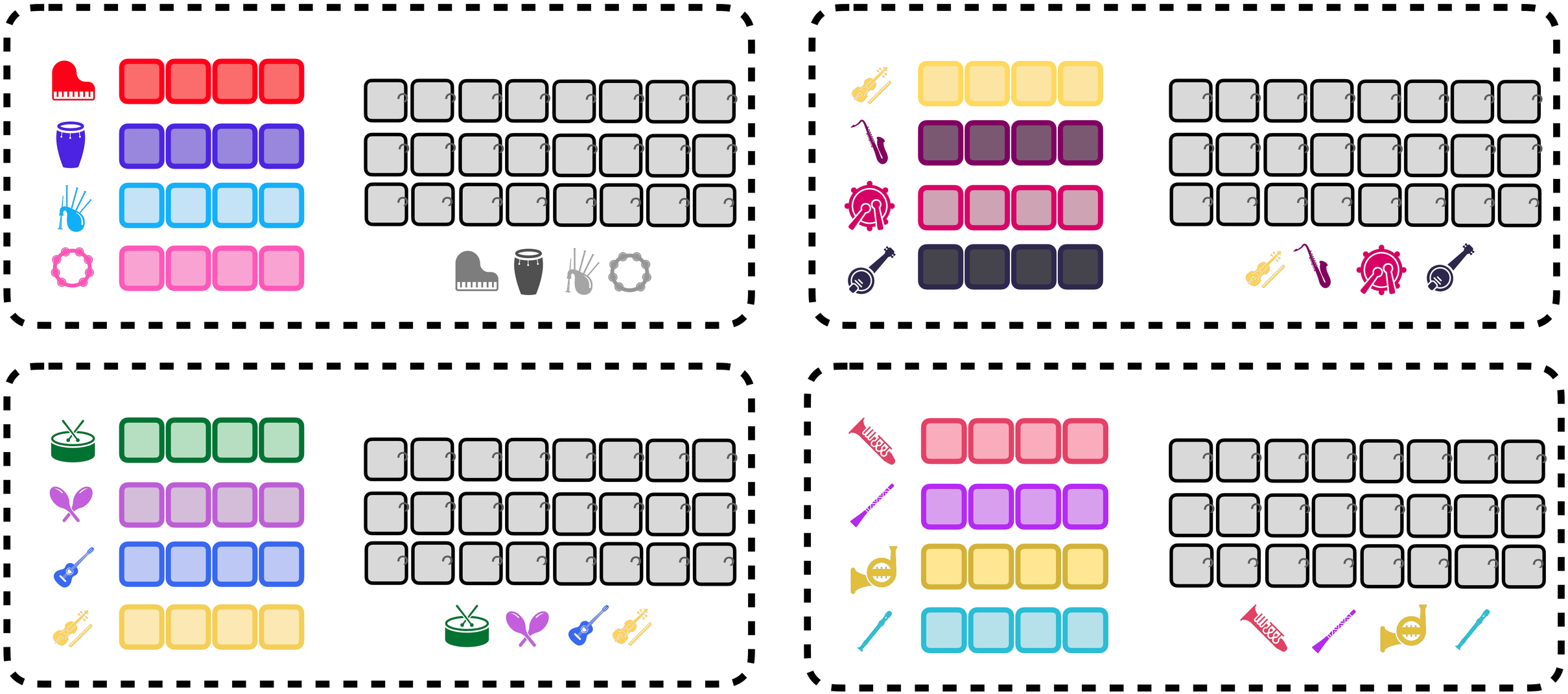
Defining the Problem

We will refer to a few-shot learning instance as an episode.



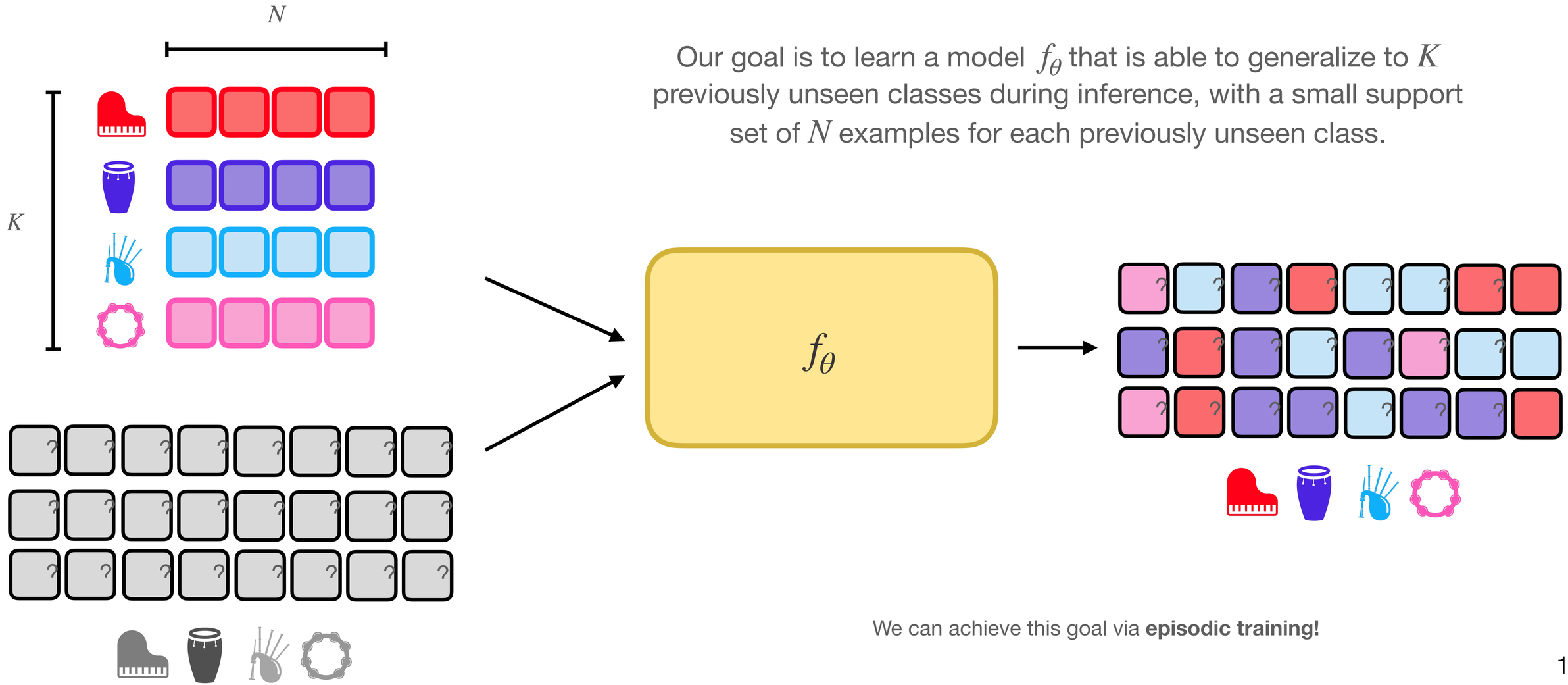
Defining the Problem

In few-shot learning, we want to generalize to many different episodes during inference.



Learning to Learn

Our goal is to learn a model f_θ that is able to generalize to K previously unseen classes during inference, with a small support set of N examples for each previously unseen class.

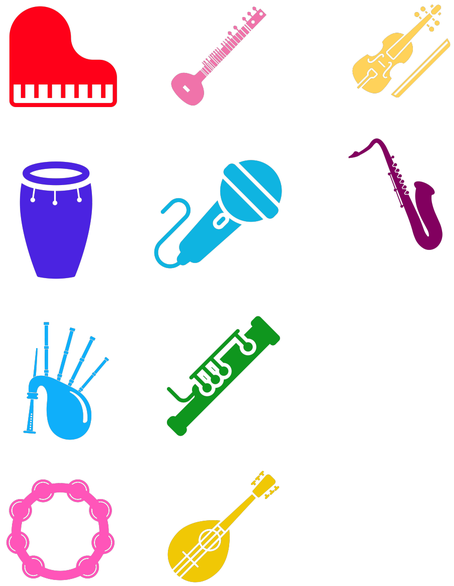


We can achieve this goal via **episodic training!**

Episodic Training

To mimic a K-way, N-shot few-shot learning scenario, we can randomly sample episodes from our training dataset.

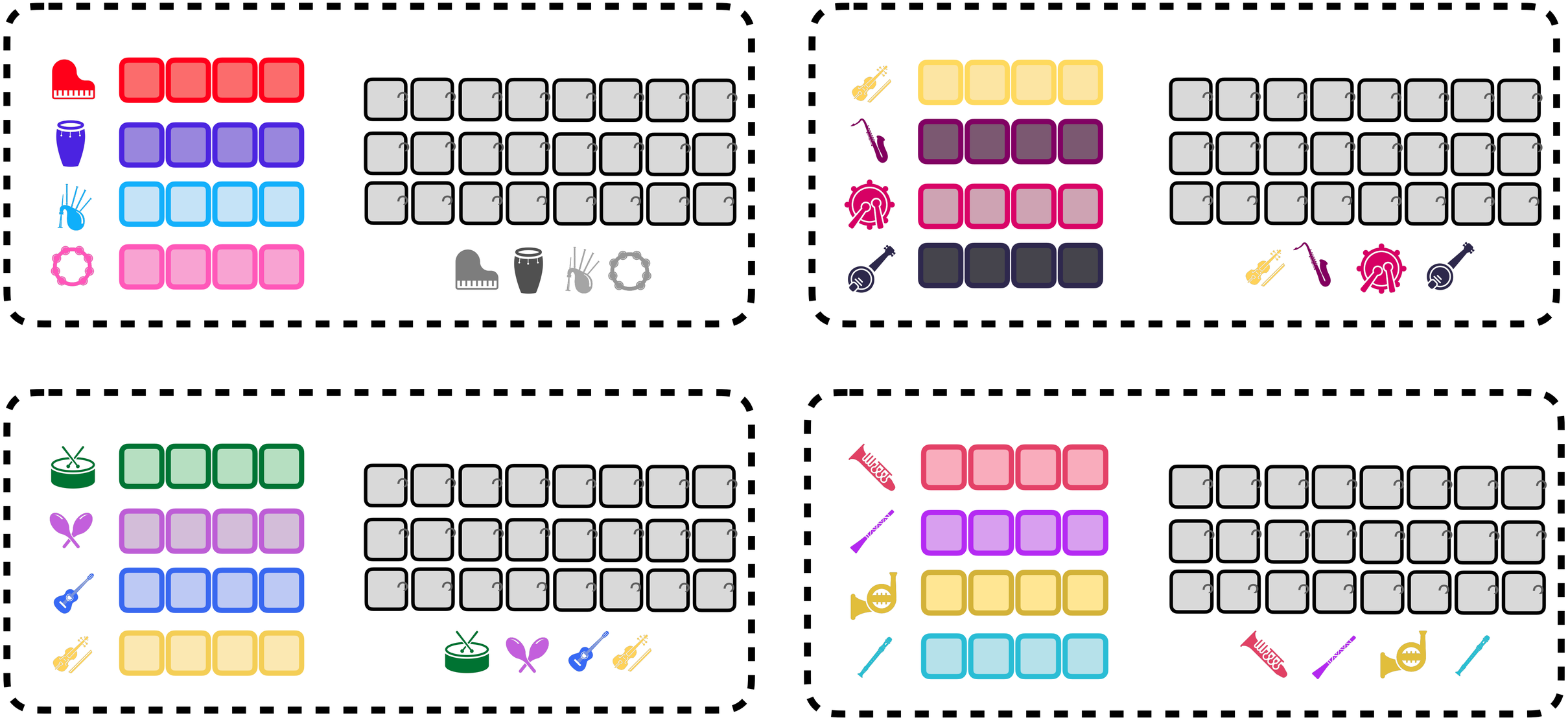
Train Data



Training for $K = 2$:

Episodic Training

By doing episodic training, our model “learns how to learn” new classes.



Episodic Training

Episodic training mimics few-shot learning episodes during each iteration of training.

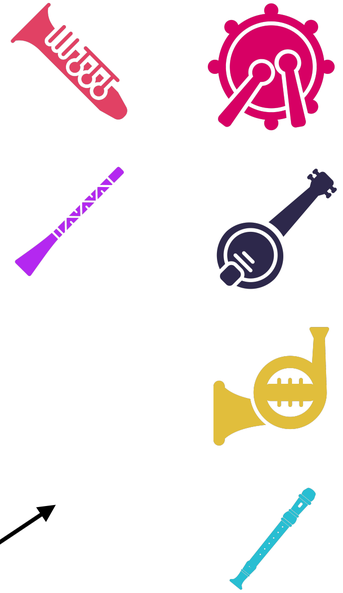
To train a model using episodic training, we must start with a large dataset with a number of classes $\gg K$!

Class-conditional Splits

For episodic training, we must test our model's ability to learn novel classes, which means that there should be no overlap between classes in the train and test set.

Train Data

Test Data

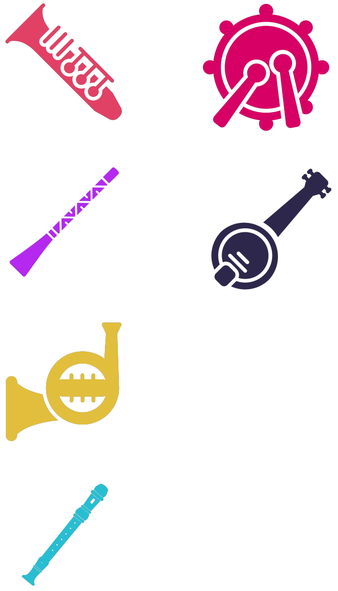


No class overlap!

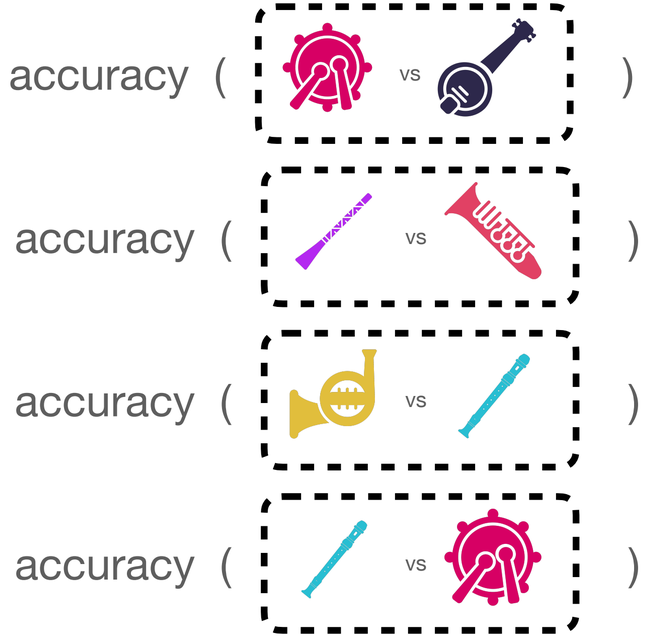
Evaluating a Few-Shot Model

We can evaluate a few-shot model in a fashion similar to training.
Simply build episodes and compute your metric of choice!

Test Data

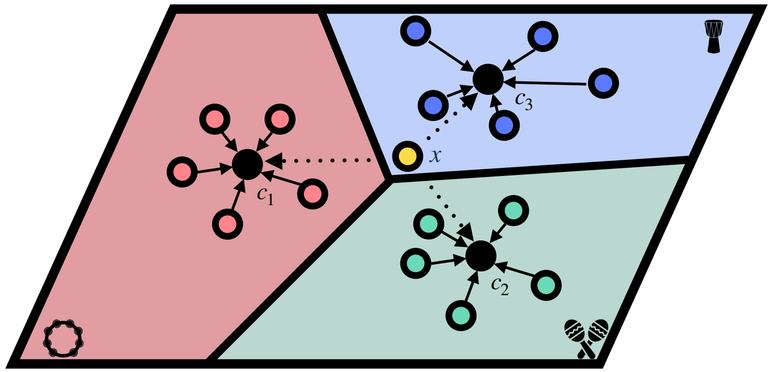


$K = 2$



Learning to Learn - Approaches

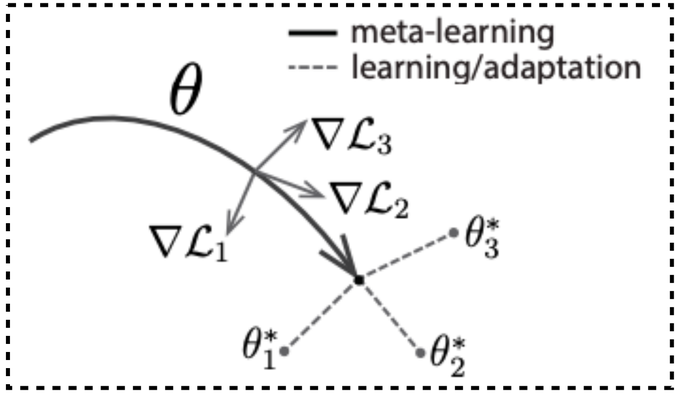
Metric-Based Learning



(Snell et al., 2017)

learning a discriminative embedding space for unseen classes using a similarity metric

Optimization-Based Learning

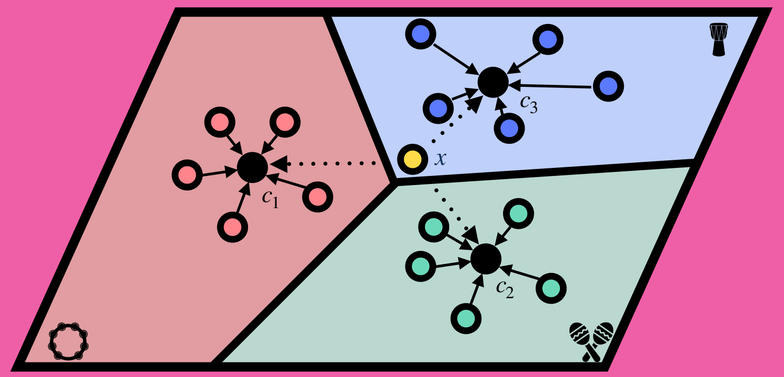


(Finn et al., 2017)

learning model parameters that are easily fine-tuned for unseen classes

Learning to Learn - Approaches

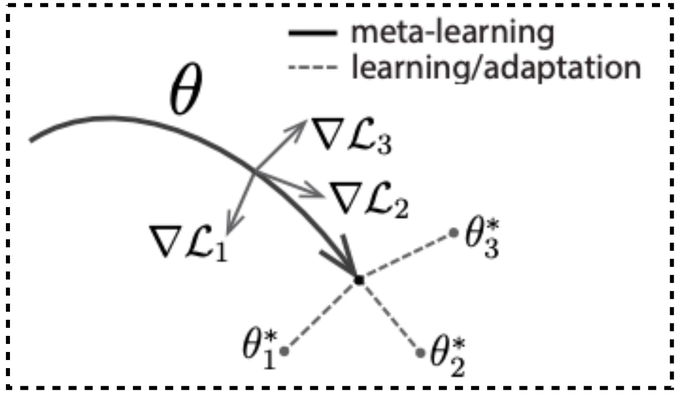
Metric-Based Learning



(Snell et al., 2017)

learning a discriminative embedding space for unseen classes using similarity metrics

Optimization-Based Learning

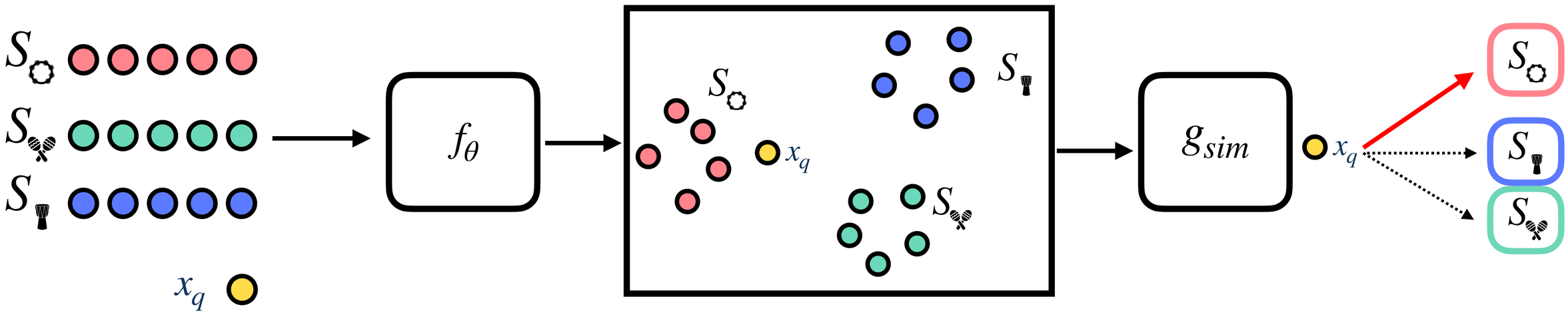


(Finn et al., 2017)

learning model parameters that are easily fine-tuned for unseen classes

Metric-based FSL

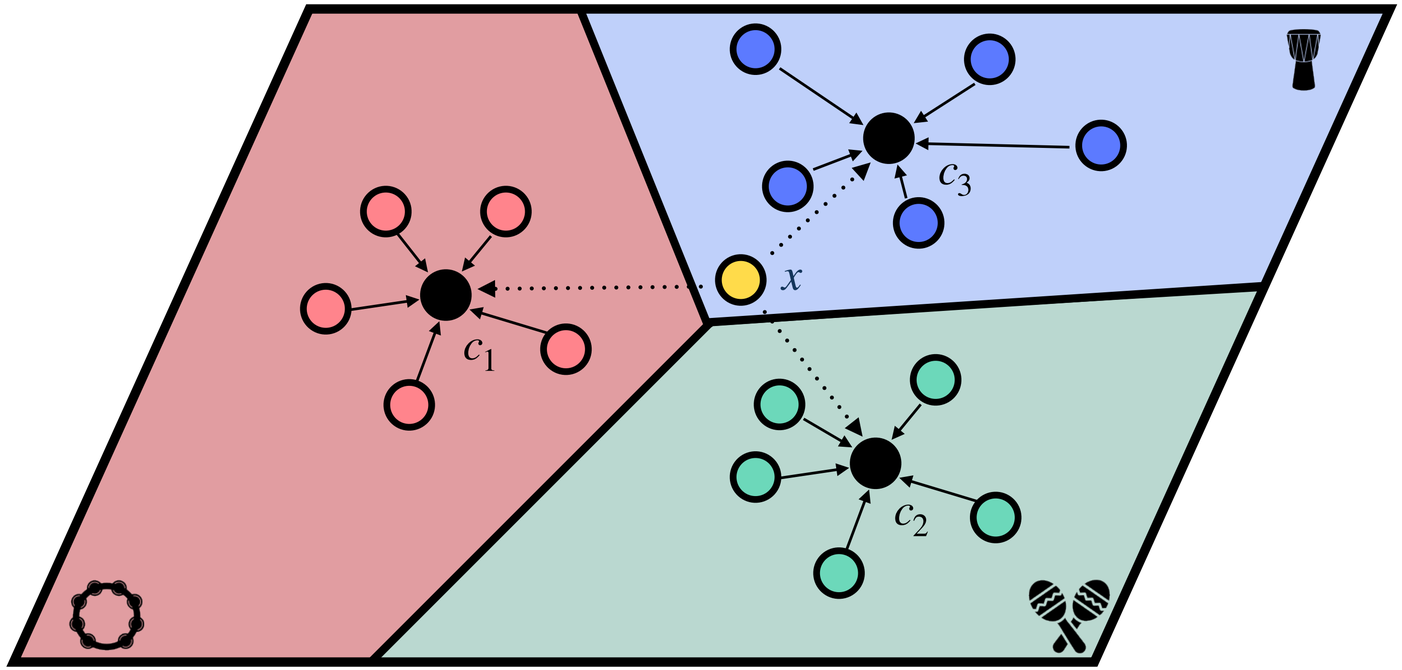
Goal: learn an embedding space where examples that belong to the same class are close together, according to some **metric**.



Metric-based FSL: Prototypical Networks

(Snell et al., 2017)

Prototypical networks compute
one embedding to represent an entire class.
This class representation is called the **prototype.**

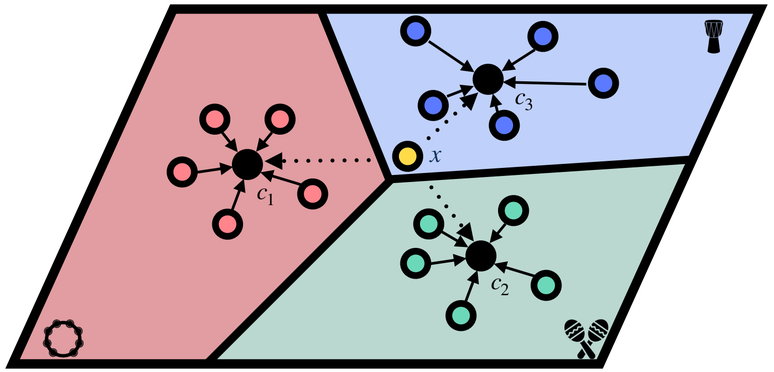


To calculate a class prototype, we must take the **mean** of all embeddings in the support set belonging to a given class.

Given a **query** example, we can make a prediction by finding the **nearest prototype**, using the squared euclidean distance as our metric.

Learning to Learn - Approaches

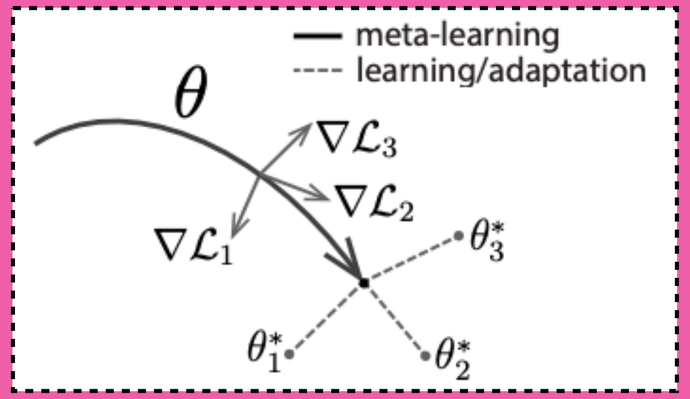
Metric-Based Learning



(Snell et al., 2017)

learning a discriminative embedding space for unseen classes using similarity metrics

Optimization-Based Learning



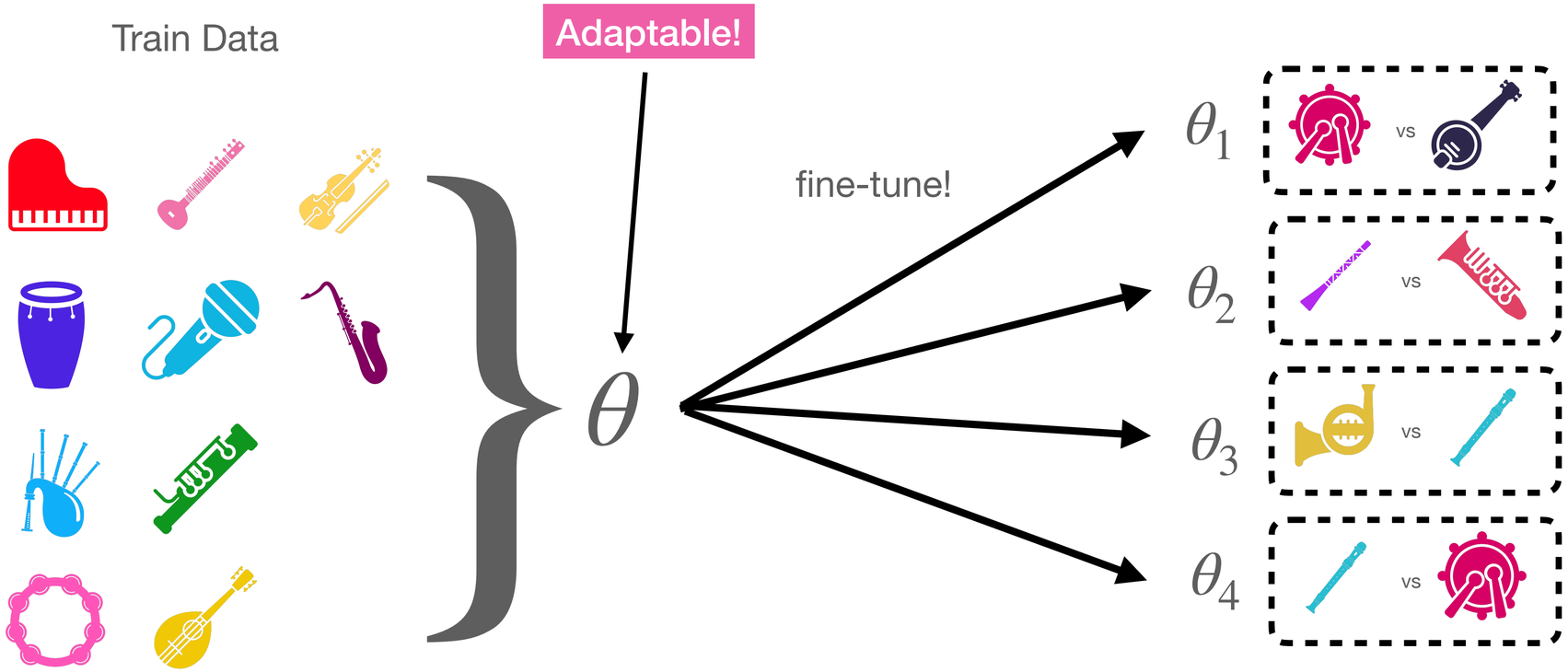
(Finn et al., 2017)

learning model parameters that are easily fine-tuned for unseen classes

Optimization-based FSL

Goal: learn some model parameters θ that can easily adapt to new tasks, and thus new classes, using a few gradient steps.

Some starting parameters θ are more easily adaptable to new tasks than others!



How do we learn θ such that we can easily adapt it to new tasks?

Optimization-based FSL: MAML

The canonical method for optimization-based few-shot learning is
Model-Agnostic Meta Learning (MAML)

(Finn et al., 2017)

Start with some initial parameters θ, a fine-tune step size α and a step-size β

while not converged, do:

1. Sample a batch of episodes

2. For each episode (i) :

   - Use the initial parameters θ to compute the gradient $\nabla_\theta L_i f(\theta)$ of the loss $L_i f(\theta)$ for the current episode.

   - Compute a new set of parameters θ_i by fine-tuning in the direction of the gradient w.r.t. the starting parameters θ:

     $$\theta_i = \theta - \alpha \nabla_\theta L_i$$

3. Using the fine-tuned parameters θ_i, make a prediction and compute the loss $L_i f(\theta_i)$

These losses are NOT the same!

4. Update the starting parameters θ by taking a gradient step in the direction of the loss we computed with the fine-tuned parameters $L_i f(\theta_i)$:

$$\theta = \theta - \beta \nabla_\theta \sum_{E_i \in D_{train}} L_i f(\theta_i)$$

Foundations: ZSL

Zero-Shot Learning (ZSL)

Data : $x \in X \xrightarrow{\text{Train}}$ **Class label :** $y \in Y$

Data : $x' \in X \xrightarrow[\text{Predict}]{}$ **Class label :** $z \in Z$ (where $Y \cap Z = \emptyset$)

No additional shots

Overview on ZSL Framework

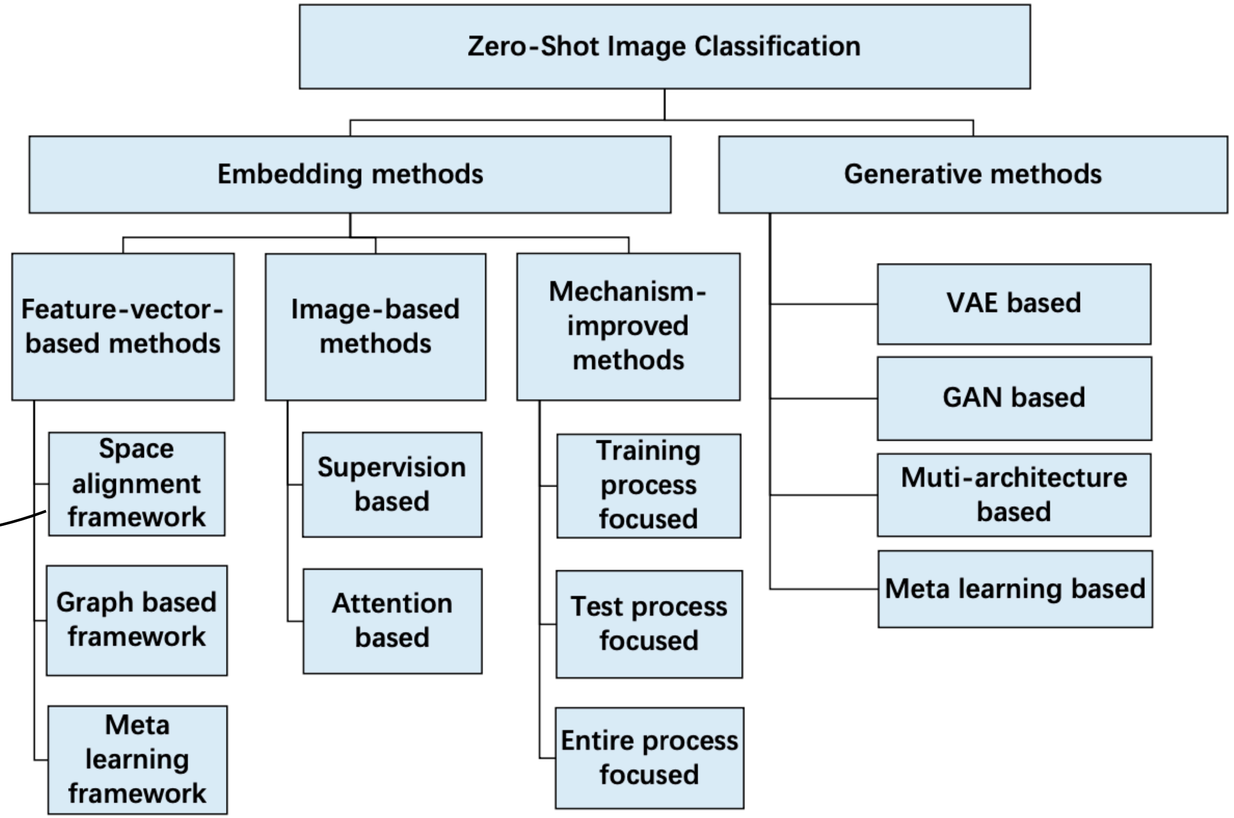
- Basic framework (an example of a space-alignment framework).
- Generative method (an example of a GAN based framework).
- Data split & evaluation scheme
- Side information

Overview on ZSL Framework

- Basic framework (an example of a space-alignment framework).
- Generative method (an example of a GAN based framework).
- Data split & evaluation scheme
- Side information

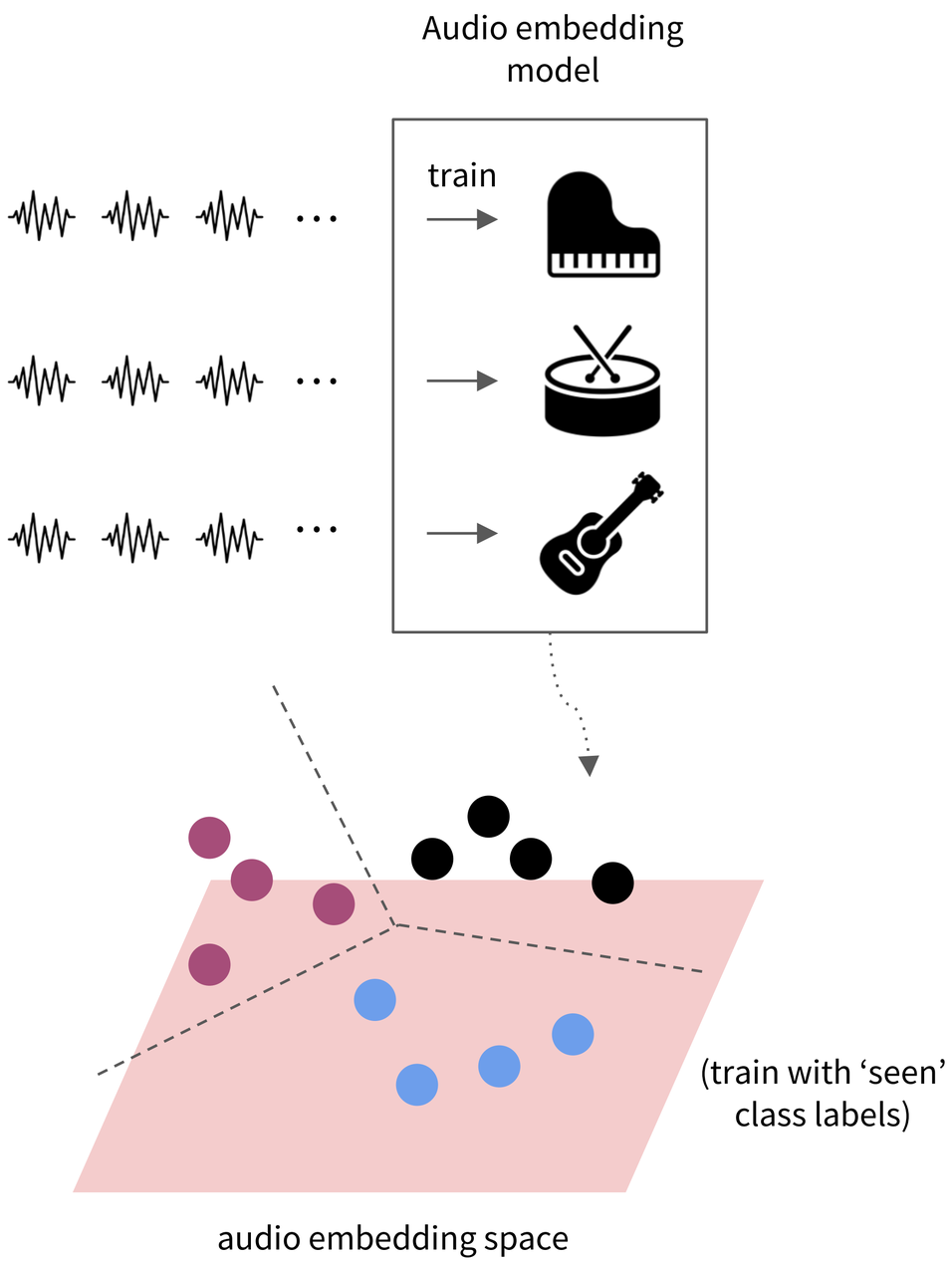
Overview on ZSL Framework

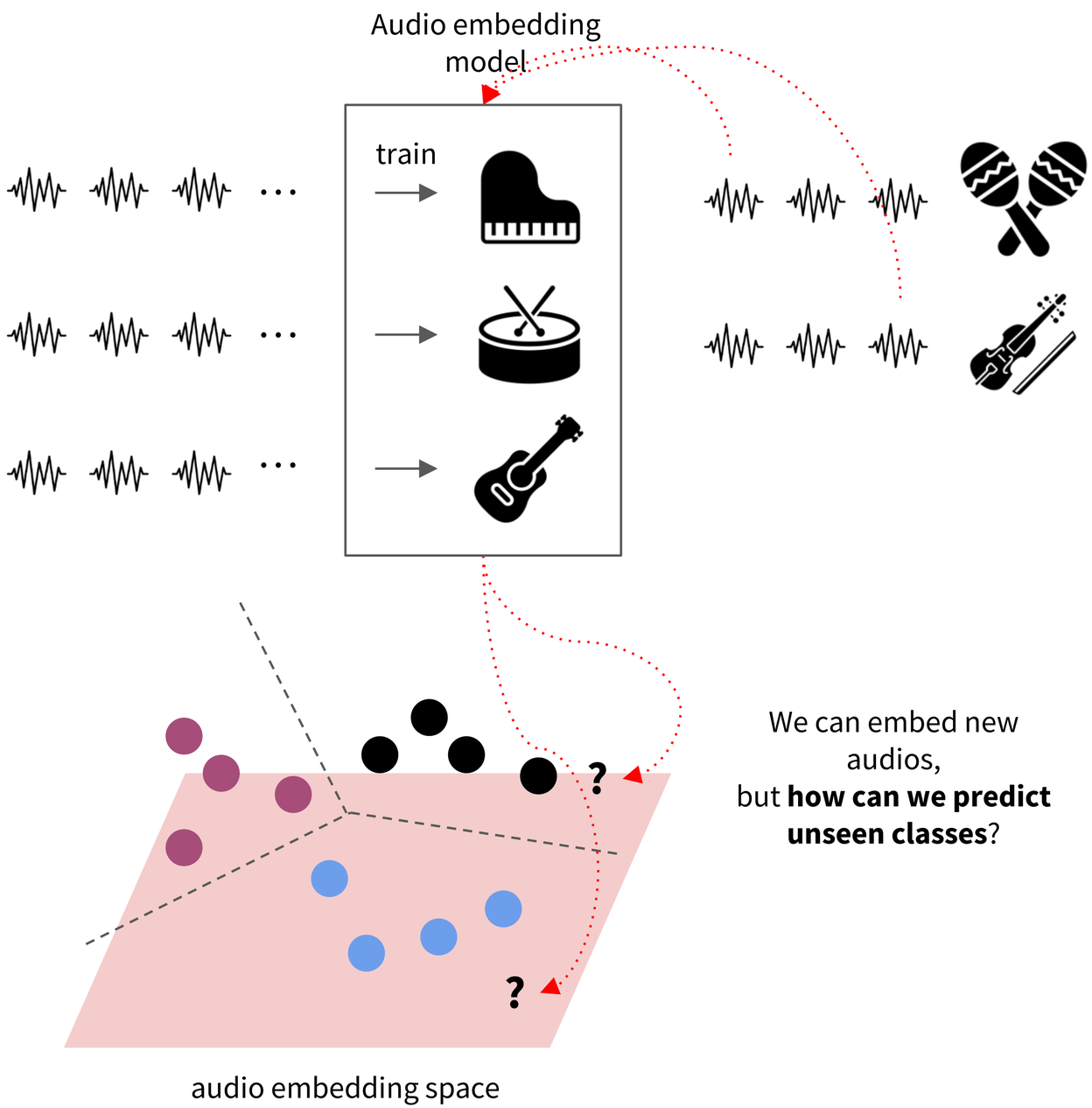
We'll go over this.



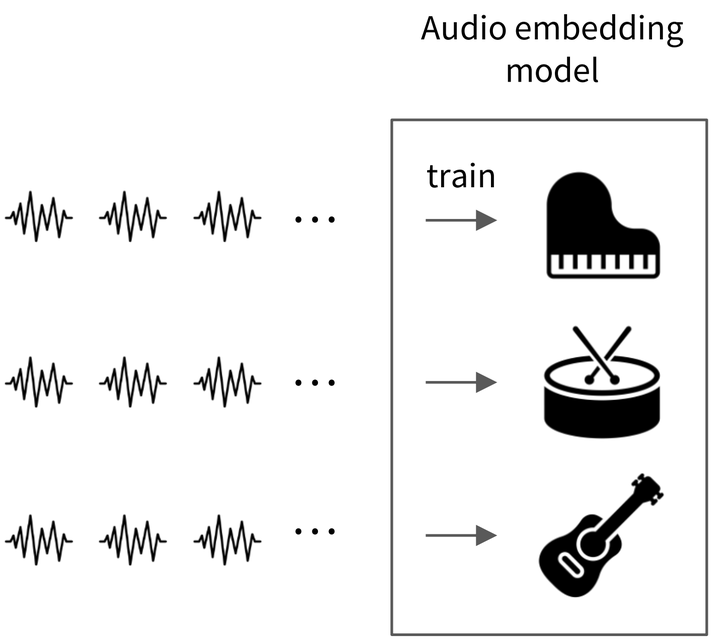
ZSL research branches (computer vision)

A comprehensive survey of zero-shot image classification: methods, implementation, and fair evaluation
Yang et al., Applied Computing and Intelligence2022


Audio embedding model
train
(train with 'seen' class labels)
audio embedding space

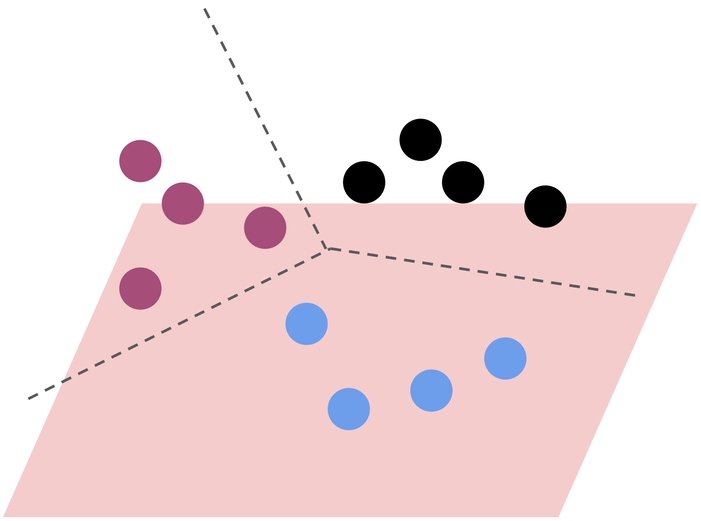


We can embed new audios,
but **how can we predict unseen classes**?

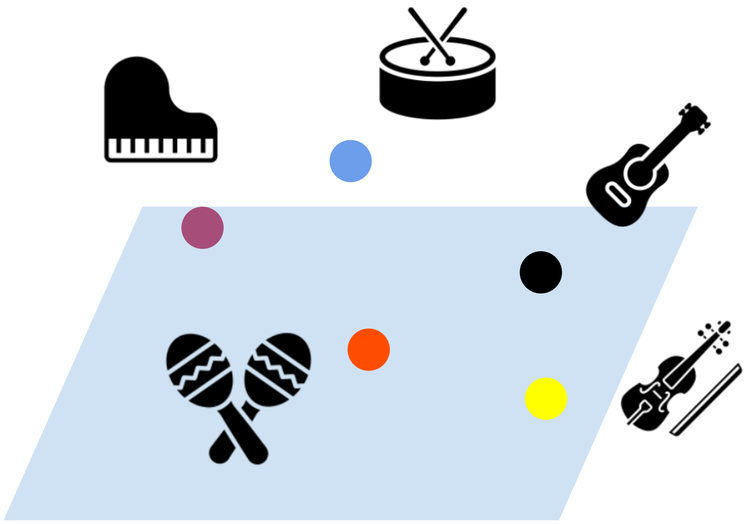


Use **side information** of class vectors,
- Class-attribute vector
- General semantic word vector (e.g. word2vec)
- General image embedding vector (e.g. trained with ImageNet)
- or etc..

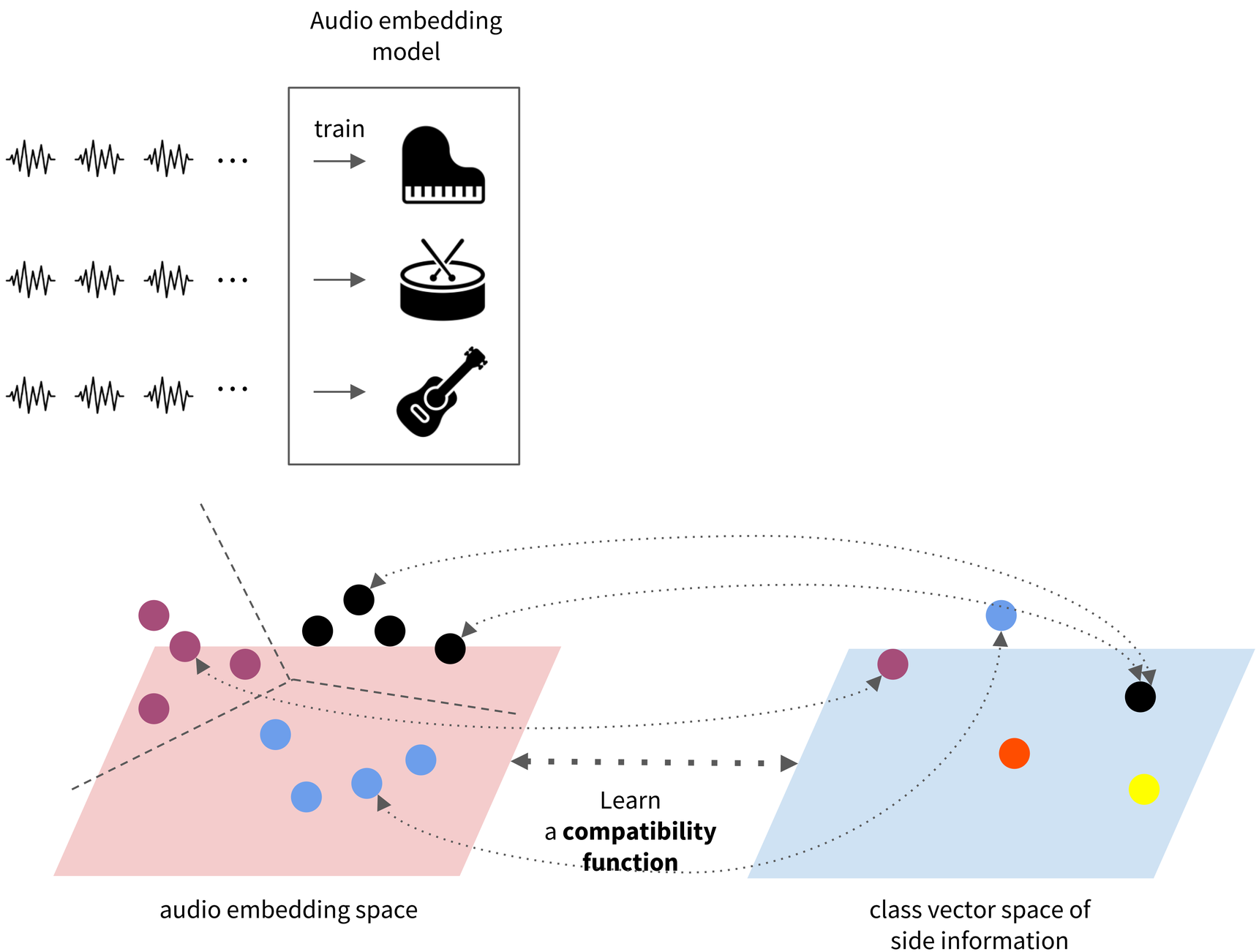
which can **generalize to new classes**.

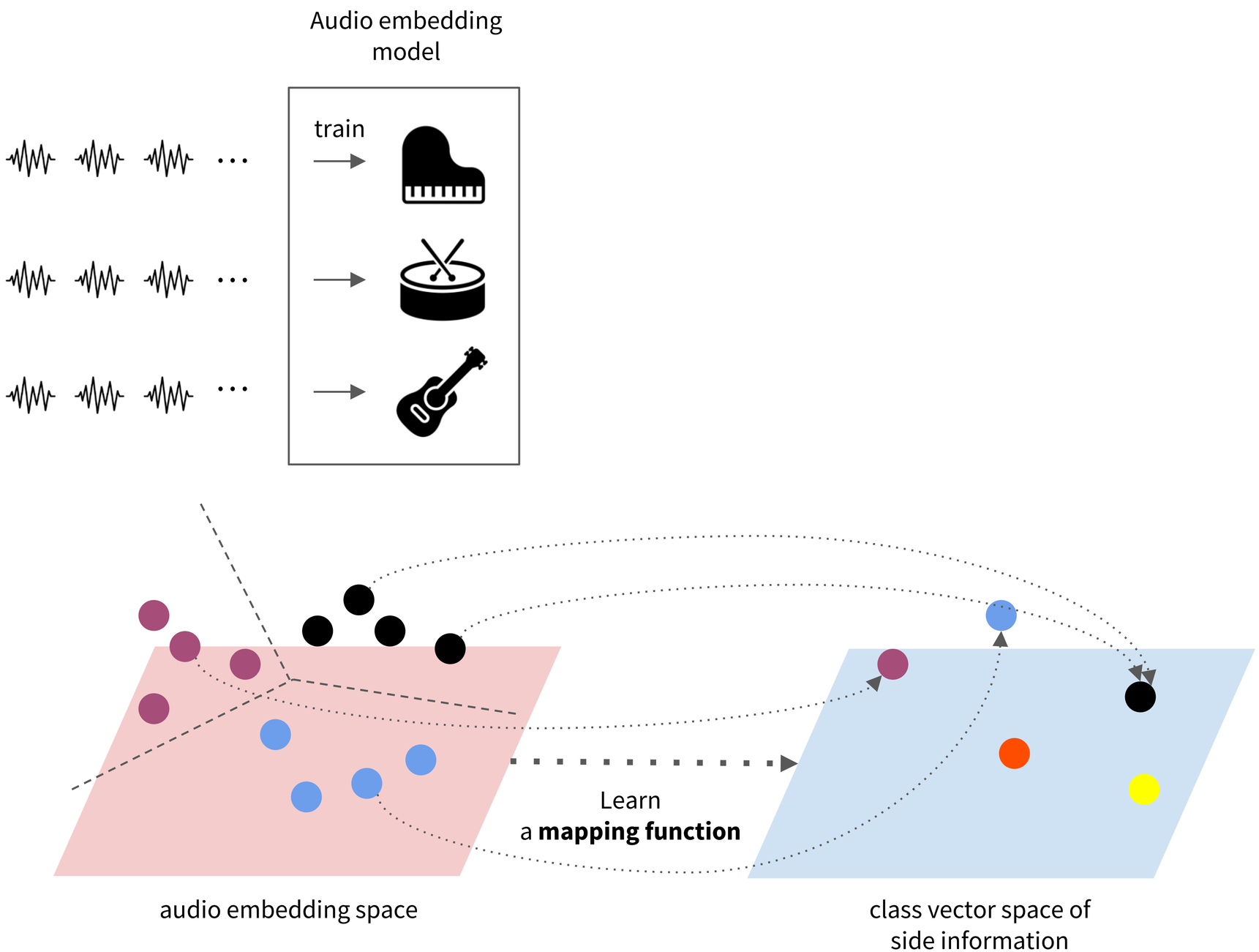


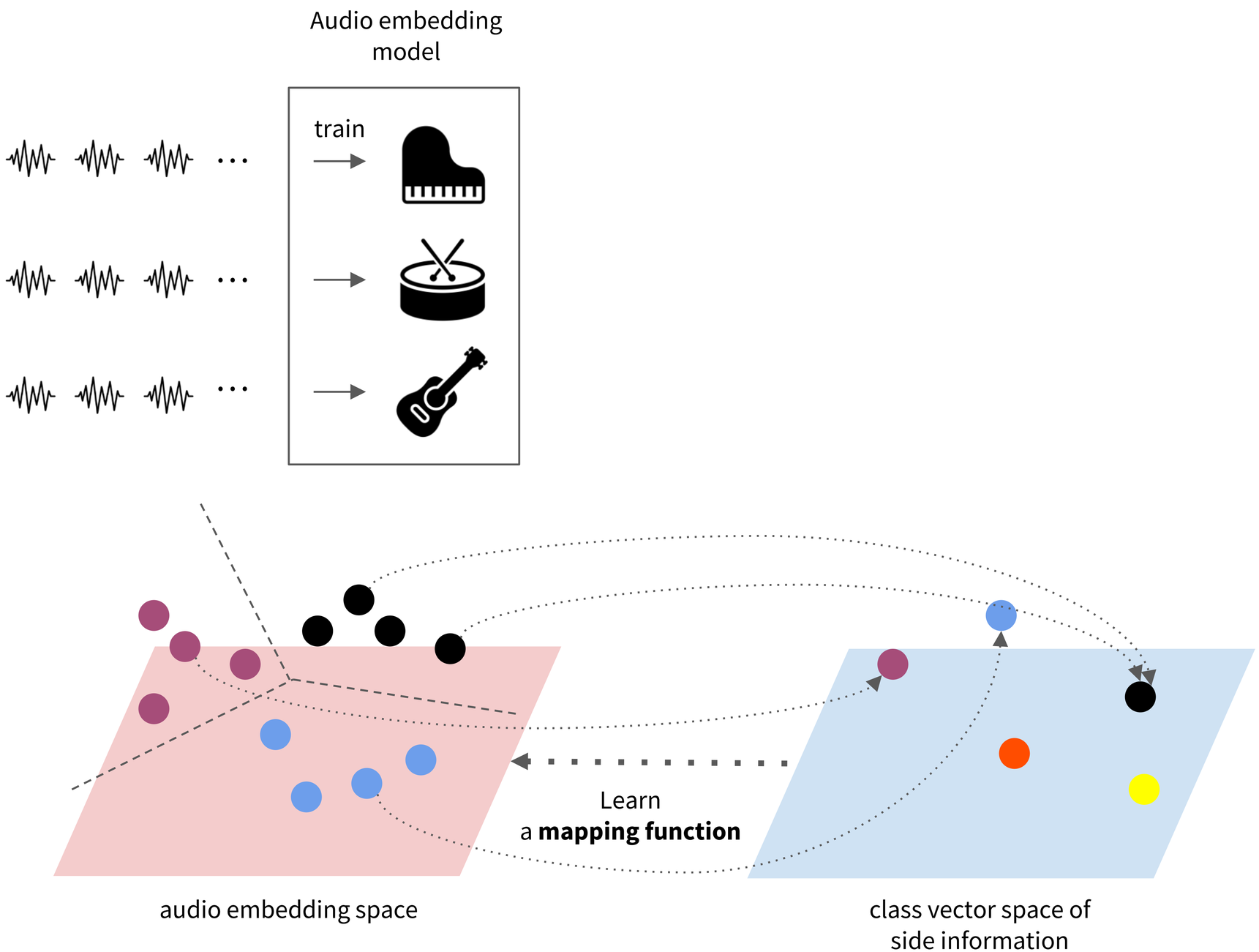
audio embedding space

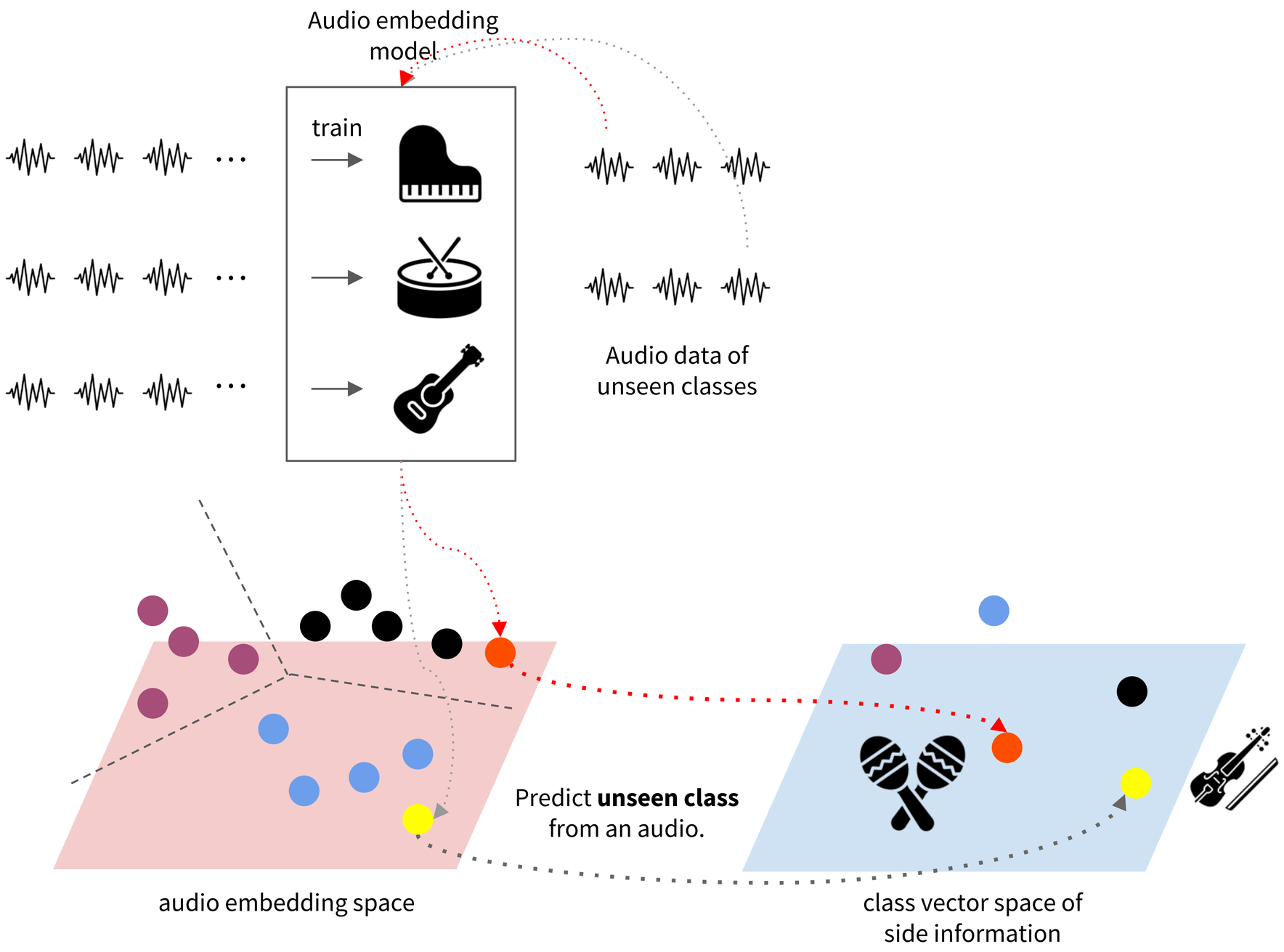


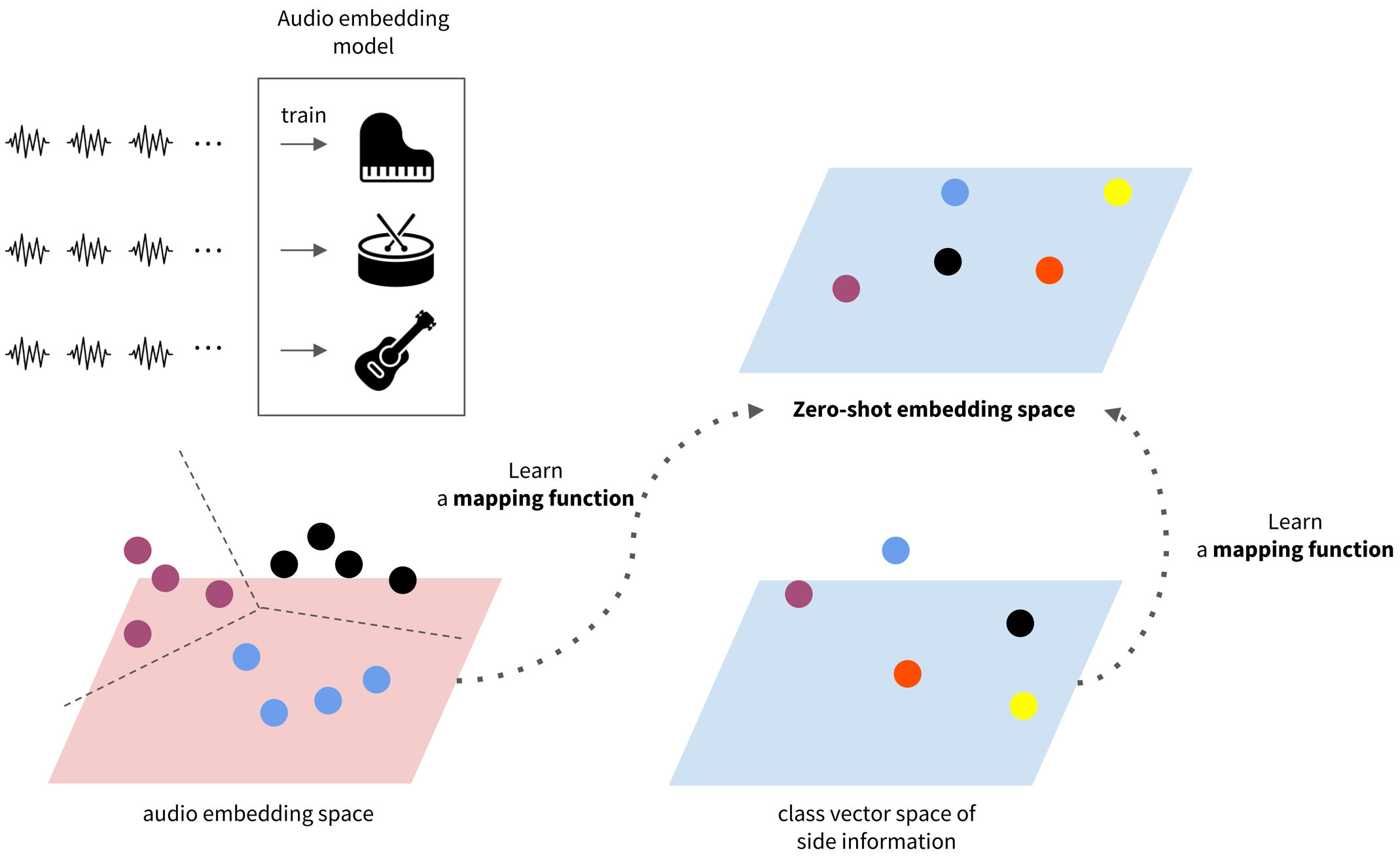
class vector space of
side information

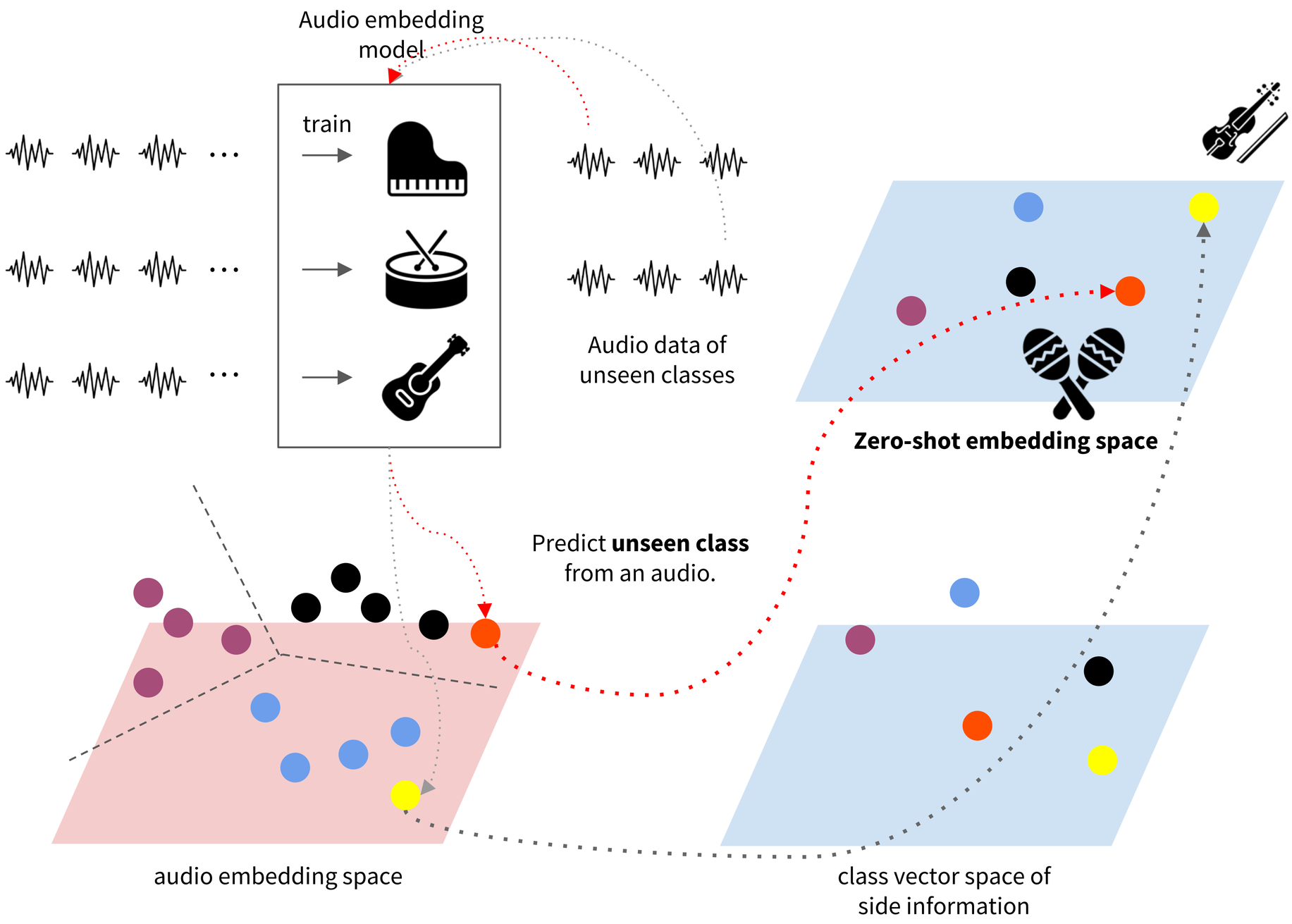

Audio embedding model
train
Learn
a **compatibility function**
audio embedding space
class vector space of side information


Audio embedding model
train
Learn
a **mapping function**
audio embedding space
class vector space of side information


Audio embedding model
train
Learn
a **mapping function**
audio embedding space
class vector space of side information


Audio embedding model
train
Audio data of unseen classes
Predict **unseen class** from an audio.
audio embedding space
class vector space of side information

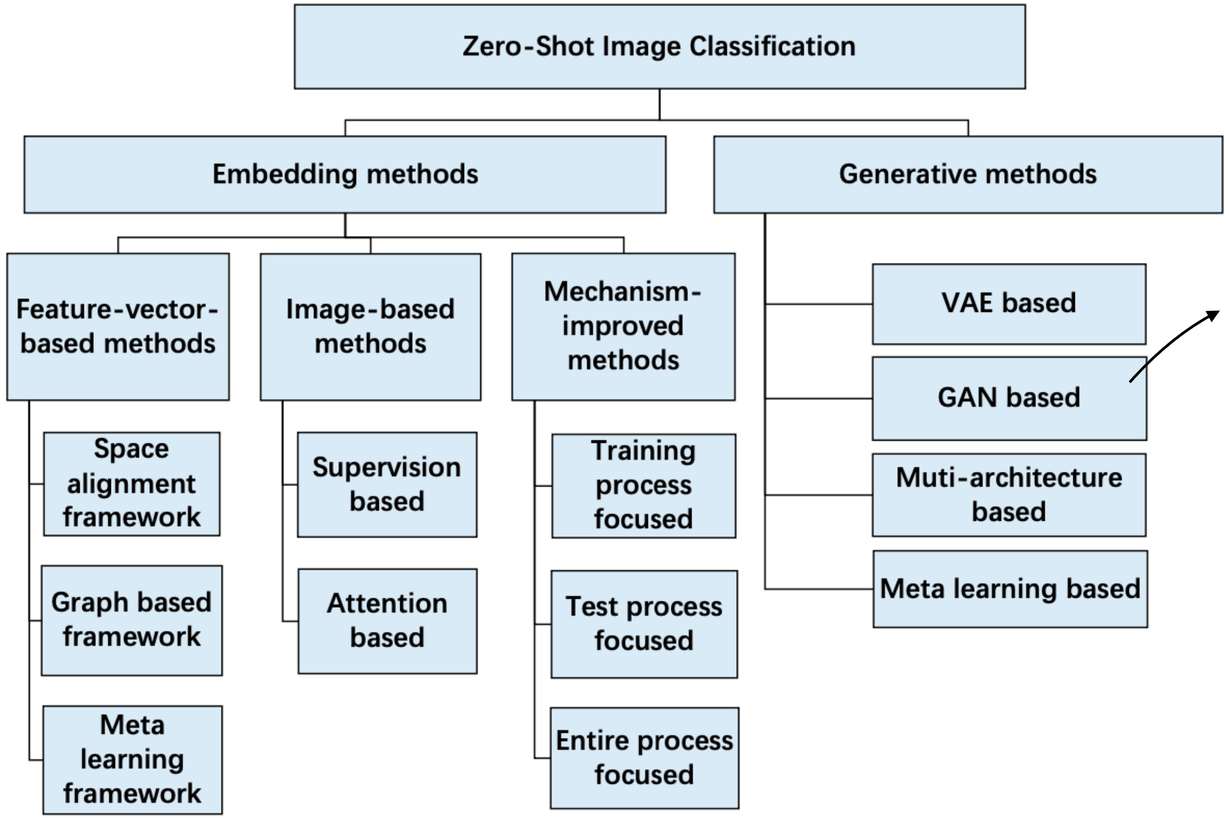

Audio embedding model
train
Zero-shot embedding space
Learn
a **mapping function**
Learn
a **mapping function**
audio embedding space
class vector space of
side information


Audio embedding model
train
Audio data of unseen classes
Zero-shot embedding space
Predict **unseen class** from an audio.
audio embedding space
class vector space of side information

Overview on ZSL Framework

- Basic framework (an example of a space-alignment framework).
- Generative method (an example of a GAN based framework).
- Data split & evaluation scheme
- Side information

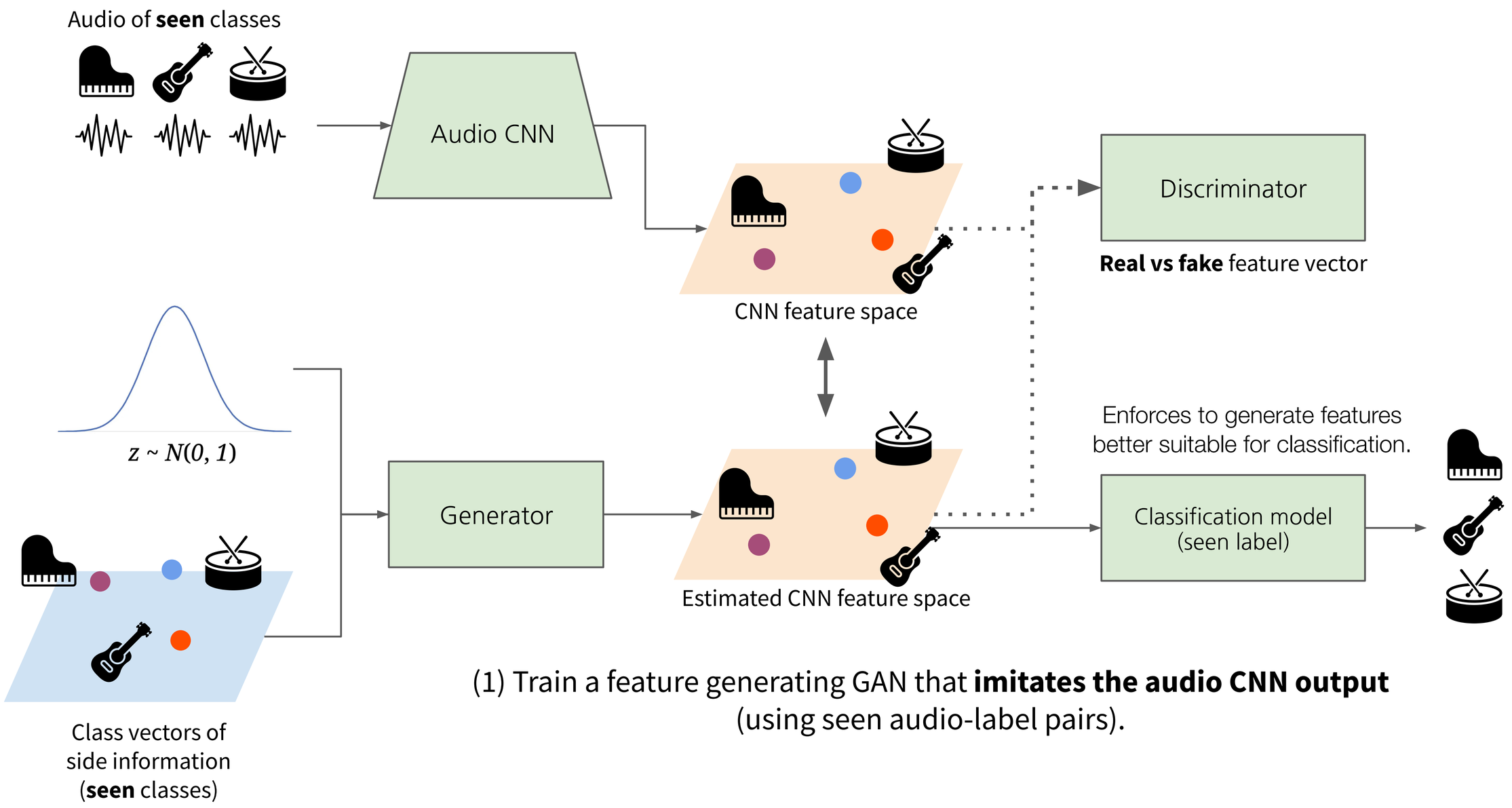
Overview on ZSL Framework



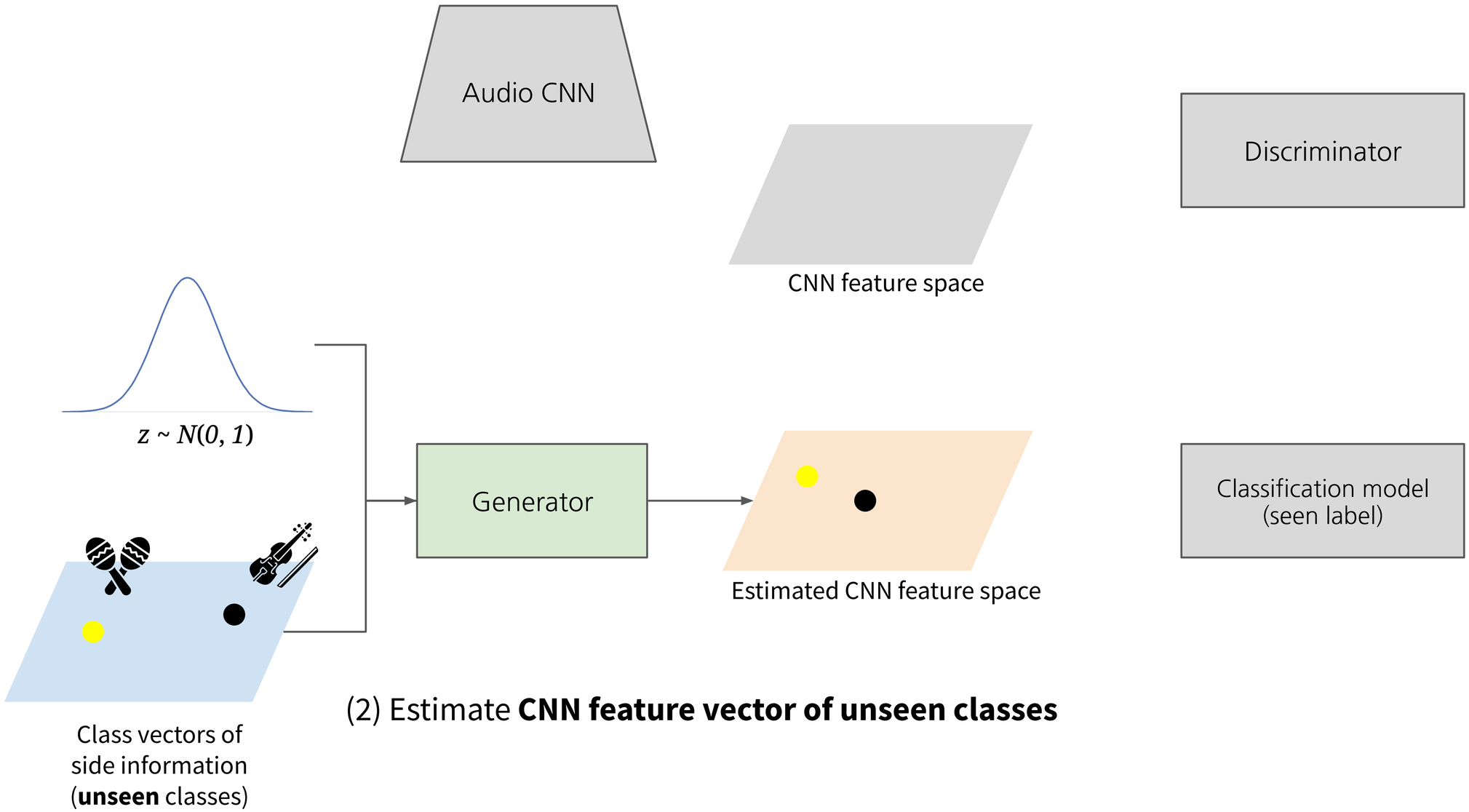
We'll go over this.

ZSL research branches (computer vision)

A comprehensive survey of zero-shot image classification: methods, implementation, and fair evaluation
Yang et al., Applied Computing and Intelligence2022

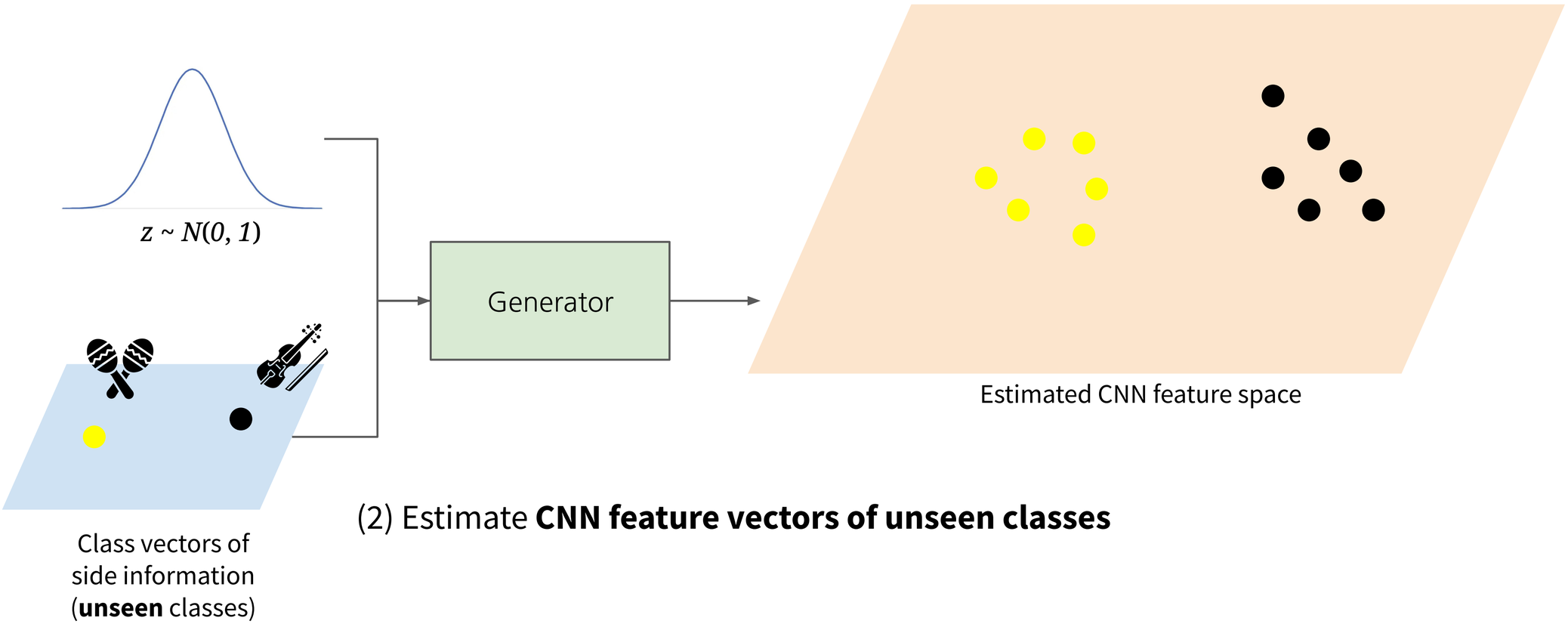


(1) Train a feature generating GAN that **imitates the audio CNN output** (using seen audio-label pairs).

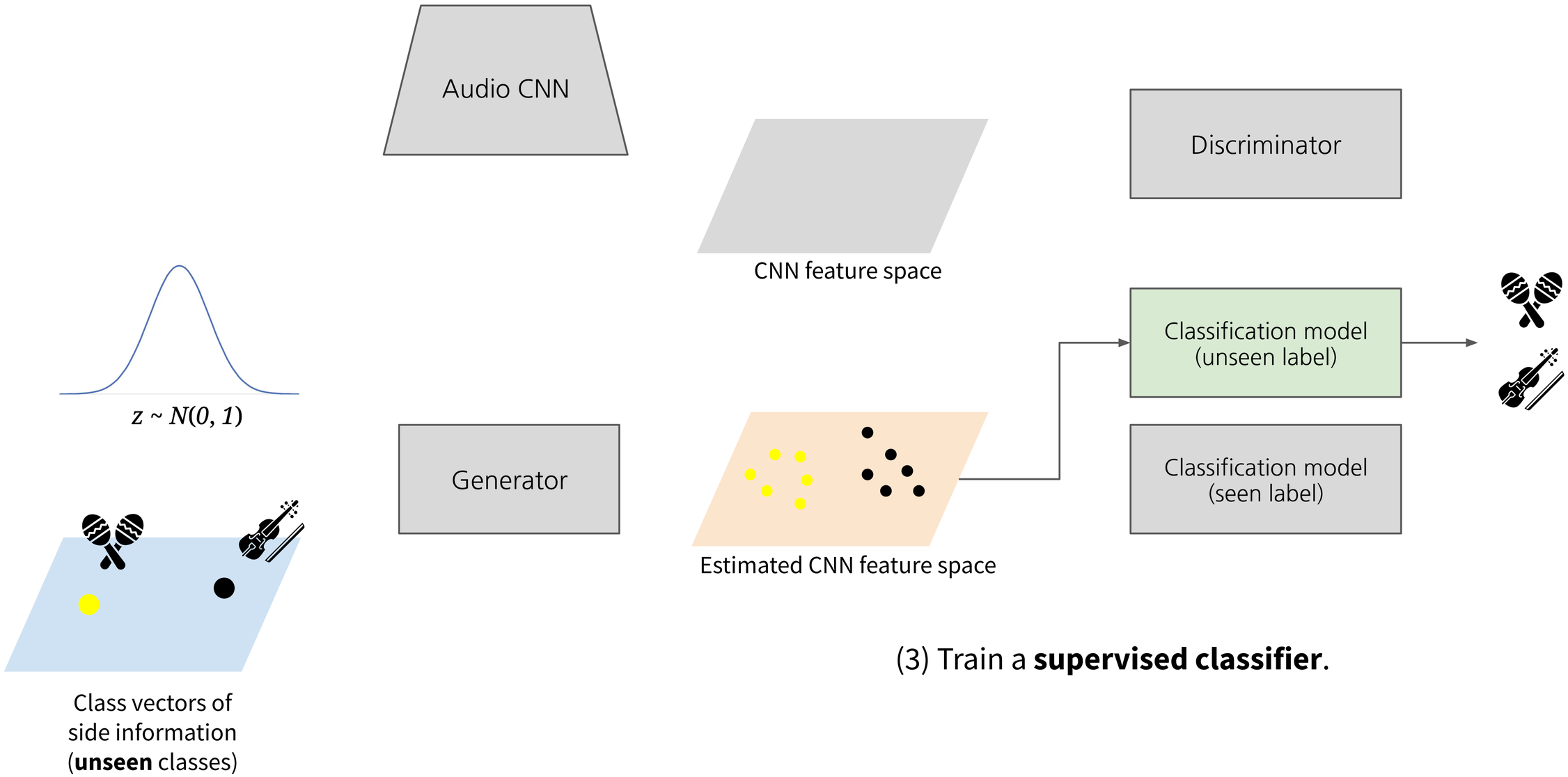


(2) Estimate **CNN feature vector of unseen classes**

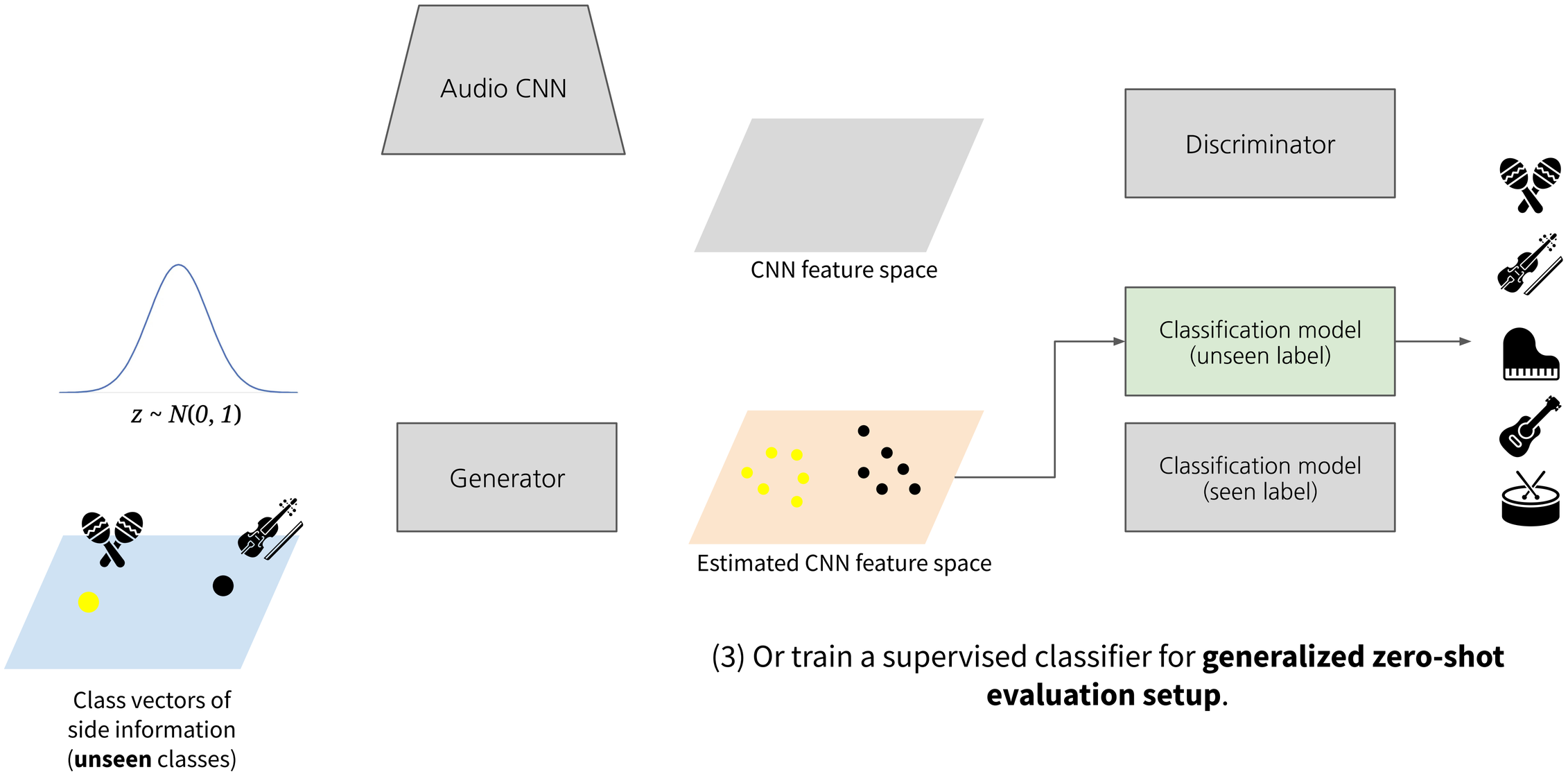
We can **sample as many as we want**!



(2) Estimate **CNN feature vectors of unseen classes**

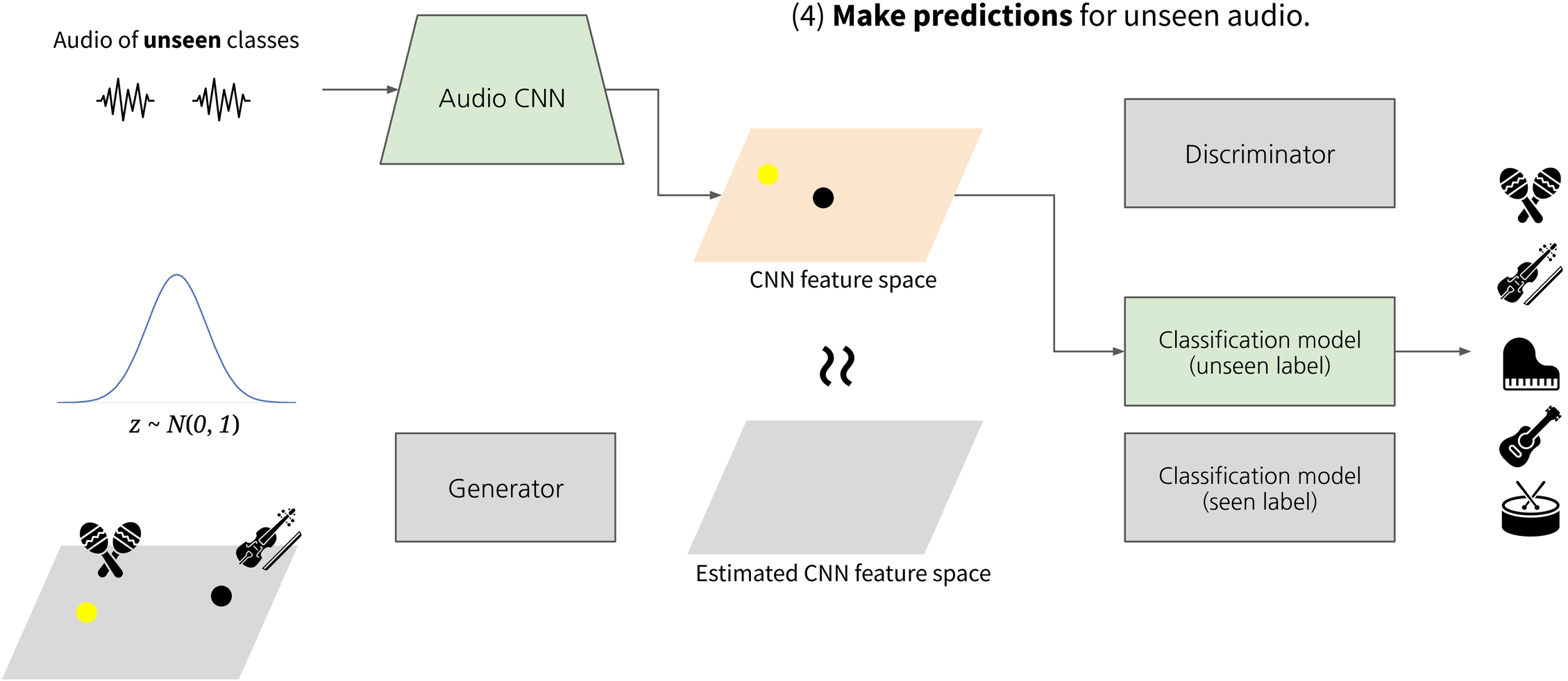


(3) Train a **supervised classifier**.



(3) Or train a supervised classifier for **generalized zero-shot evaluation setup**.

(4) **Make predictions** for unseen audio.

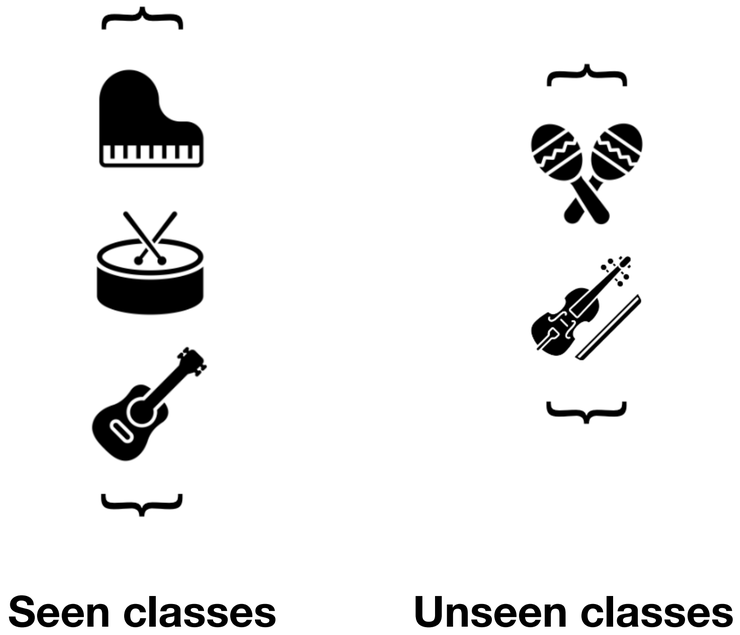


Data Split Scheme in ZSL

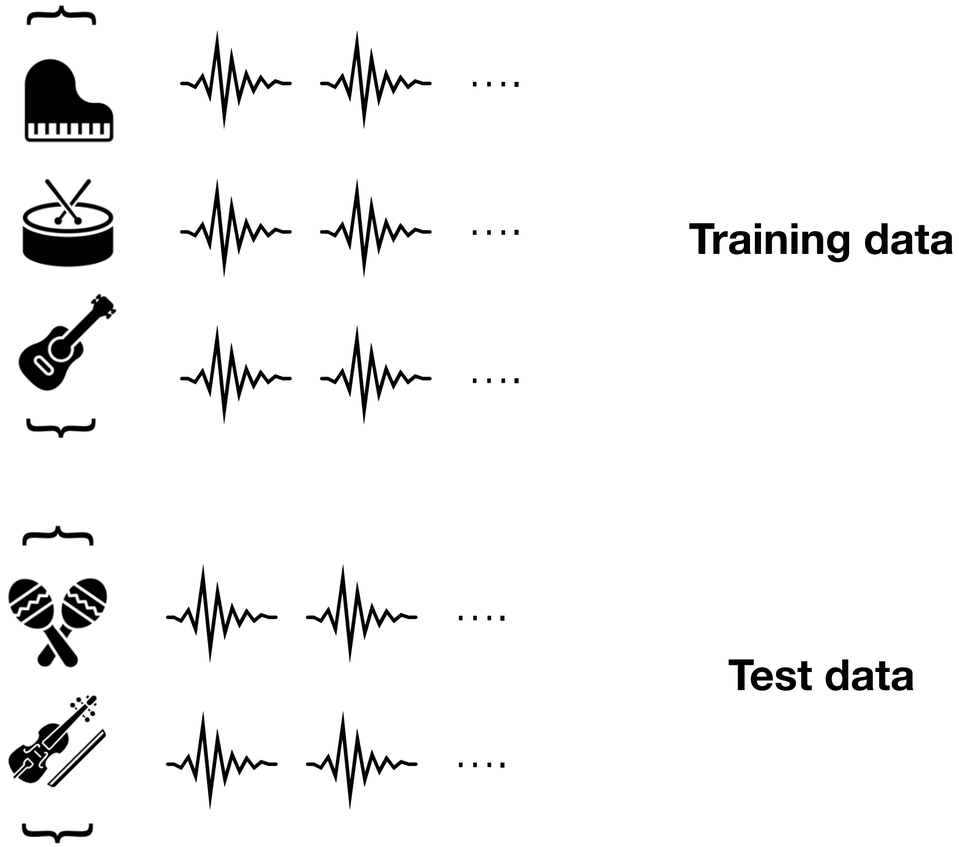
- Basic framework (an example of a space-alignment framework).
- Generative method (an example of a GAN based framework).
- Data split & evaluation scheme
- Side information

Data Split Scheme in ZSL

(1) **First, split labels** into seen and unseen groups.



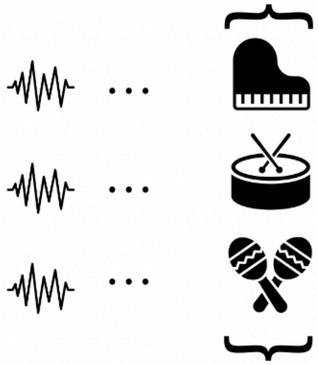
(2) Next, **split data samples**. All data samples associated with seen classes are used during training, and ones with unseen classes are used for evaluation.



Data Split Scheme in ZSL (Training Input Scope)

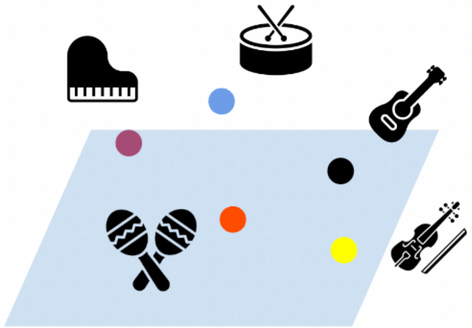
Inductive ZSL

Data samples
paired with seen classes



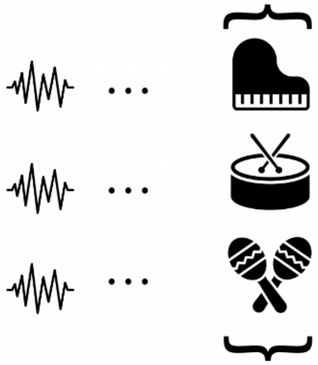
+

Side information of
both seen and unseen classes



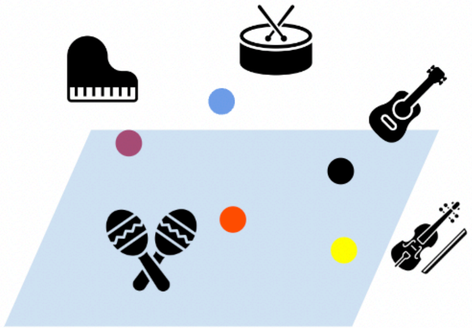
Transductive ZSL

Data samples
paired with seen classes



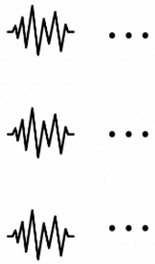
+

Side information of
both seen and unseen classes



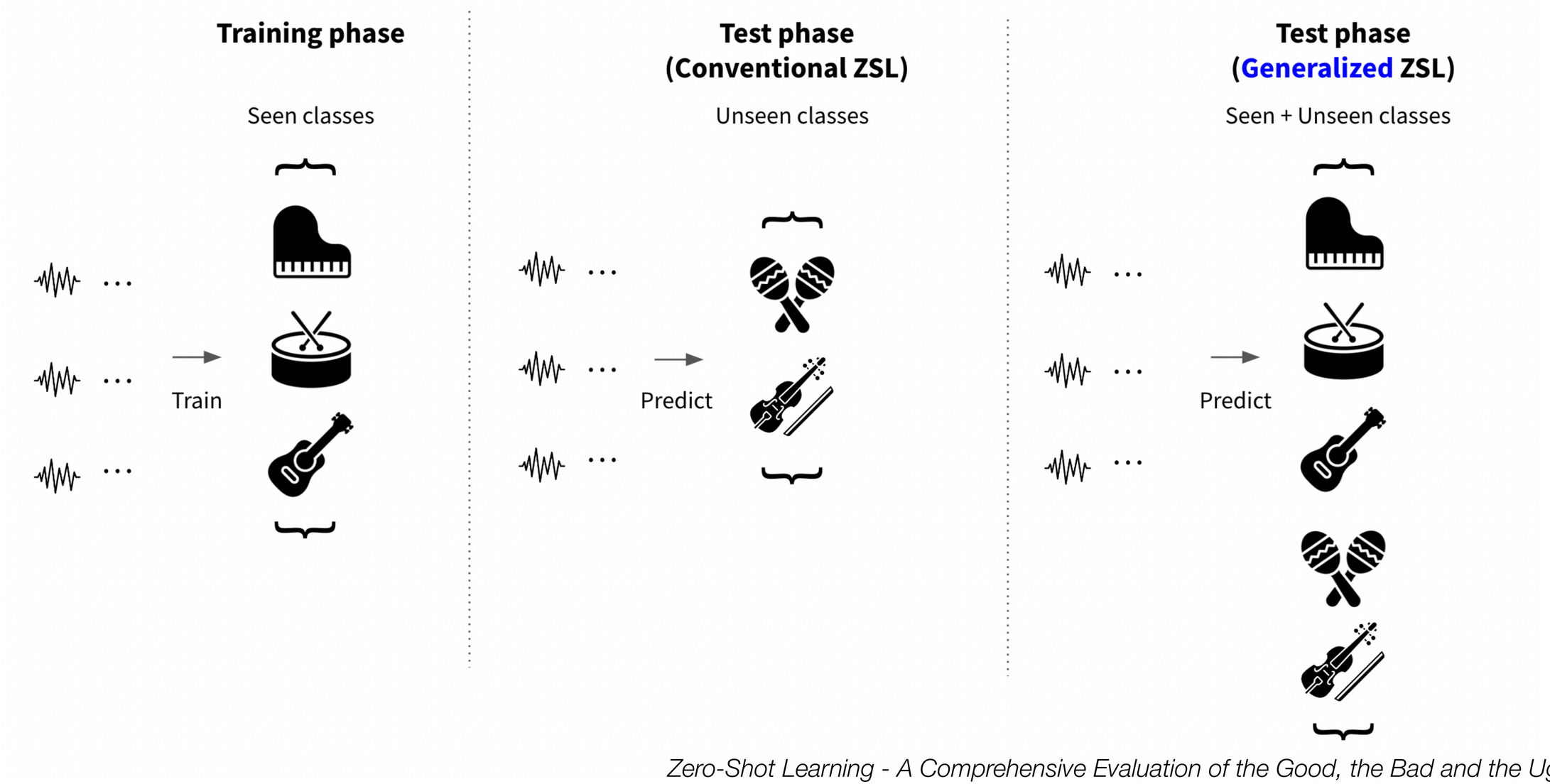
Data samples of unseen classes
NOT paired

+



Transductive Multi-view Zero-Shot Learning
Fu et al., CVPR 2015

Evaluation in ZSL



Zero-Shot Learning - A Comprehensive Evaluation of the Good, the Bad and the Ugly
Xian et al., CVPR 2017

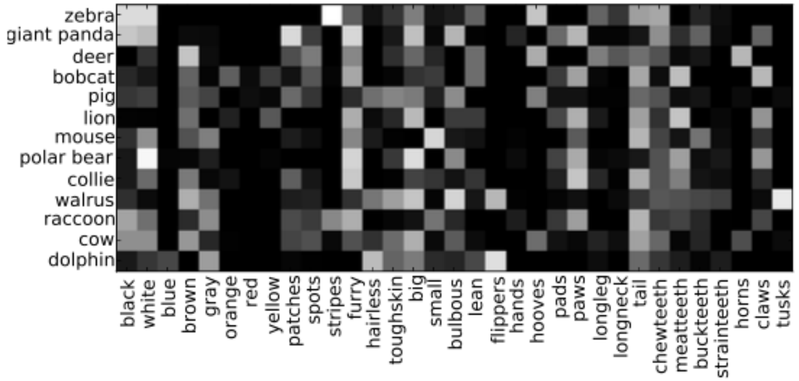
Overview on ZSL Framework

- Basic framework (an example of a space-alignment framework).
- Generative method (an example of a GAN based framework).
- Data split & evaluation scheme
- Side information

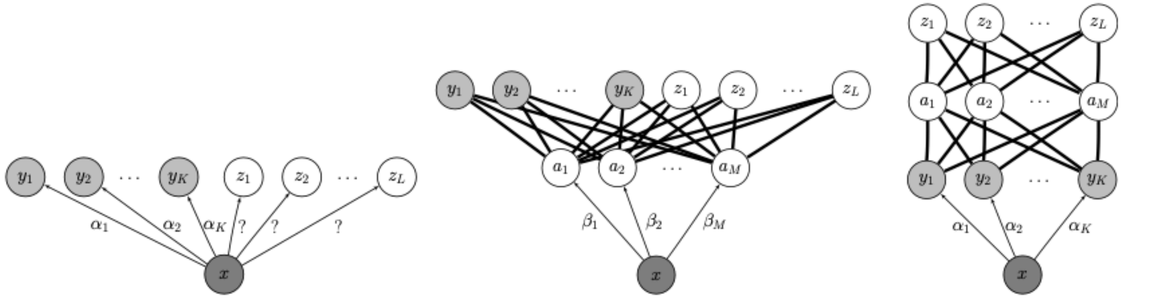
Side Information

Class-attribute datasets

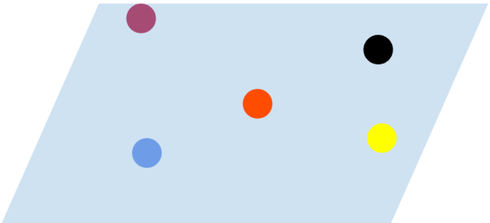
Human annotated visual attributes for each label class.
e.g. Animals With Attributes (AWA)



Examples of animal class and attribute mappings in AWA dataset.



Train the **intermediate attribute classifiers.**



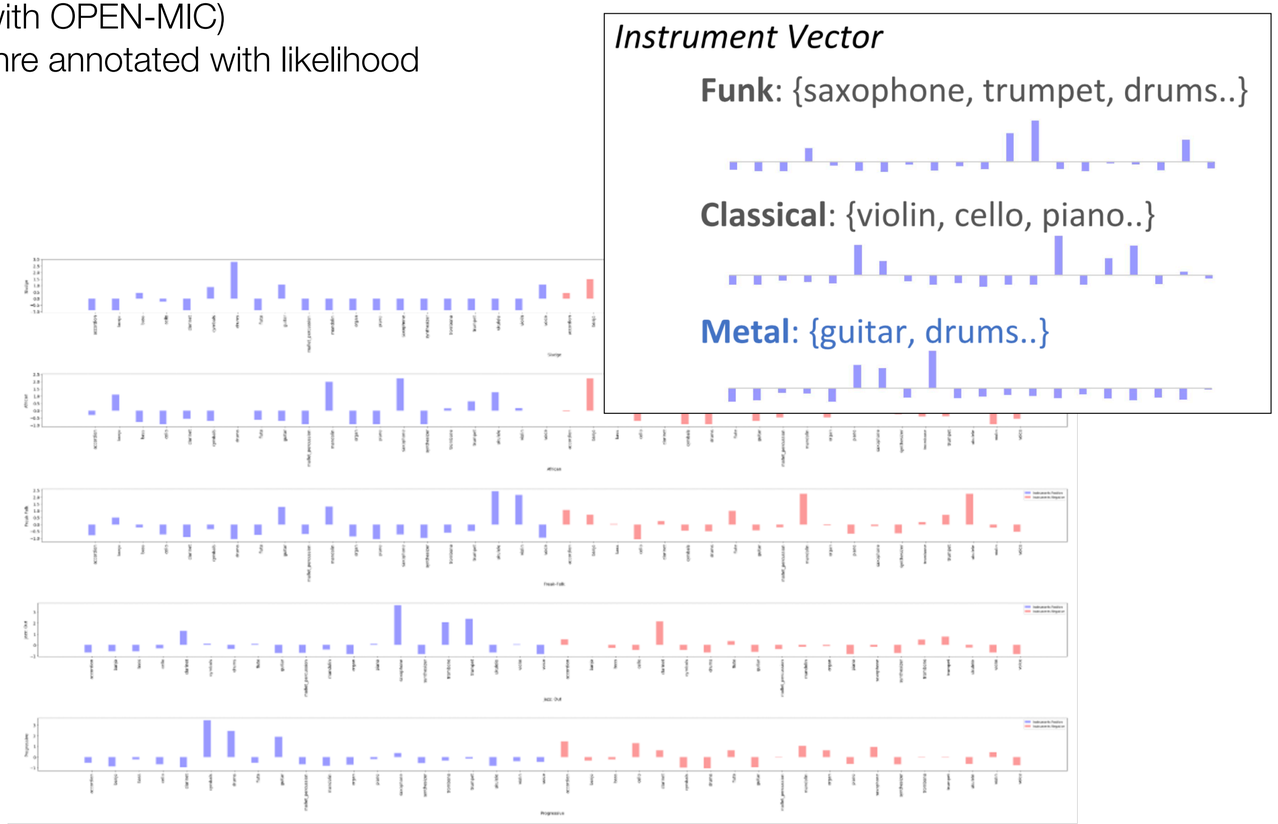
Formulate the **class embedding vector space**.

Attribute-based classification for zero-shot visual object categorization
Lampert et al., transactions on pattern analysis and machine intelligence, 2013

Side Information

Class-attribute datasets in MIR

e.g. Genre classes with instrument attributes (FMA with OPEN-MIC)
(Audio annotations of 157 genre labels with each genre annotated with likelihood measures of 20 instruments)

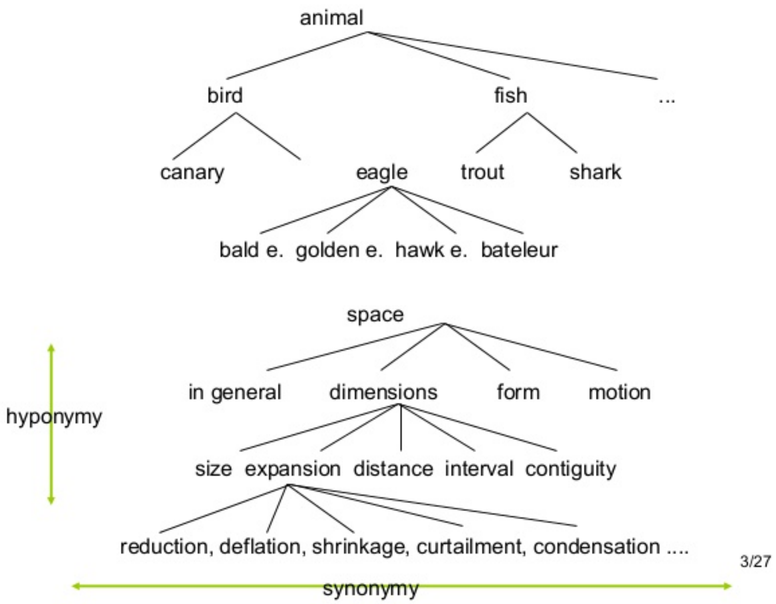


Examples of normalized 40-dim (20 pos / 20 neg) **instrument vectors for each genre** constructed from the overlapping classes of FMA and OPEN-MIC datasets.

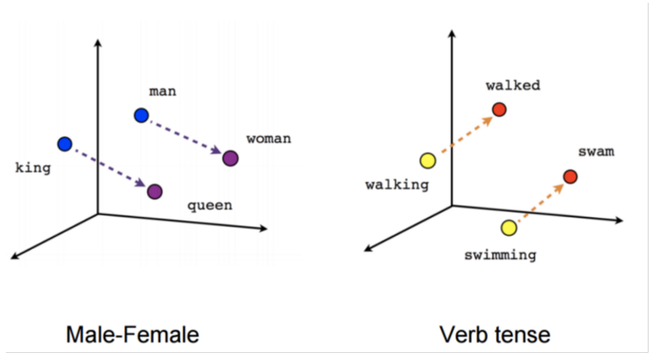
Side Information

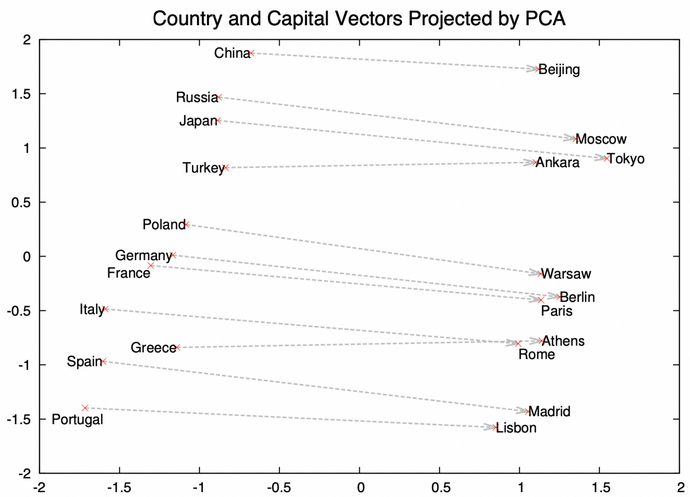
Textual data embedding space

- Hierarchical relationship between words.
- Semantic word embedding trained with a large document corpus.



Example of **Wordnet** semantic hierarchy.



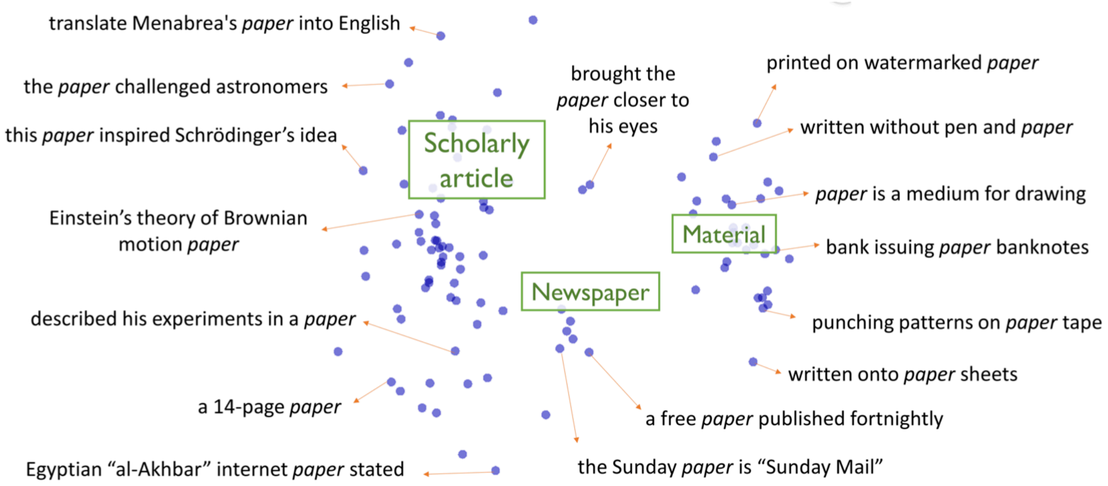


Example of relative positions in **Word2vec** embedding space.

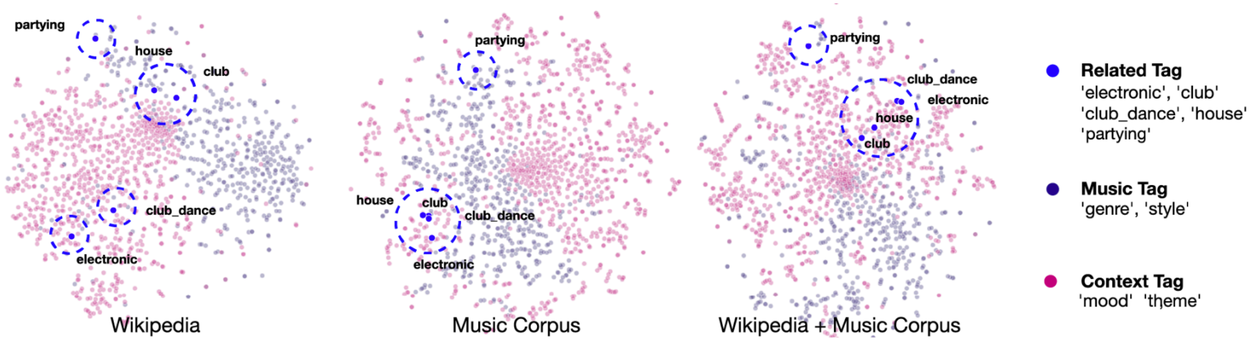
Side Information

Textual data embedding space

- Hierarchical relationship between words.
- Semantic word embedding trained with a large document corpus.



Example of **BERT** embedding vectors
(different embeddings depending on the context for the same word).

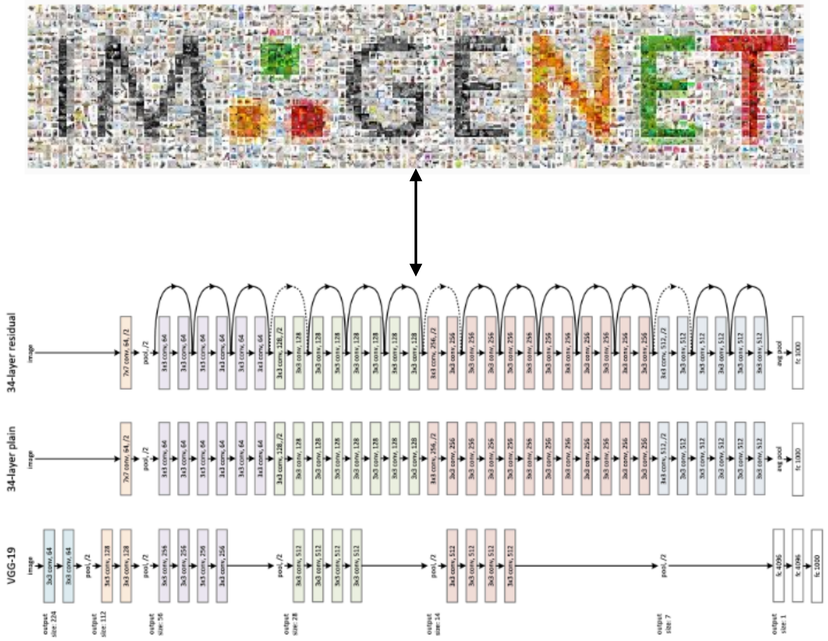


Example of **musical word** embedding vectors.

Musical Word Embedding: Bridging the Gap between Listening Contexts and Music
S Doh et al., ICML 2020 ML4MD Workshop

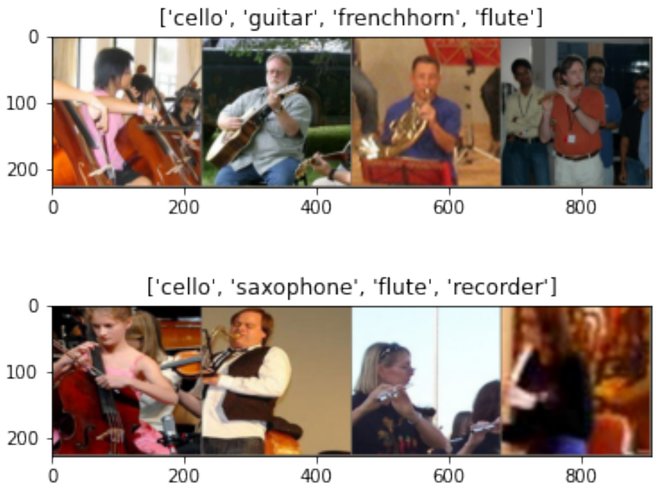
Side Information

Image embedding space

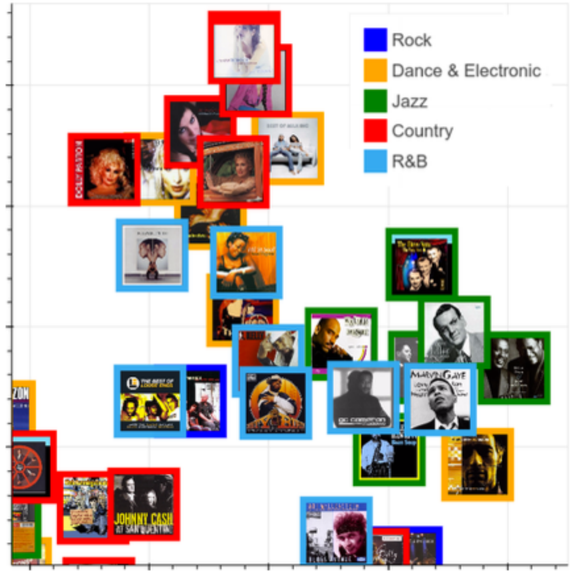


An image embedding model pretrained with large image dataset
(e.g Resnet-101 trained on ImageNet)

+



Example of **instrument images.**
People Playing Musical Instrument (PPMI) dataset.



Example of **album cover images.**
MuMu dataset.

Multi-label Music Genre Classification from Audio, Text, and Images Using Deep Features
Oramas et al., ISMIR 2017

Summary of Part 1: Foundations

1. Introduction of FSL/ZSL
2. Foundations of FSL
   - K-way N-shot classification
   - Episodic training
   - Different approaches
3. Foundations of ZSL
   - Side information
   - Research direction

Tutorial Outline

- Part 1: Introduction & Foundations
- Part 2: Coding Examples: Musical instrument recognition
- Part 3: Recent Advances
- Closing Remarks

Tutorial Outline

- Part 1: Introduction & Foundations
- Part 2: Coding Examples
- Part 3: Recent Advances
- Closing Remarks

Recent Works in MIR Leveraging FSL/ZSL

Classification

Transcription

Source Separation

Recent Works in MIR Leveraging FSL/ZSL

Classification

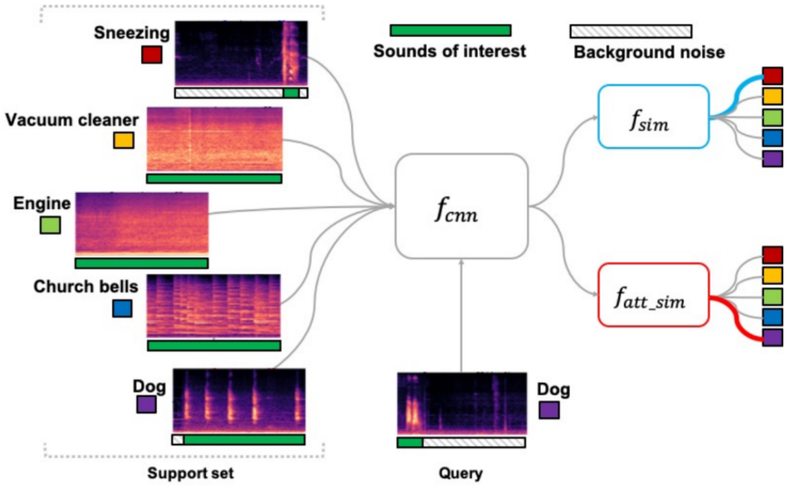
Transcription

Source Separation

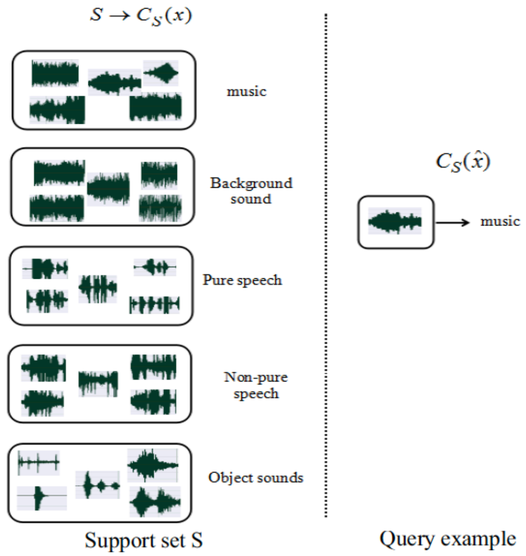
Few-Shot Sound Classification

- Early works focused on other audio domains
- Standard K-way N-shot classification problem

Learning to match transient sound events using attentional similarity for few-shot sound recognition (Chou et al. ICASSP 2019)



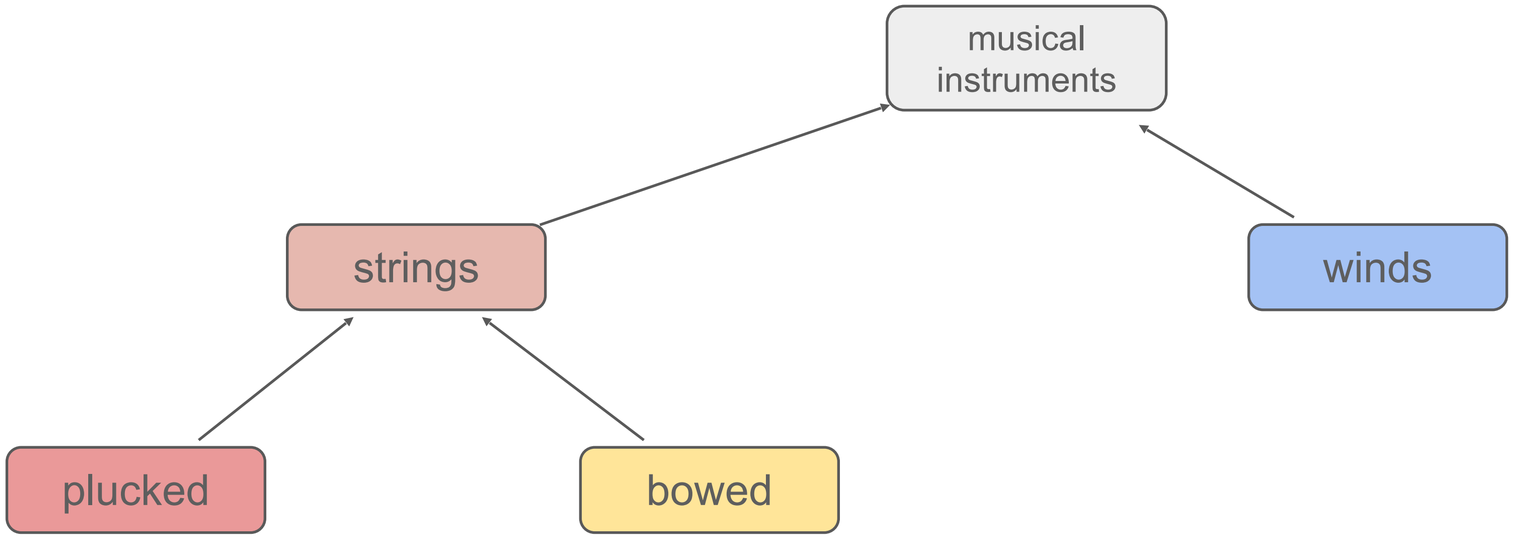
Few-Shot Audio Classification with Attentional Graph Neural Networks (Zhang et al. INTERSPEECH 2019)



Hierarchical Musical Instrument Recognition

H. Flores Garcia, A. Aguilar, E. Manilow, and B. Pardo, "Leveraging Hierarchical Structures for Few-Shot Musical Instrument Recognition," *(ISMIR 2021)*.

musical instrument hierarchy



guitar

cuatro

transverse flute

quenacho

lots of data!

little to no data (unseen during training)

lots of data!

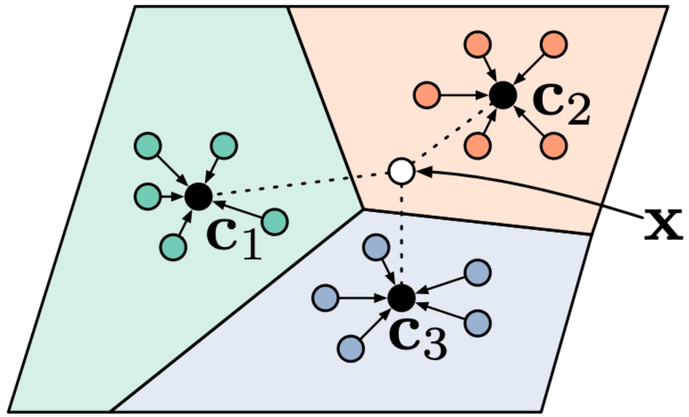
little to no data (unseen during training)

H. Flores Garcia, A. Aguilar, E. Manilow, and B. Pardo, "Leveraging Hierarchical Structures for Few-Shot Musical Instrument Recognition," *(ISMIR 2021)*.

how do we learn **new** instruments **on the fly,** taking advantage of their **hierarchical relationships**?

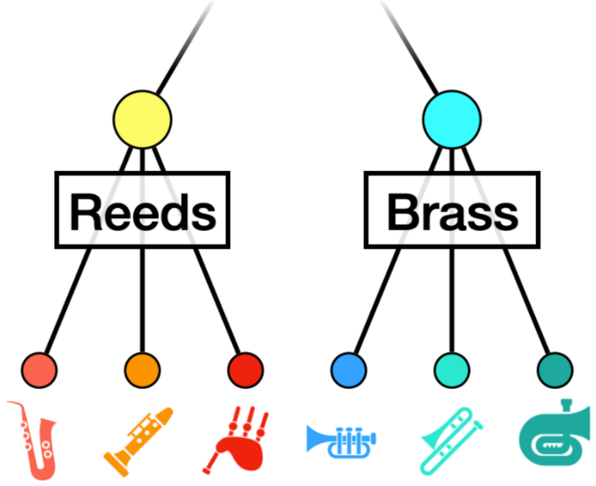
H. Flores Garcia, A. Aguilar, E. Manilow, and B. Pardo, "Leveraging Hierarchical Structures for Few-Shot Musical Instrument Recognition," *(ISMIR 2021)*.

the big idea:
[hierarchical?] prototypical networks?



protonets: learn new classes with a few shots

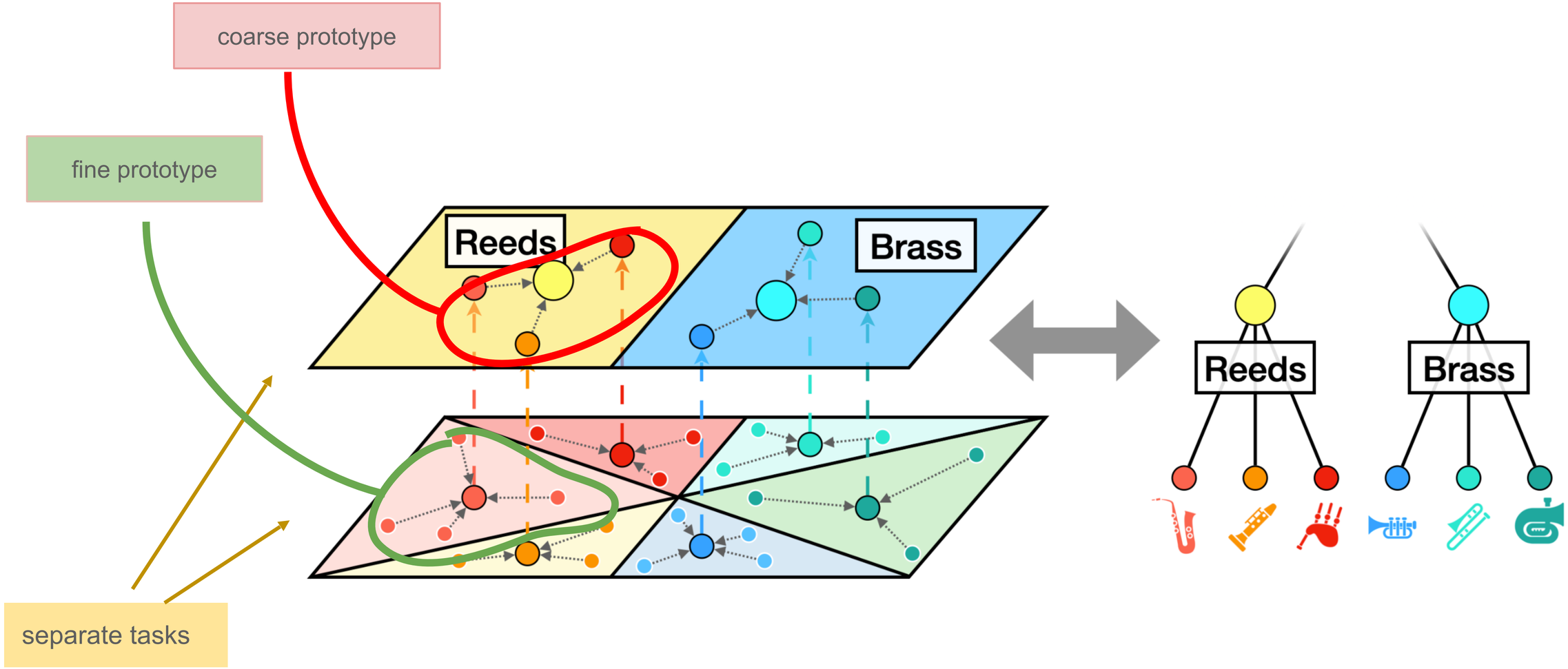
+



hierarchical relationships in musical instruments

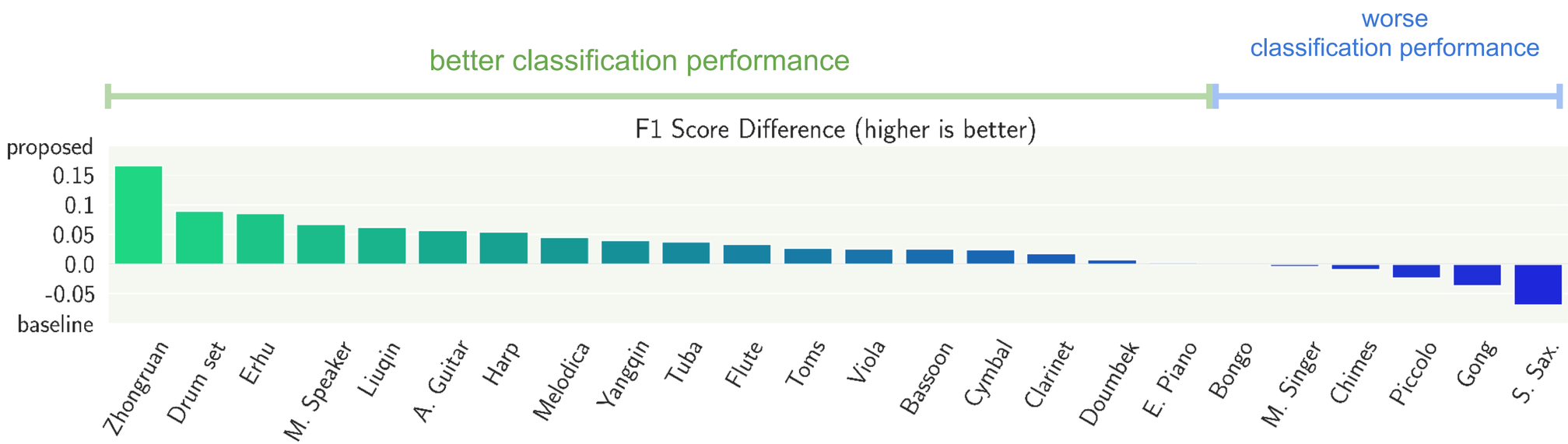
H. Flores Garcia, A. Aguilar, E. Manilow, and B. Pardo, "Leveraging Hierarchical Structures for Few-Shot Musical Instrument Recognition," *(ISMIR 2021)*.

Hierarchical Prototypical Networks



H. Flores Garcia, A. Aguilar, E. Manilow, and B. Pardo, "Leveraging Hierarchical Structures for Few-Shot Musical Instrument Recognition," *(ISMIR 2021)*.

Is the hierarchical protonet capable of classifying a **wider range** of **musical instruments**?



our model is better at identifying a wider range of instrument classes!

H. Flores Garcia, A. Aguilar, E. Manilow, and B. Pardo, "Leveraging Hierarchical Structures for Few-Shot Musical Instrument Recognition," *(ISMIR 2021)*.

Takeaways

what we did:

A simple extension to **prototypical networks** that incorporates a class **hierarchy**

Significantly **better classification** performance **than** a **non-hierarchical** baseline

Makes **less severe mistakes**

what we can do now:

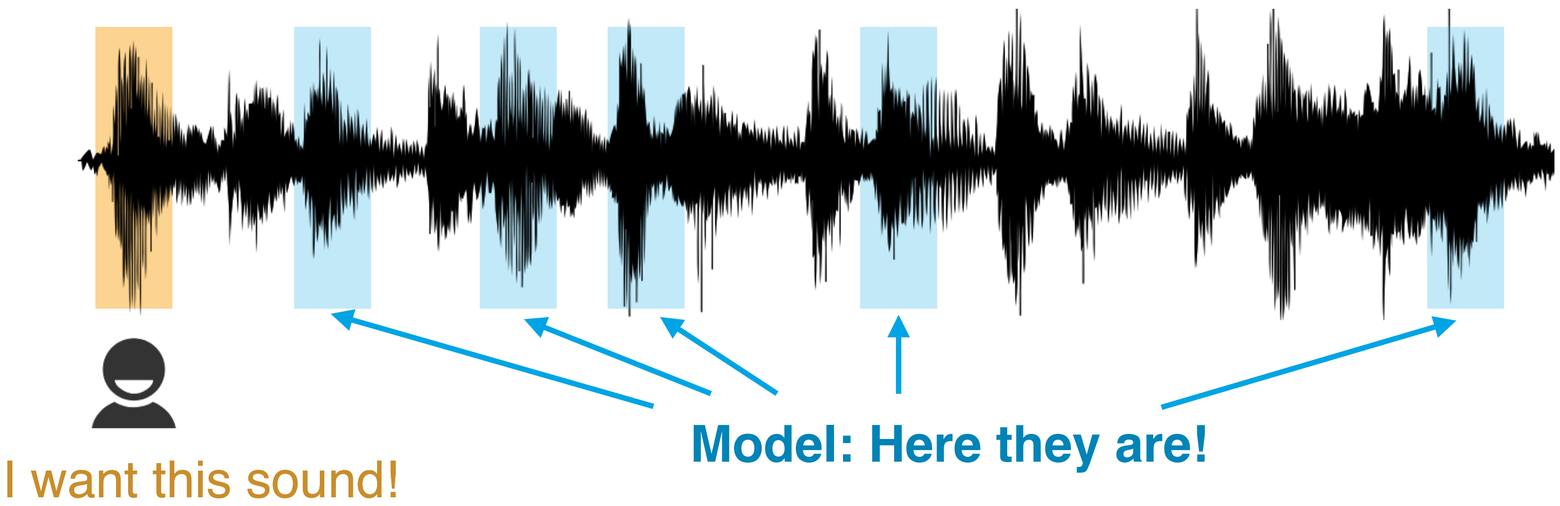
Musical instrument recognition for **classes** defined by the **end user**

deploying instrument recognition in audio editing software!

H. Flores Garcia, A. Aguilar, E. Manilow, and B. Pardo, "Leveraging Hierarchical Structures for Few-Shot Musical Instrument Recognition," *(ISMIR 2021)*.

Few-Shot Sound Event Detection

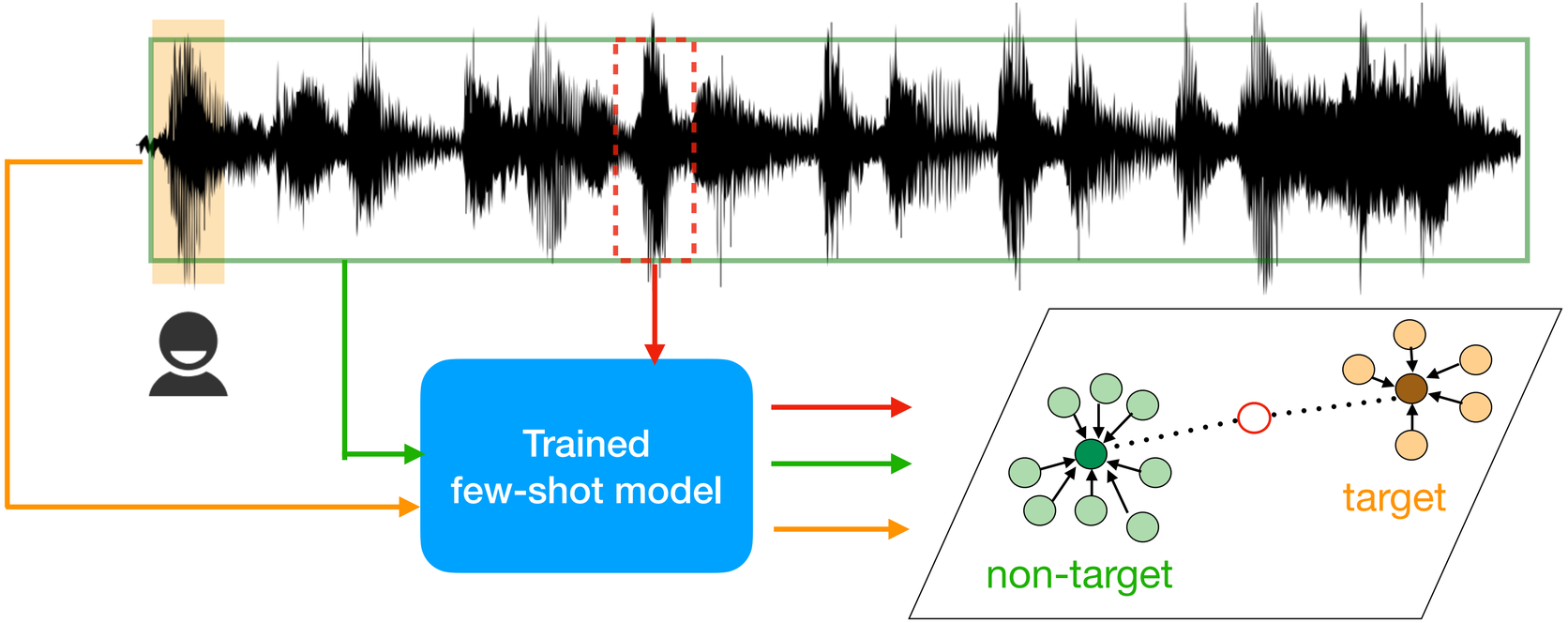
Goal: Ctrl-F for audio



1. No prior knowledge of the target sound event
2. Minimal human effort: label up to 5 examples
3. Real-time, on-the-fly detection

Few-Shot Sound Event Detection
Y. Wang *, J. Salamon, N. J. Bryan, and J. P. Bello, ICASSP, 2020*

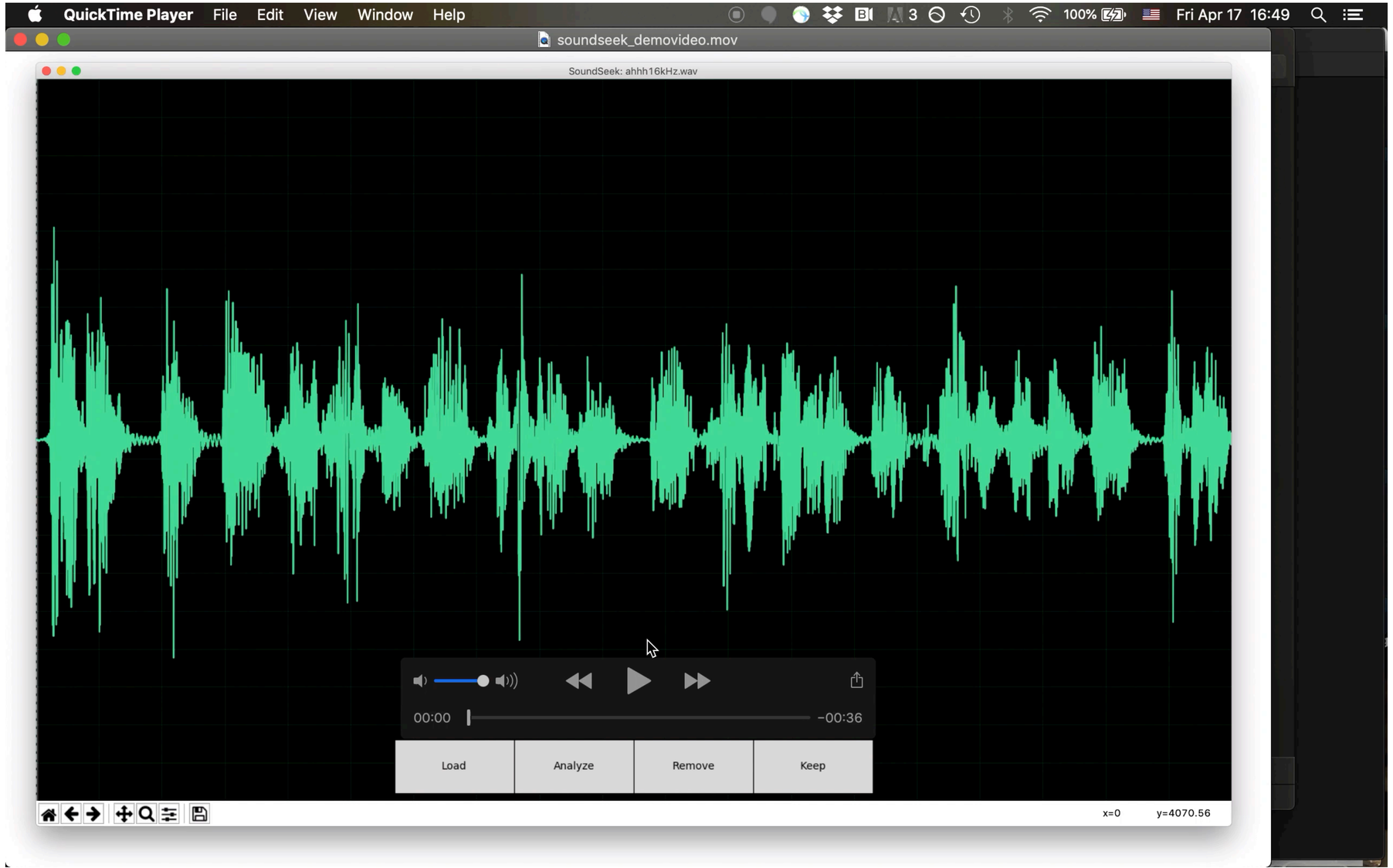
Few-Shot Sound Event Detection



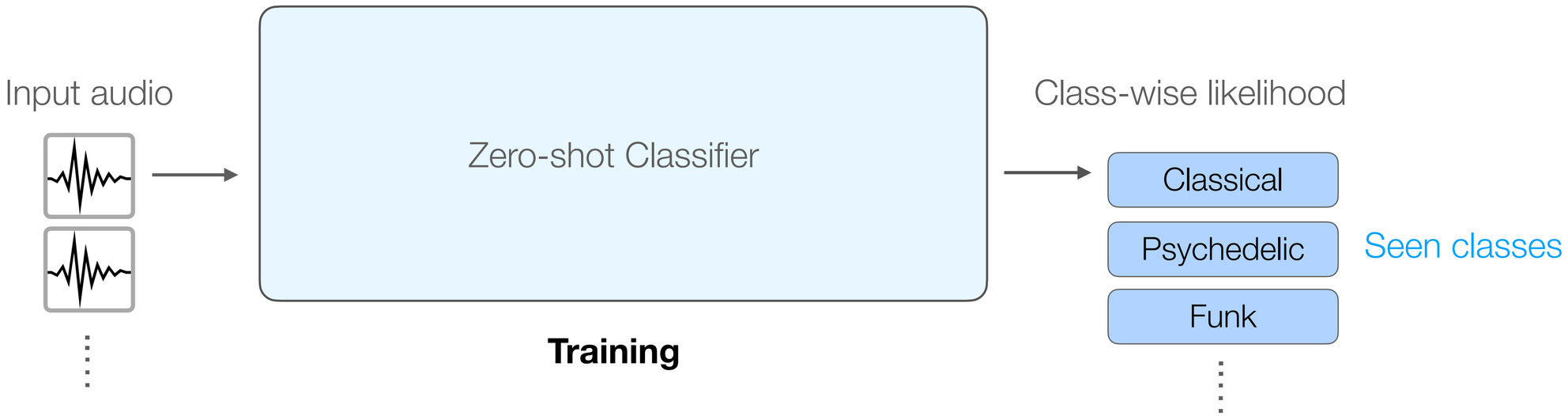
Take the entire track as the support set for non-target class

Few-Shot Sound Event Detection
Y. Wang *, J. Salamon, N. J. Bryan, and J. P. Bello, ICASSP, 2020*

Few-Shot Sound Event Detection

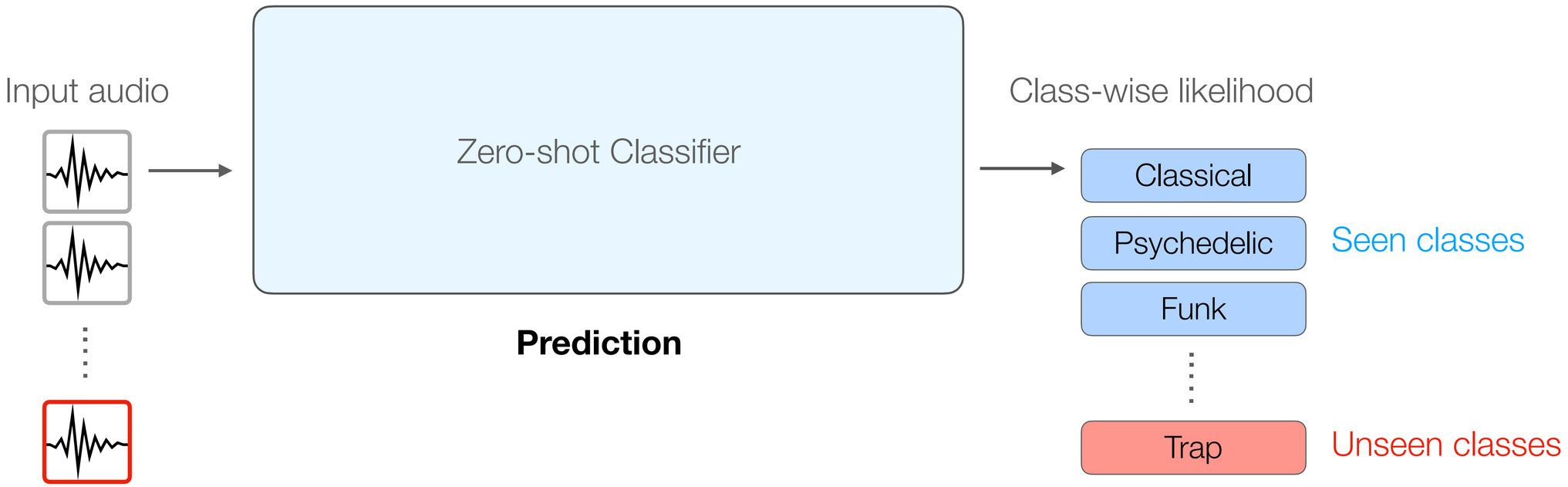


Zero-Shot Music Tagging



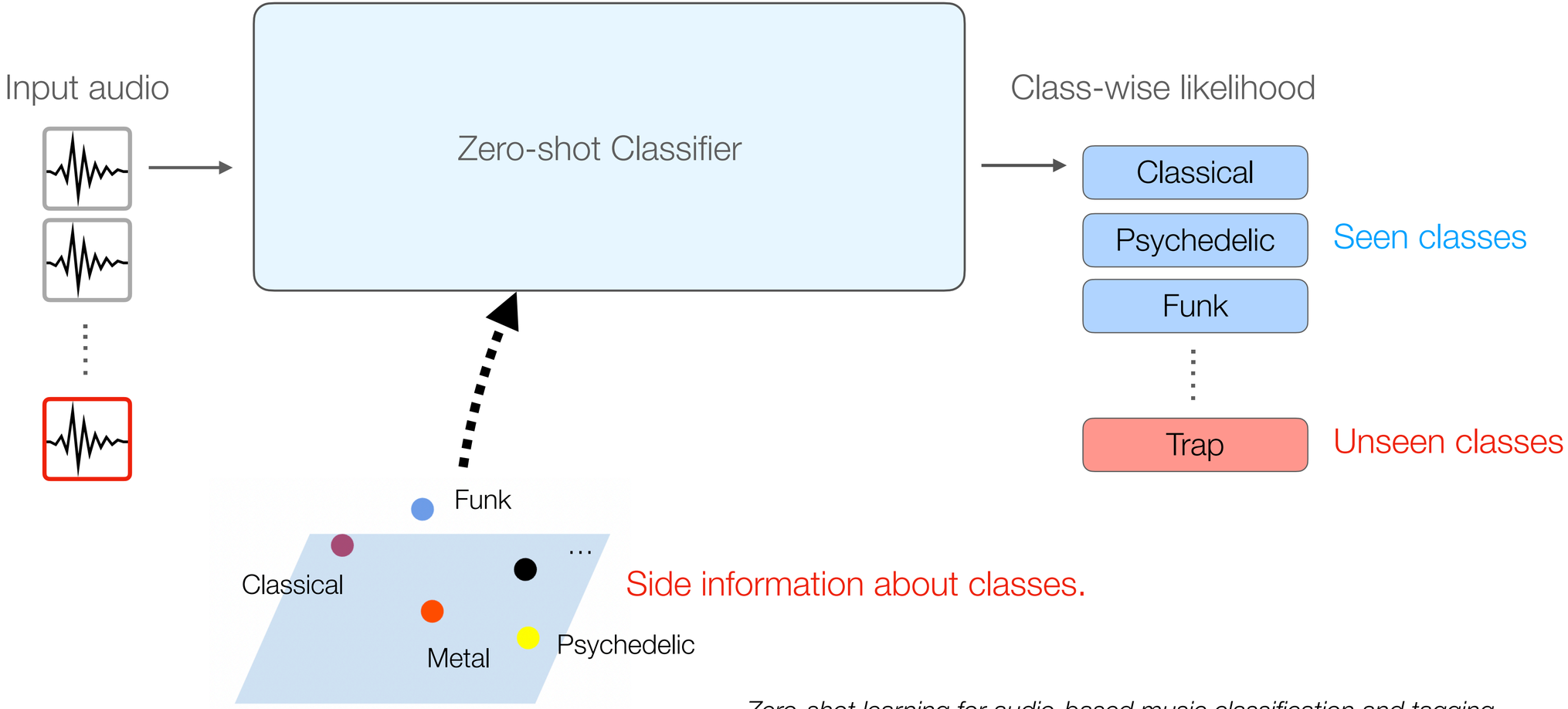
Zero-shot learning for audio-based music classification and tagging
J. Choi*, *J. Lee*, J. Park, and J. Nam ISMIR, 2019*

Zero-Shot Music Tagging



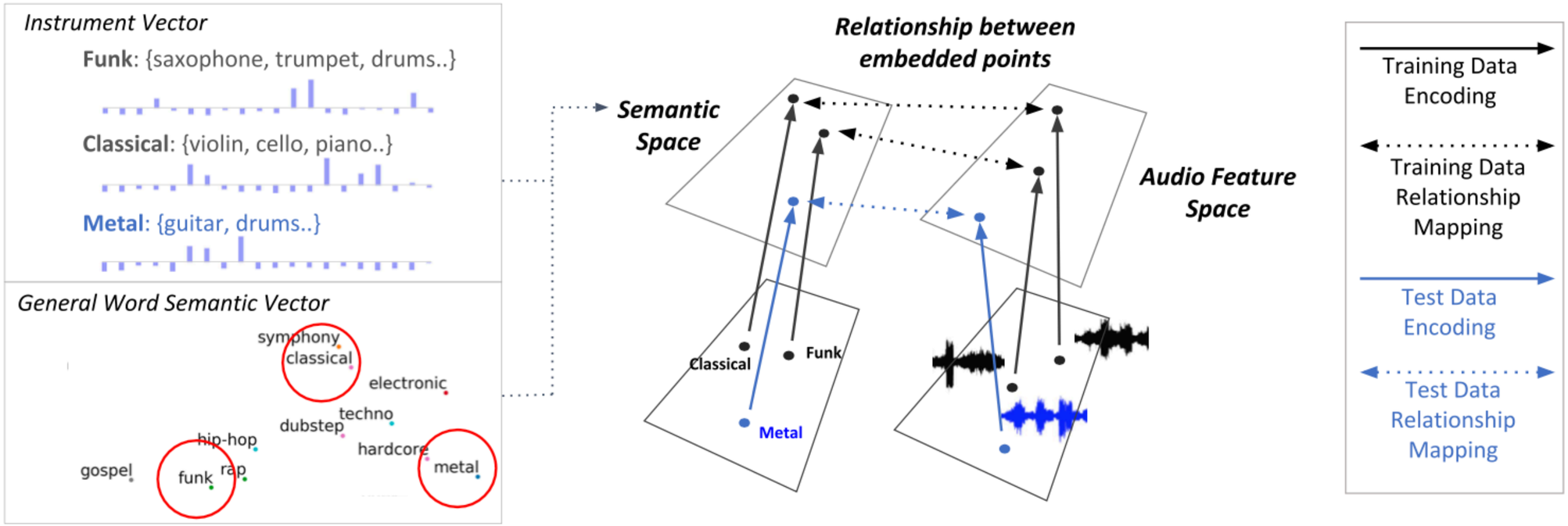
Zero-shot learning for audio-based music classification and tagging
J. Choi*, *J. Lee*, J. Park, and J. Nam ISMIR, 2019*

Zero-Shot Music Tagging



Zero-shot learning for audio-based music classification and tagging
J. Choi*, *J. Lee*, J. Park, and J. Nam ISMIR, 2019*

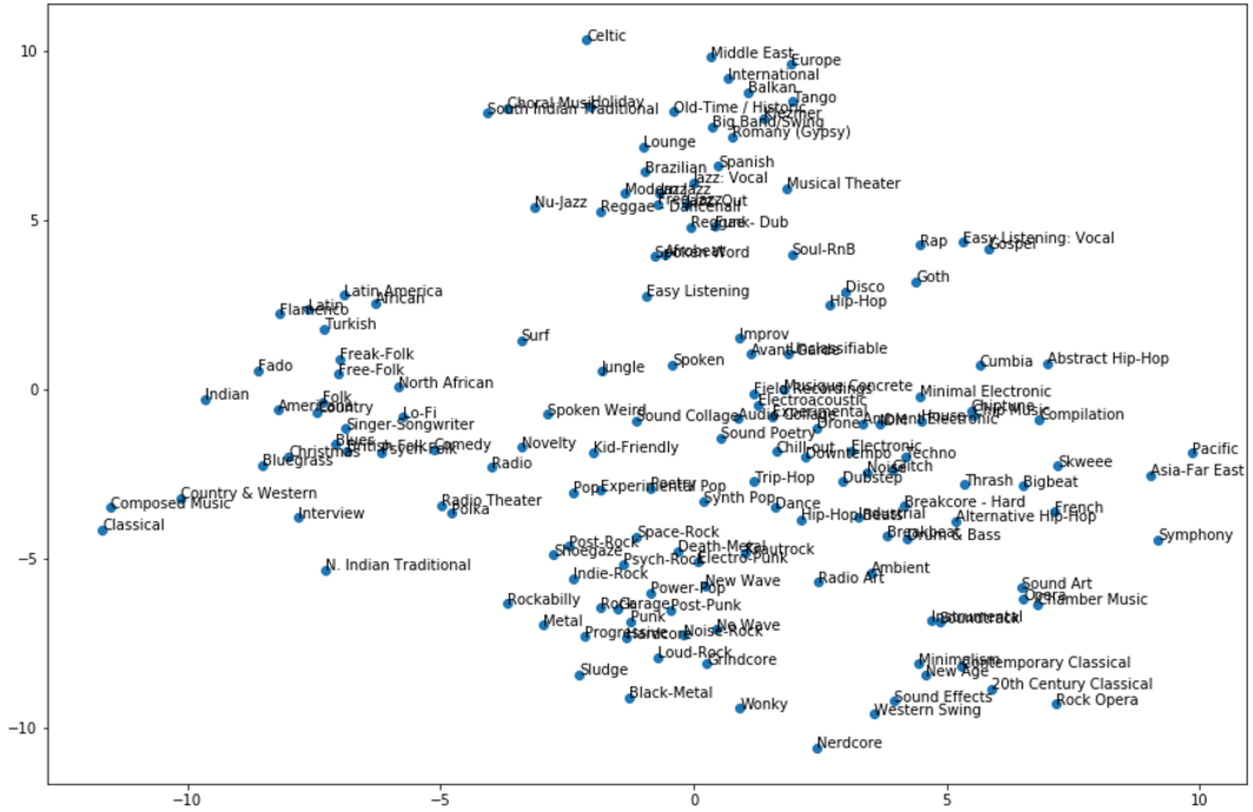
Zero-Shot Music Tagging



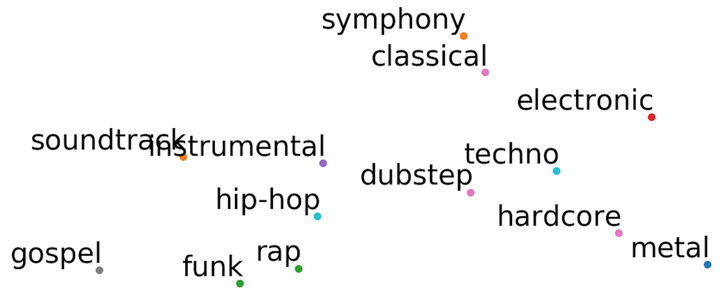
Zero-shot music tagging paradigm using different types of side information on tags.

Zero-shot learning for audio-based music classification and tagging
J. Choi*, *J. Lee*, J. Park, and J. Nam ISMIR, 2019*

Zero-Shot Music Tagging



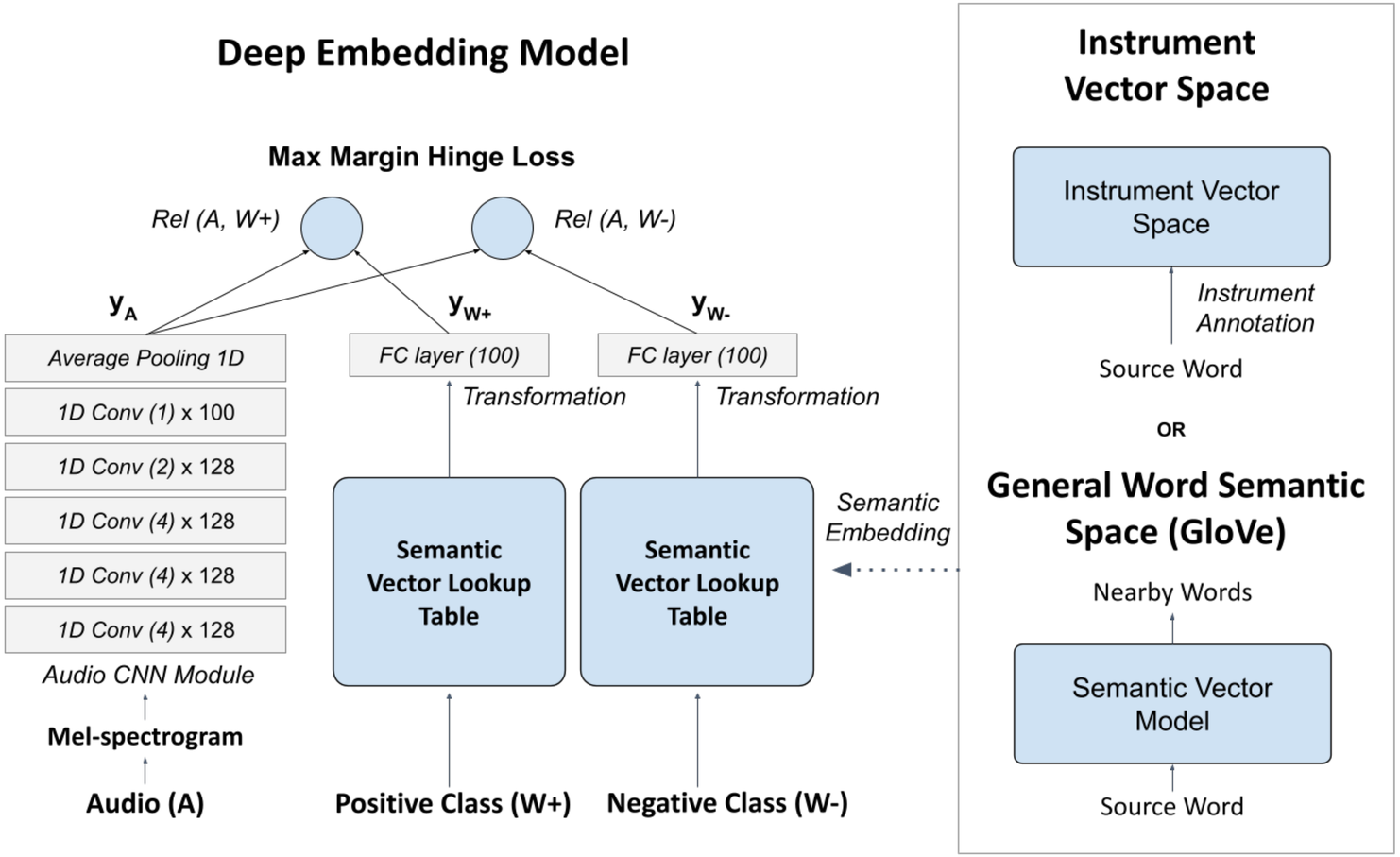
Instrument vectors for 157 genre tags (TSNE visualization).



Examples of GloVe word embedding vectors for musical tags.
(TSNE visulazation)

Zero-shot learning for audio-based music classification and tagging
J. Choi*, *J. Lee**, *J. Park, and J. Nam ISMIR, 2019*

Zero-Shot Music Tagging



Zero-shot learning for audio-based music classification and tagging
J. Choi*, *J. Lee*, J. Park, and J. Nam ISMIR, 2019*

Zero-Shot Music Tagging

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

Table 6. Top 10 auto-tagging results for examples of well-known songs including unseen tags during training.

Examples of zero-shot annotation
(**generalized** zero-shot evaluation).

Zero-shot learning for audio-based music classification and tagging
J. Choi*, *J. Lee*, J. Park, and J. Nam ISMIR, 2019*

Zero-Shot Music Tagging

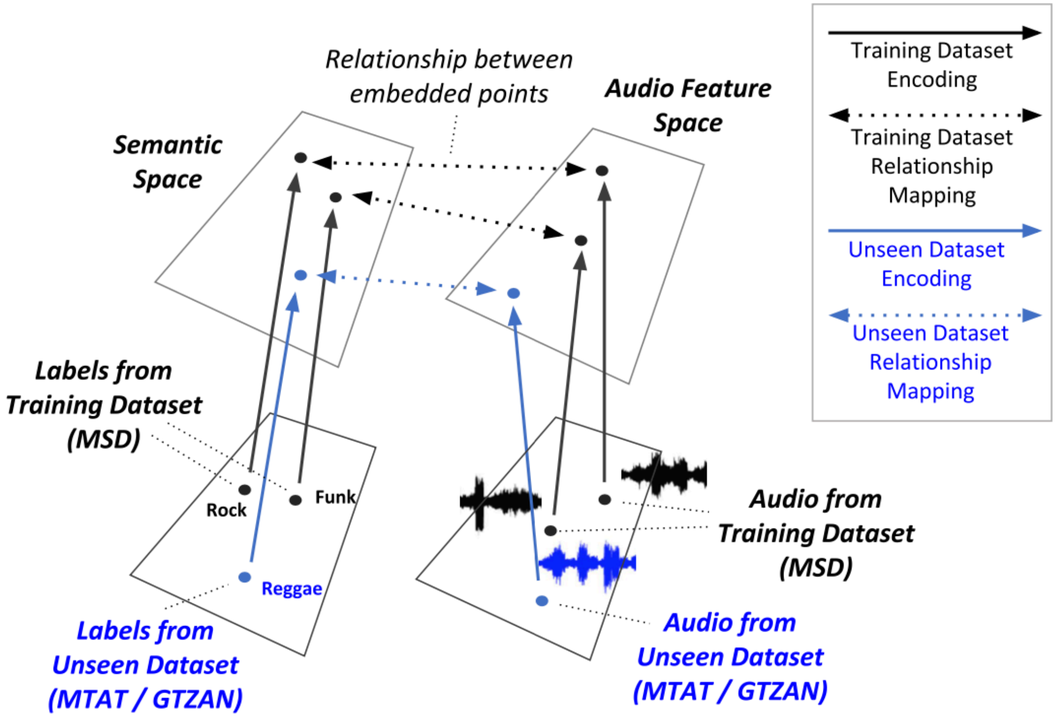


Table 1. Classification results. Tag retrieval AUC is reported for the MTAT dataset and genre annotation accuracy is reported for the GTZAN dataset.

Model	MTAT (Ret. AUC)	GTZAN (Anno. Acc.)
Zero-Shot learning model (MSD-GloVe)	0.7390	0.7310
Classification model (Lee & Nam, 2017)	0.9021	0.7203

Evaluating the generalization power of zero-shot model on multiple music tagging datasets.

Zero-shot Learning and Knowledge Transfer in Music Classification and Tagging
***J. Choi**, J. Lee, J. Park, and J. Nam, ICML Workshop on ML4MD, 2019*

Zero-Shot Music Tagging

Table 2. Comparison of label predictions for MTAT tracks on the original MTAT tags and 1,126 MSD Last.fm tags.

Track ID	Ground Truth (MTAT)	Predictions out of MTAT tags (threshold = 0.6)	Predictions out of 1,126 MSD tags (threshold = 0.7)
8900	slow ambient synth	ambient piano electronic	instrumental chillout / ambient soundtrack / jazz / piano relaxing
11308	slow / techno electronic / beat synth / weird	electronic techno dance	electronic / techno electro / electronica downtempo / dub bassline / chillout experimental

Table 3. Comparison of label predictions for GTZAN tracks on the original GTZAN genres and 1,126 MSD Last.fm tags.

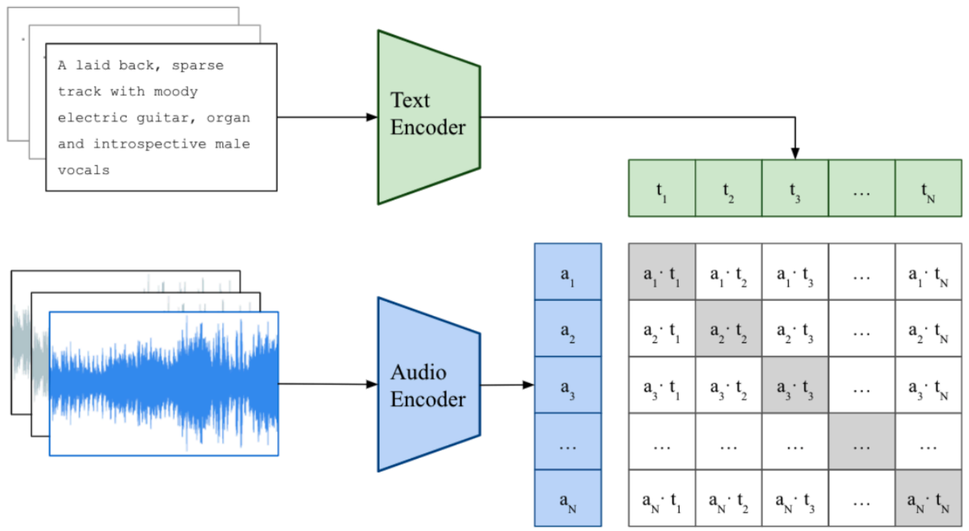
Track ID	Ground Truth (GTZAN)	Top 1 Prediction out of GTZAN genres	Predictions out of 1,126 MSD tags (threshold = 0.6)
560	jazz	jazz	jazz / blues instrumental swing / smoothjazz piano
600	reggae	reggae	reggae / dub / ska roots / dancehall rootsreggae

Case studies from zero-shot transfer learning experiments on different datasets.

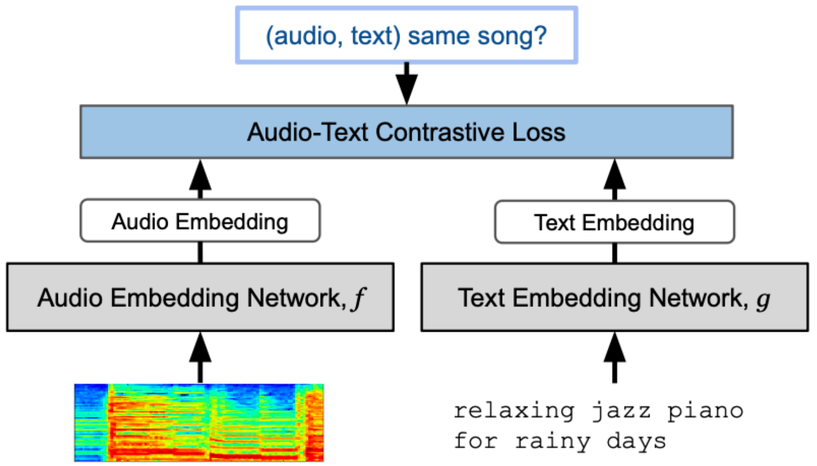
Zero-shot Learning and Knowledge Transfer in Music Classification and Tagging
***J. Choi**, J. Lee, J. Park, and J. Nam, ICML Workshop on ML4MD, 2019*

Recent Advances in Zero-shot Music Tagging

Contrastive Audio-Language Learning for Music
Manco et al., ISMIR 2022



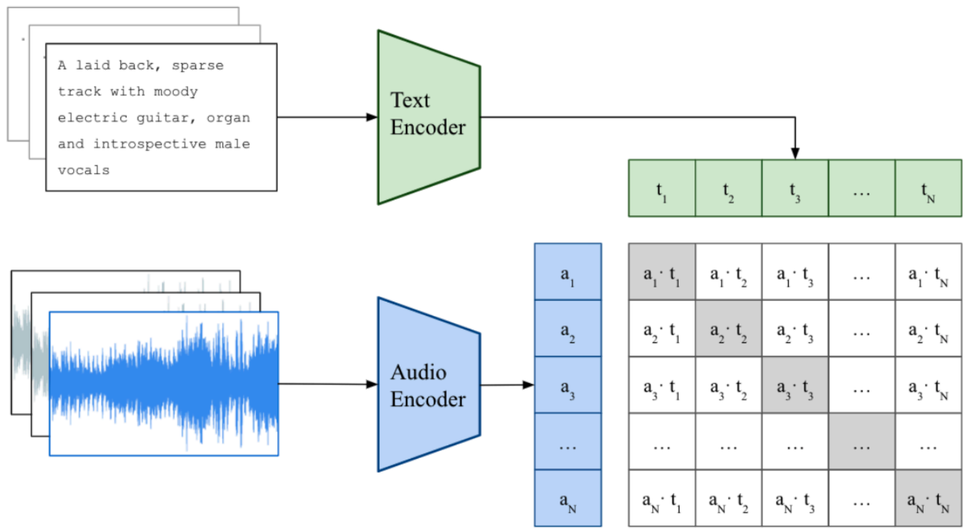
MuLan: A Joint Embedding of Music Audio and Natural Language
Huang et al., ISMIR 2022



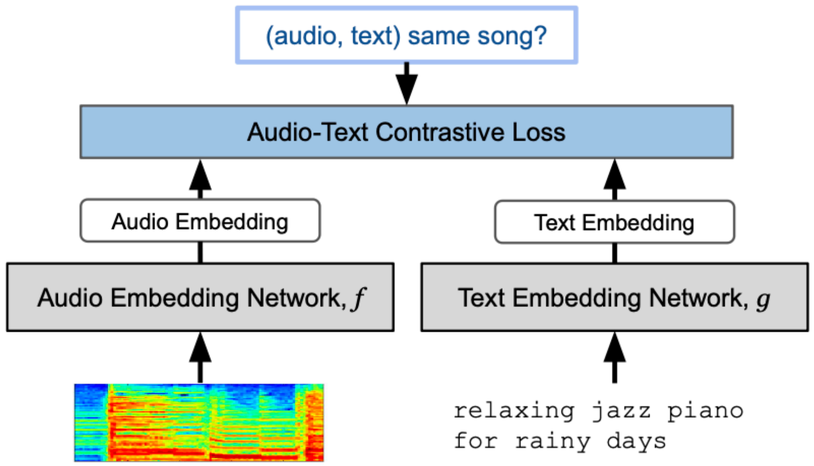
Type	Examples
Short-form (SF)	tags like genre, mood, instrument, artist name, song title, album name
Long-form (LF)	'Hip-hop features rap with an electronic backing.' 'The melody is so nostalgic and unforgettable.'
Playlist (PL)	'Feel-good mandopop indie', 'Latin workout' 'Salsa for broken hearts', 'Piano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

Recent Advances in Zero-shot Music Tagging

Contrastive Audio-Language Learning for Music
Manco et al., ISMIR 2022



MuLan: A Joint Embedding of Music Audio and Natural Language
Huang et al., ISMIR 2022



Both trained the model using large-scaled text and music pair datasets.
Zero-shot transfer evaluation results are reported on separate tagging datasets.

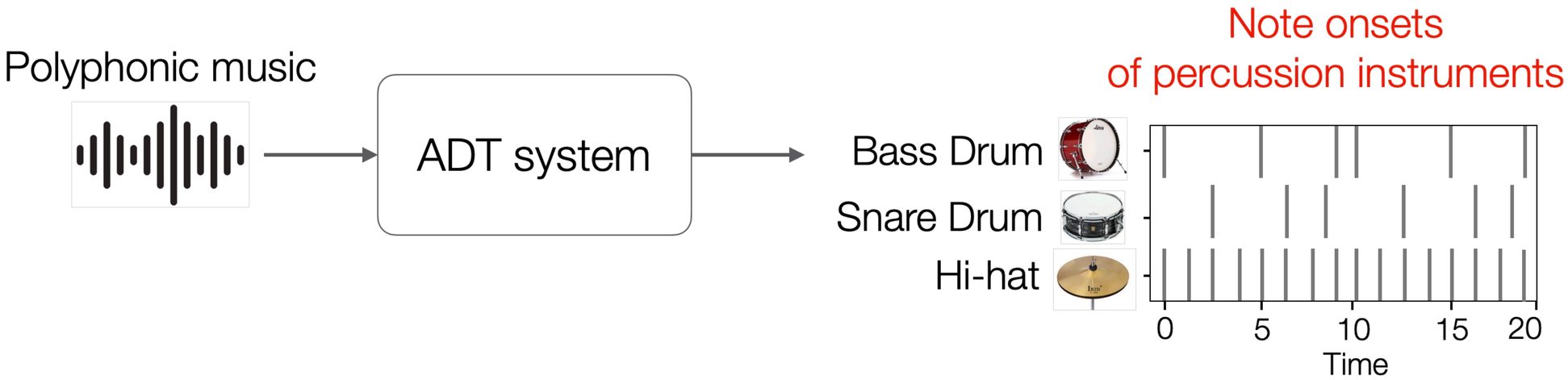
Recent Works in MIR Leveraging FSL/ZSL

Classification

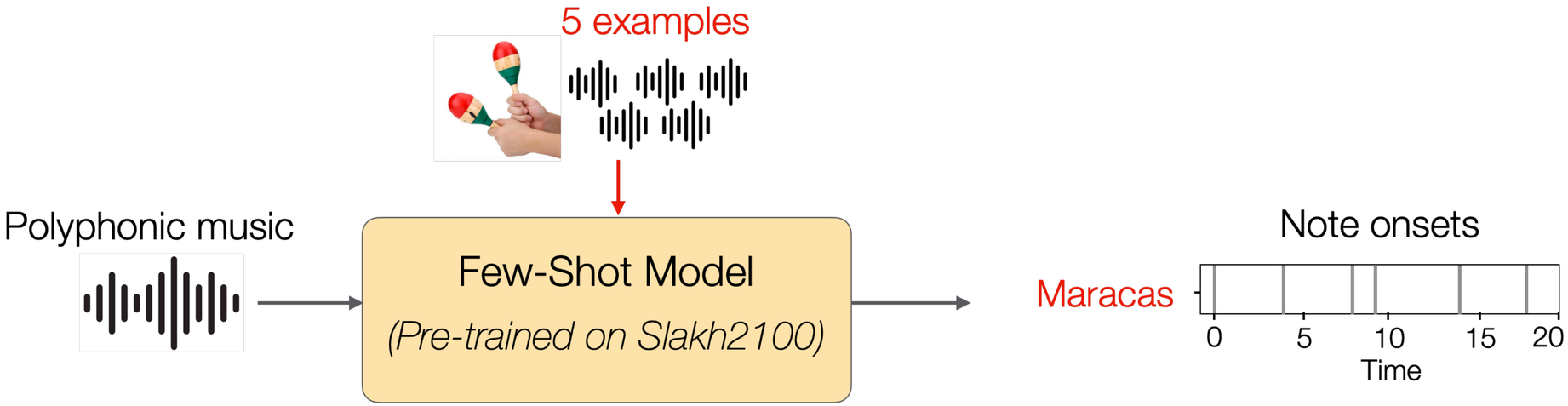
Transcription

Source Separation

Automatic Drum Transcription (ADT)

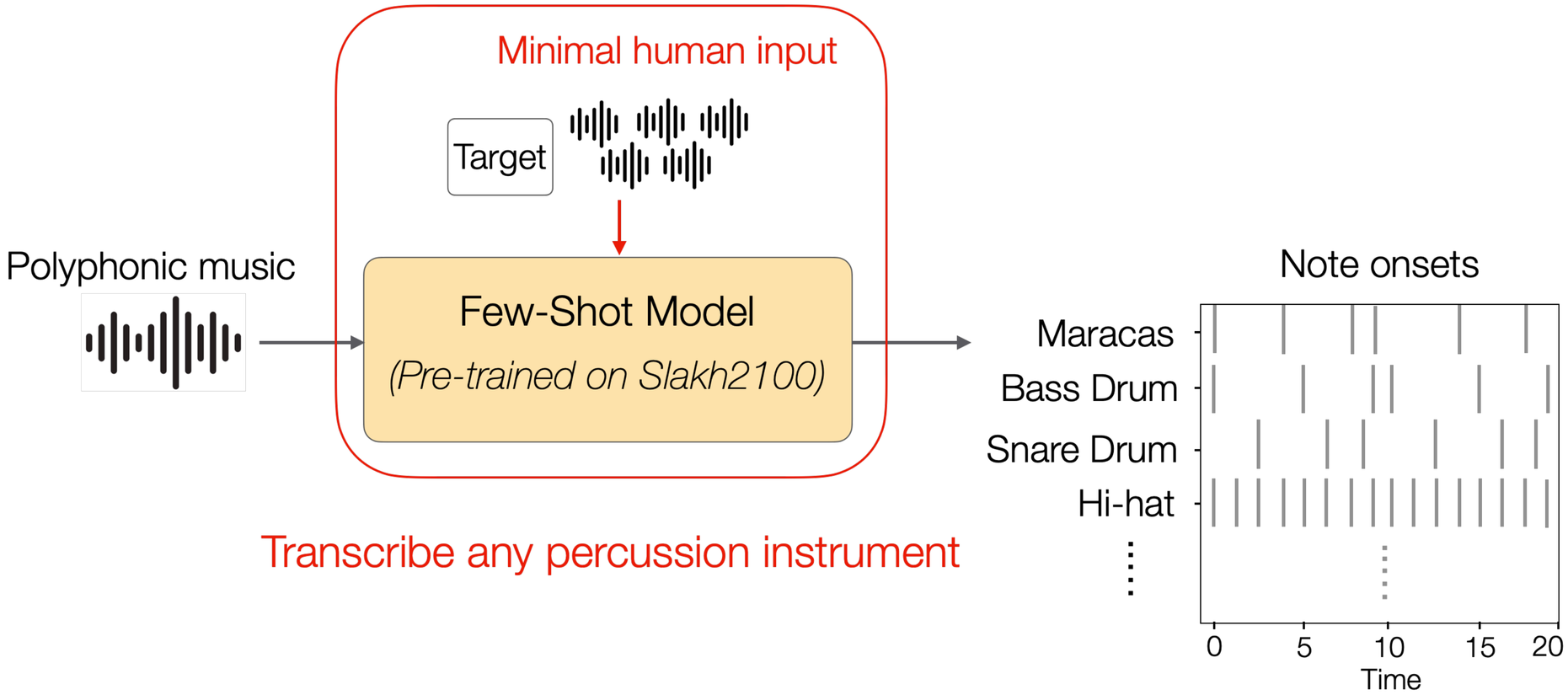


New Paradigm: Few-Shot Drum Transcription



Few-Shot Drum Transcription in Polyphonic Music
Y. Wang, *J. Salamon, M. Cartwright, N. J. Bryan, and J. P. Bello, ISMIR, 2020*

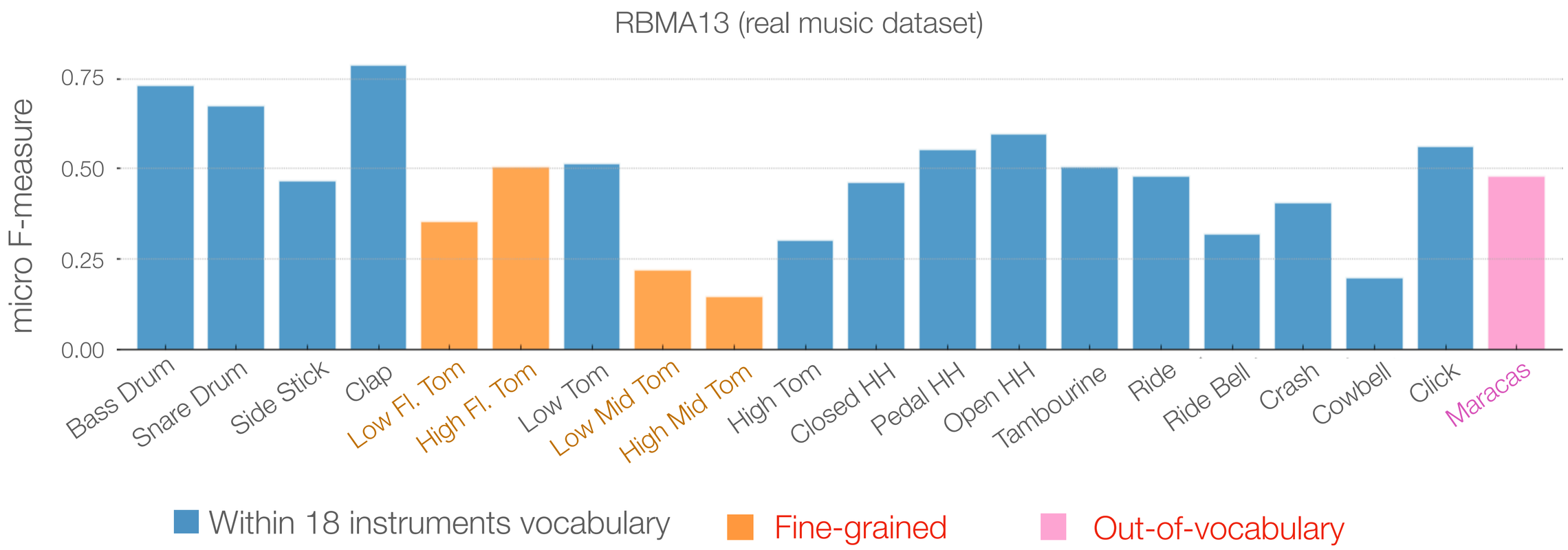
New Paradigm: Few-Shot Drum Transcription



Transcribe any percussion instrument

Few-Shot Drum Transcription in Polyphonic Music
***Y. Wang**, J. Salamon, M. Cartwright, N. J. Bryan, and J. P. Bello, ISMIR, 2020*

Expanded Vocabulary



Few-Shot Drum Transcription in Polyphonic Music
***Y. Wang**, J. Salamon, M. Cartwright, N. J. Bryan, and J. P. Bello, ISMIR, 2020*

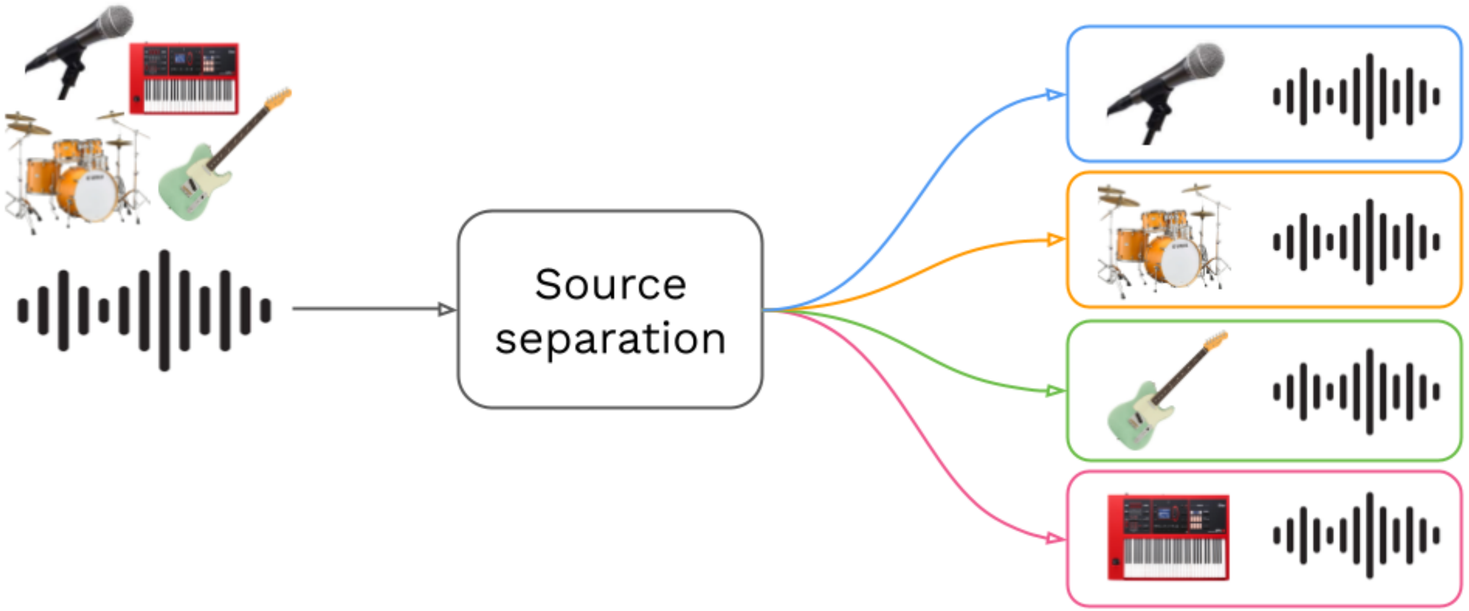
Recent Works in MIR Leveraging FSL/ZSL

Classification

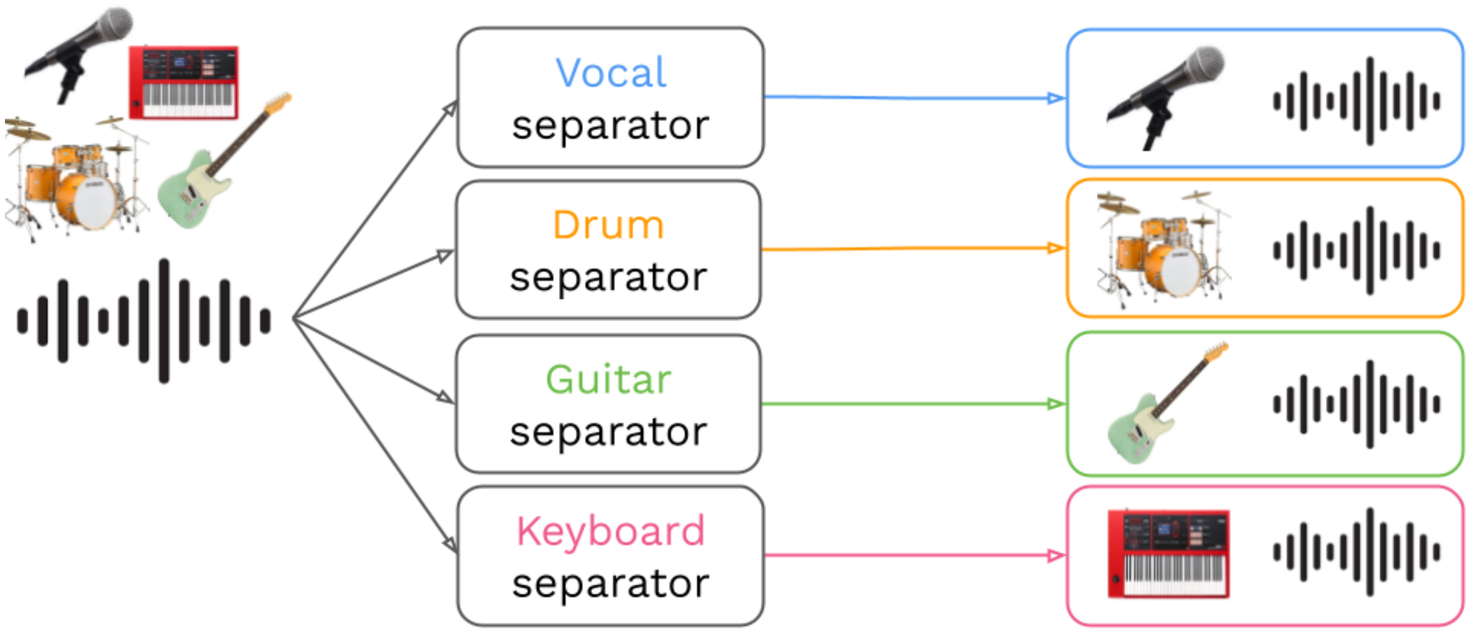
Transcription

Source Separation

Musical Source Separation (MSS)

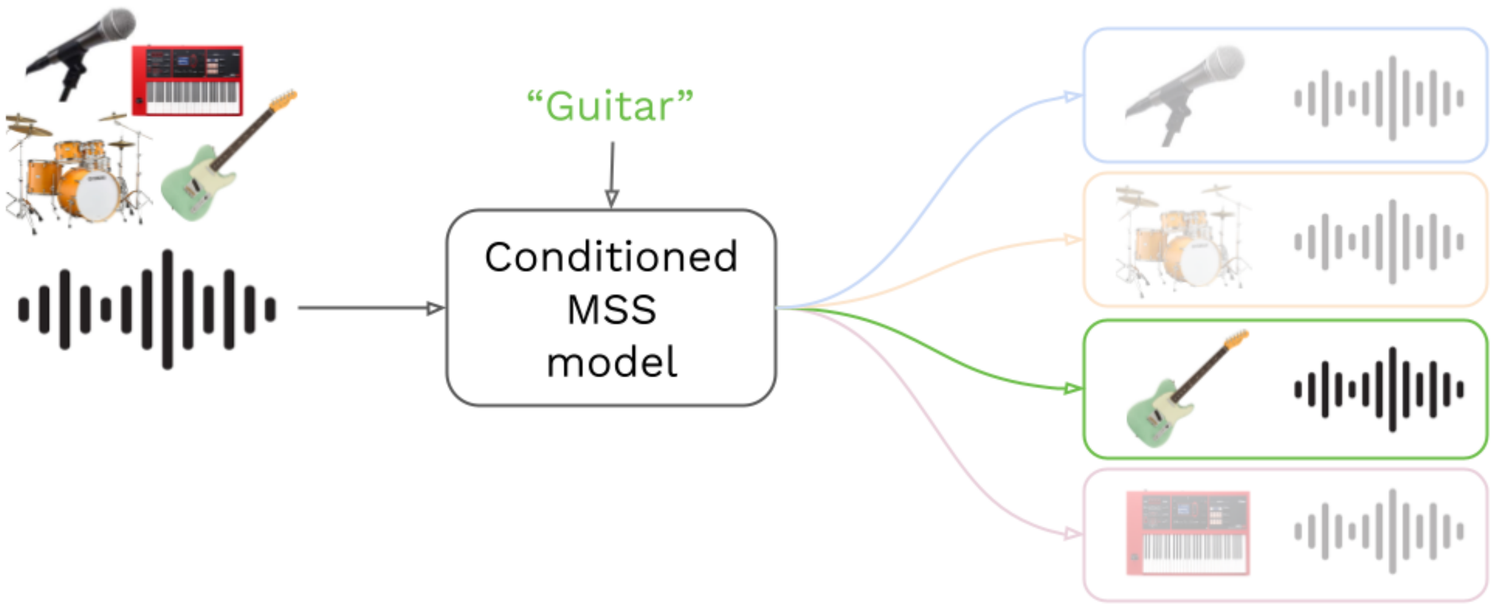


Multiple Single-Instrument Models

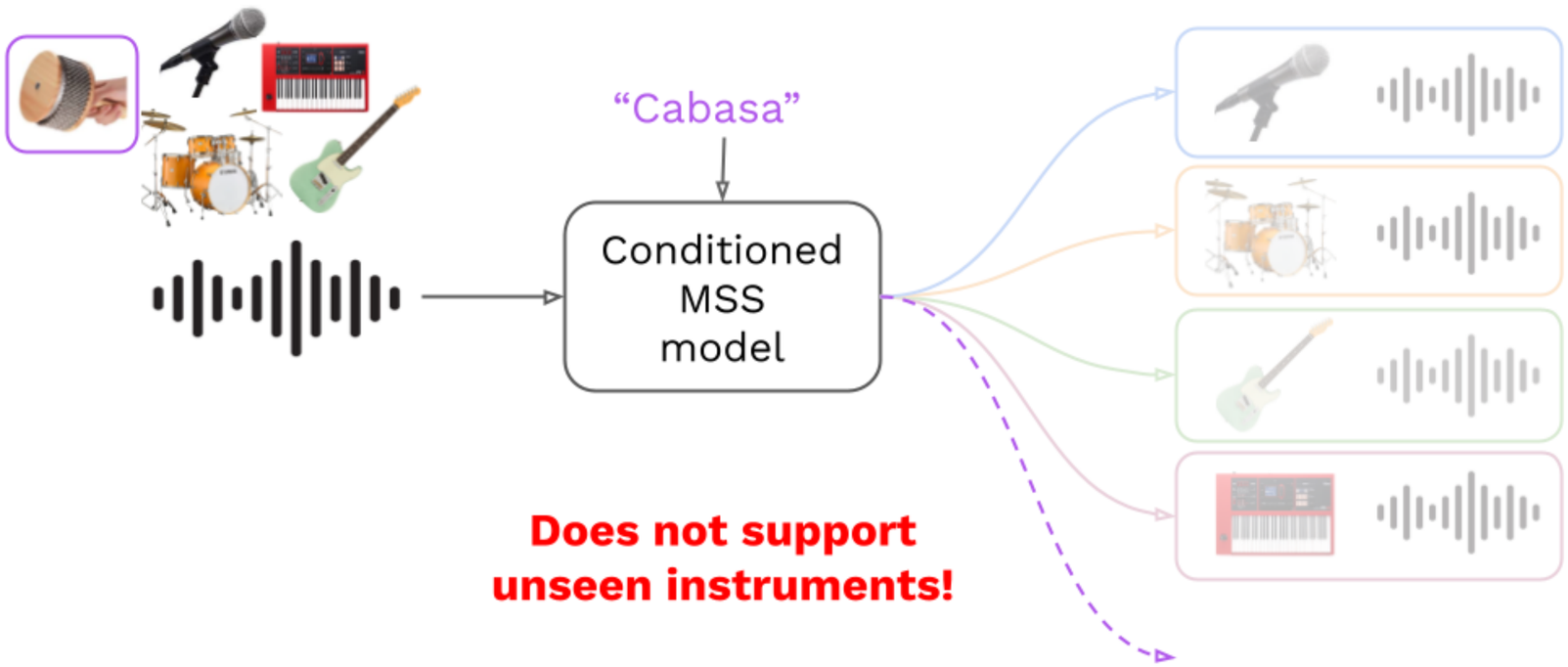


Scaling issues!

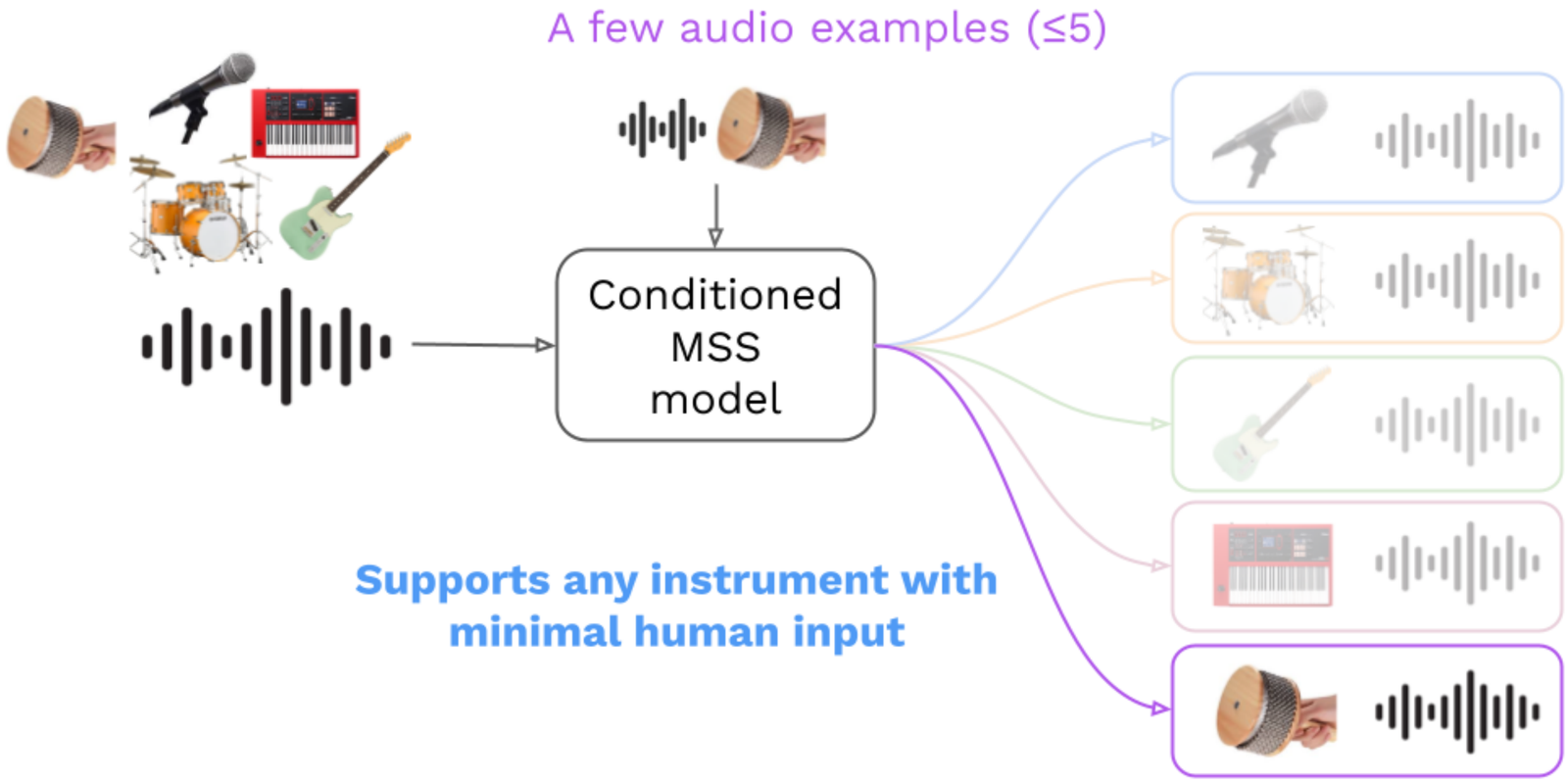
One Generic Model + Conditioning



One Generic Model + Conditioning

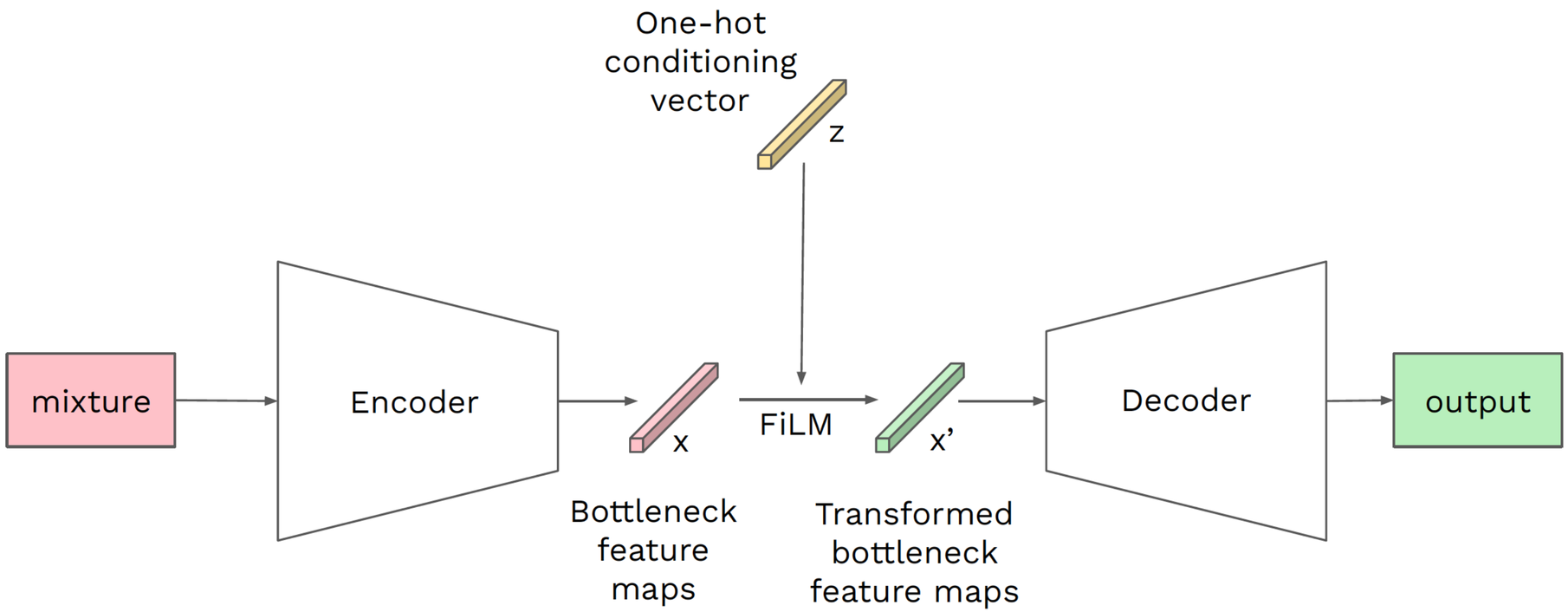


Few-Shot MSS



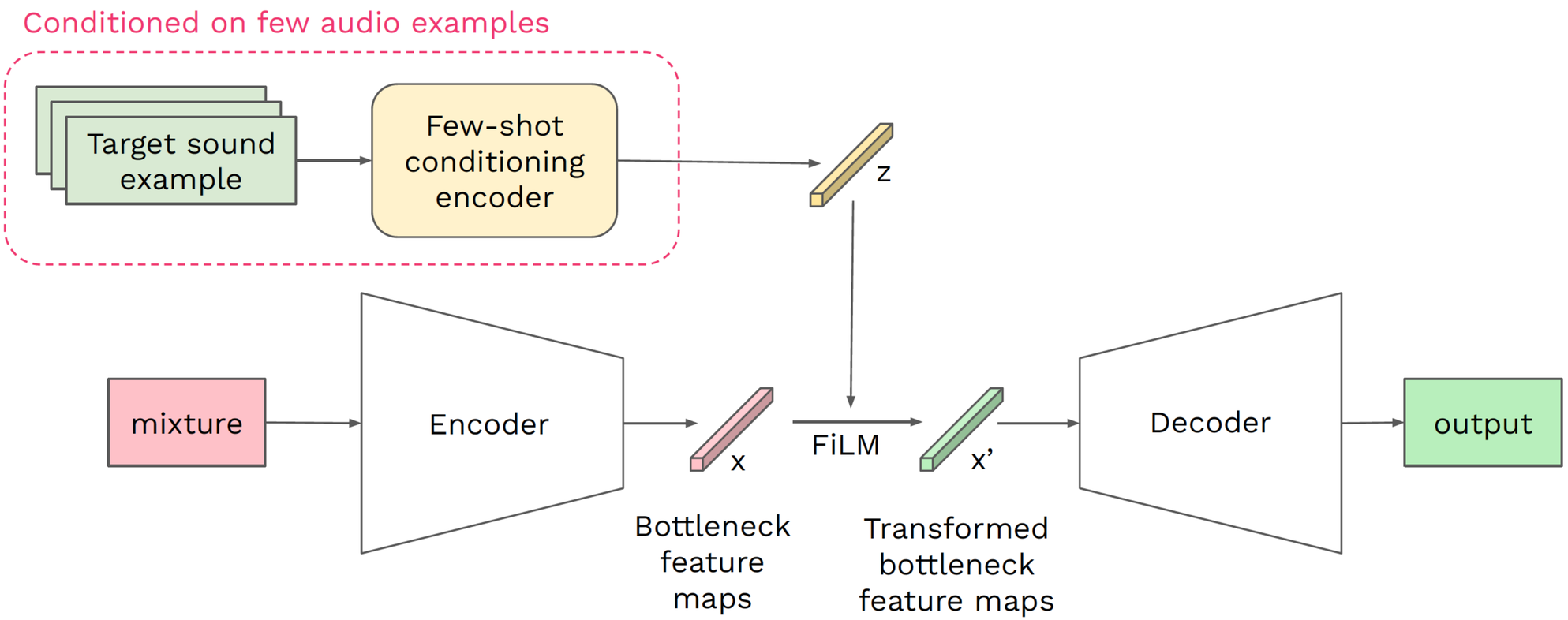
Few-shot Musical Source Separation
***Y. Wang**, D. Stoller, R. M. Bittner, and J. P. Bello ICASSP, 2022*

Few-Shot Musical Source Separation



Conditioned-u-net: Introducing a control mechanism in the u-net for multiple source separations
Gabriel Meseguer-Brocal and Geoffroy Peeters ISMIR 2019

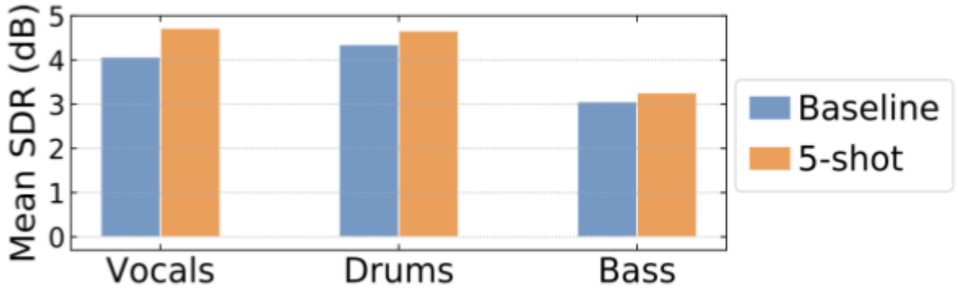
Few-Shot Musical Source Separation



Few-shot Musical Source Separation
***Y. Wang**, D. Stoller, R. M. Bittner, and J. P. Bello ICASSP, 2022*

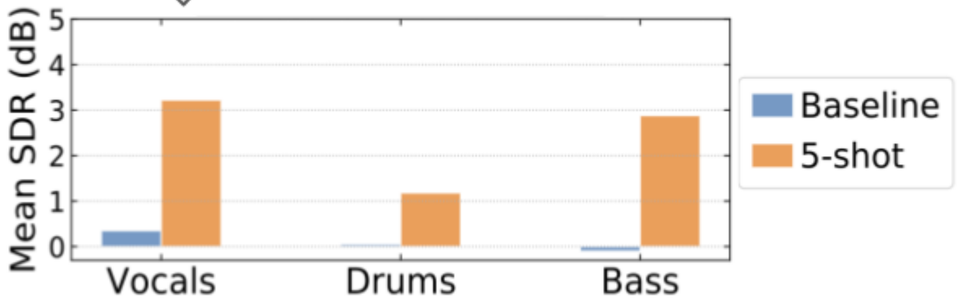
Performance on MUSDB18

Seen instruments
Unseen dataset



Remove *vocals, drums, bass* examples from training data

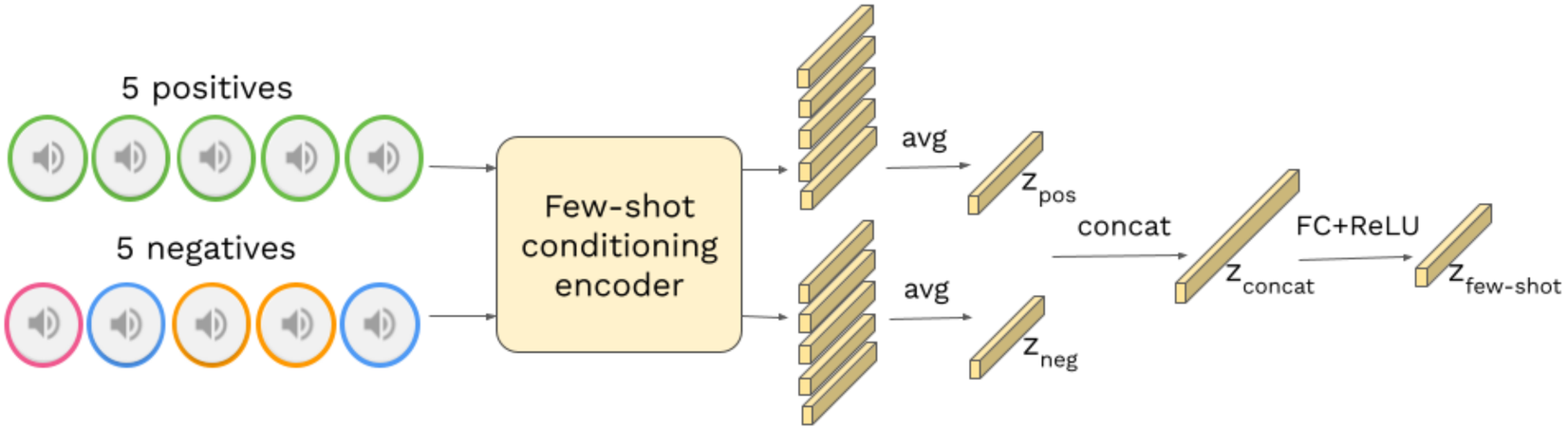
Unseen instruments
Unseen dataset



➢ Few-shot model outperforms the baseline on both seen and unseen instruments.

Few-shot Musical Source Separation
***Y. Wang**, D. Stoller, R. M. Bittner, and J. P. Bello ICASSP, 2022*

Conditioning on Negative Examples



Few-shot Musical Source Separation
***Y. Wang**, D. Stoller, R. M. Bittner, and J. P. Bello ICASSP, 2022*

Summary of Part 3: Recent Advances

Classification

- K-way N-shot sound classification
- Hierarchical ProtoNet
- Few-shot SED

Transcription

- Few-shot ADT in polyphonic music

Source Separation

- Few-shot source separation

Tutorial Outline

- Part 1: Introduction & Foundations
- Part 2: Coding Examples
- Part 3: Recent Advances
- Closing Remarks

Remaining Challenges & Future Directions

1. Train FSL/ZSL models with less labeled data
    - Unsupervised FSL, semi-supervised FSL
2. Leverage existing pre-trained models
    - Few-shot continual learning
3. New FSL/ZSL methods
4. Apply FSL/ZSL to more MIR tasks
    - Recommendation, generation, problems that you are working on?
5. User study
    - Support set selection, interaction flow

Summary of this Tutorial

1. What are FSL/ZSL and their strengths

2. Code that runs training & eval pipelines for few-shot/zero-shot classification models

3. What have been done/what to be done in MIR with FSL/ZSL

JupyterBook: music-fsl-zsl.github.io/tutorial/

Github Repo: github.com/music-fsl-zsl/tutorial

Code: github.com/music-fsl-zsl/music_fsl