PAT 498/598 (Winter 2025)

Music & AI

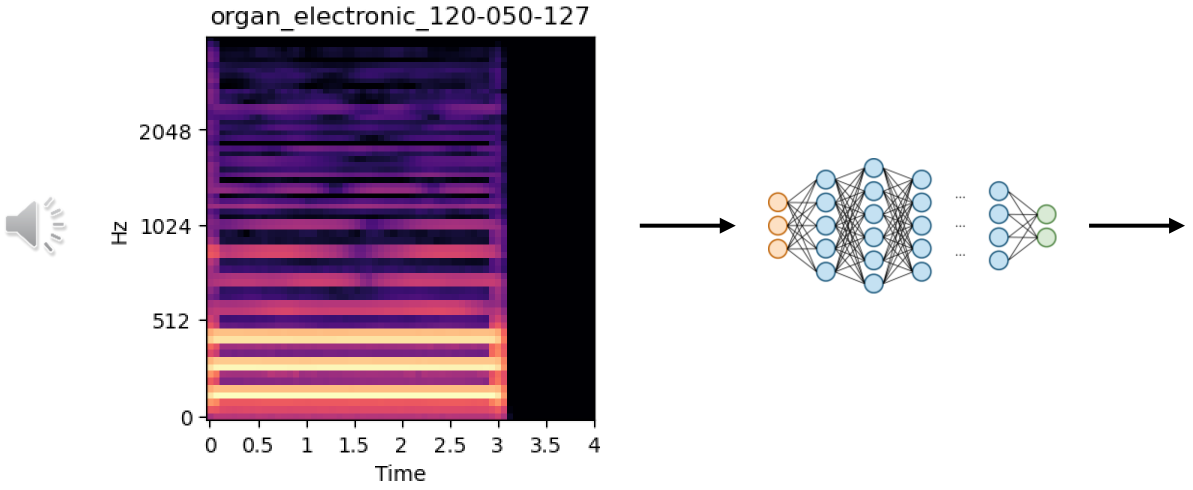
Lecture 11: Music Classification

Instructor: Hao-Wen Dong

SCHOOL OF MUSIC, THEATRE & DANCE
PERFORMING ARTS TECHNOLOGY
UNIVERSITY OF MICHIGAN

Homework 3: Musical Note Classification using CNNs

- Train a CNN that can classify audio files into their **instrument families**
  - **Input**: 64x64 mel spectrogram
  - **Output**: 11 instrument classes
  - Using the **NSynth** dataset (Engel et al., 2017)

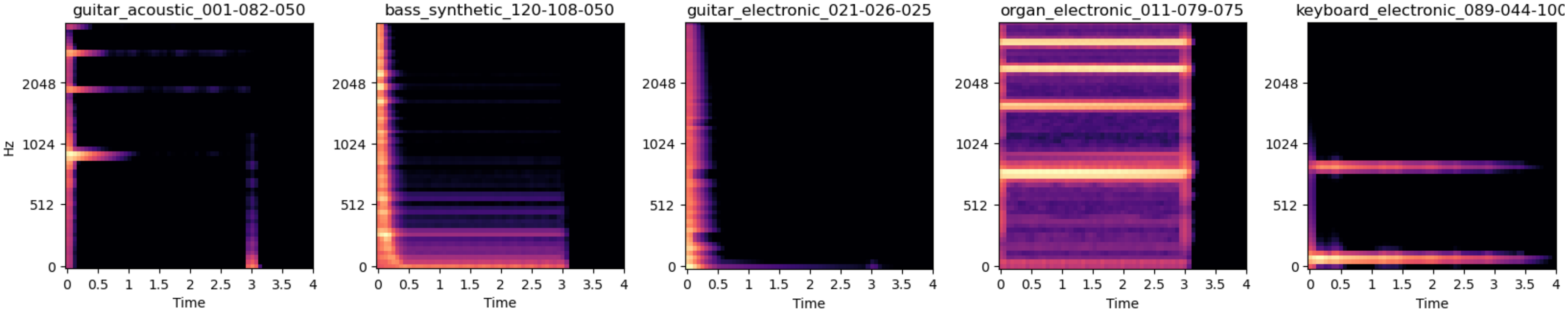


Bass
Brass
Flute
Guitar
Keyboard
Mallet
Organ
Reed
String
Synth Lead
Vocal

Jesse Engel, Cinjon Resnick, Adam Roberts, Sander Dieleman, Douglas Eck, Karen Simonyan, and Mohammad Norouzi, "Neural Audio Synthesis of Musical Notes with WaveNet Autoencoders," *ICML*, 2017.

NSynth Dataset

- A collection of 305,979 **single-shot musical notes** (Engel et al., 2017)
  - Produced from 1,006 **commercial sample libraries**
  - With different **MIDI pitches** (21–108) and **velocities** (25, 50, 75, 100, 127)



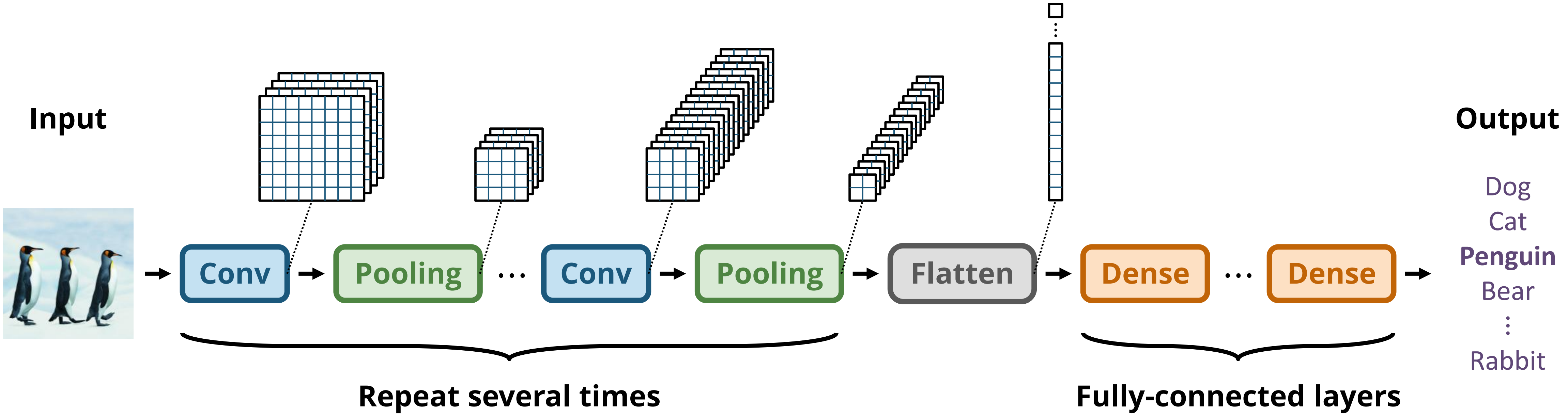
Jesse Engel, Cinjon Resnick, Adam Roberts, Sander Dieleman, Douglas Eck, Karen Simonyan, and Mohammad Norouzi, "Neural Audio Synthesis of Musical Notes with WaveNet Autoencoders," *ICML*, 2017.

Homework 3: Musical Note Classification using CNNs

- Instructions will be released on Gradescope
- Due at **11:59pm ET** on **February 17**
- Late submissions: **1 point deducted per day**

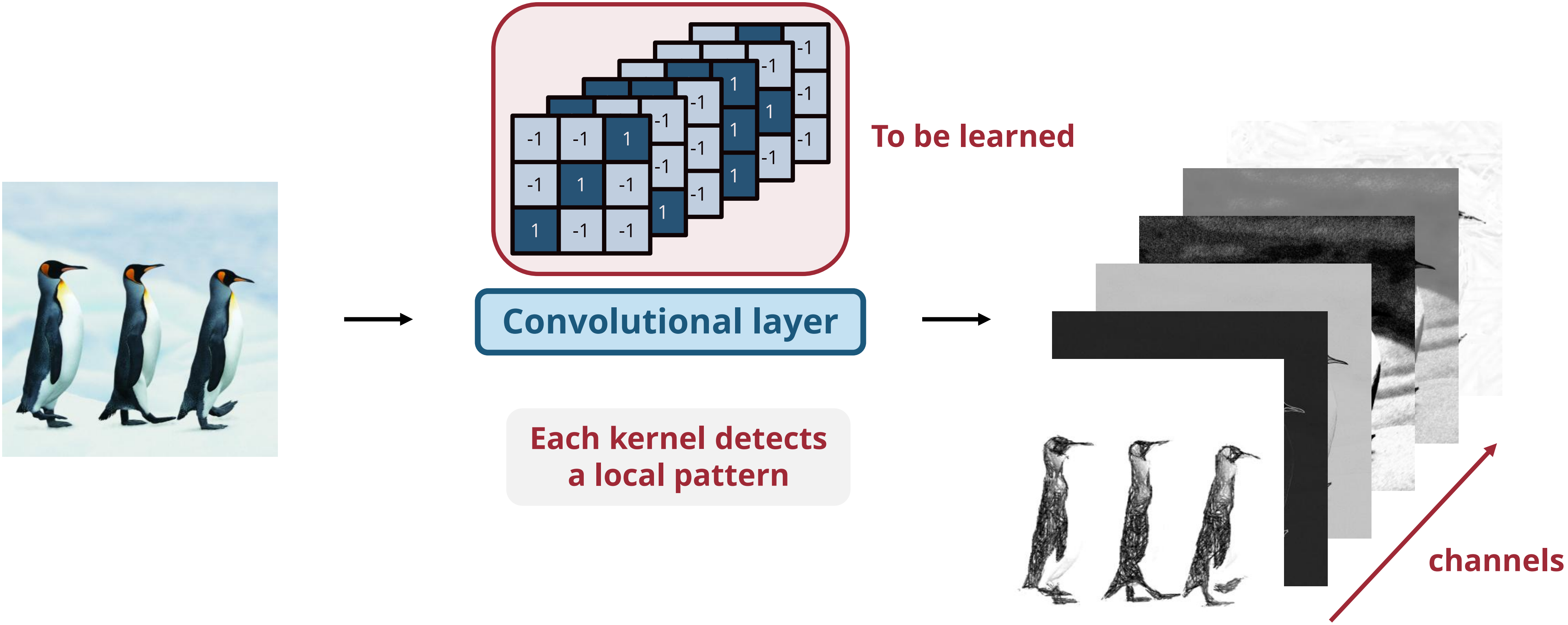
(Recap) Reusable Pattern Detectors

(Recap) Convolutional Neural Network (CNNs)



(Recap) Convolutional Layer

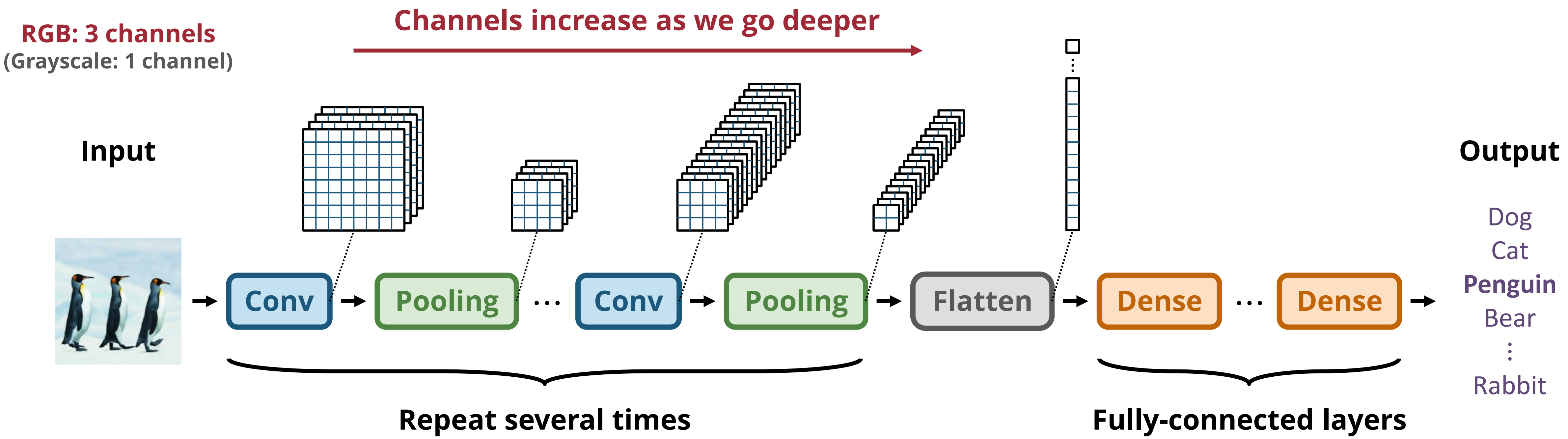
- A convolutional layer consists of many **learnable kernels** (channels)



(Recap) Benefits of CNNs

- Learn **local patterns**
- **Invariant to shifts**
  - Also called *translational invariance*
- **Reuse the learned filters** across
  - Different parts of the image
  - Across different images
- **Higher parameter-efficiency** against fully-connected neural network

(Recap) Convolutional Neural Network (CNNs)



(Recap) A Real Example

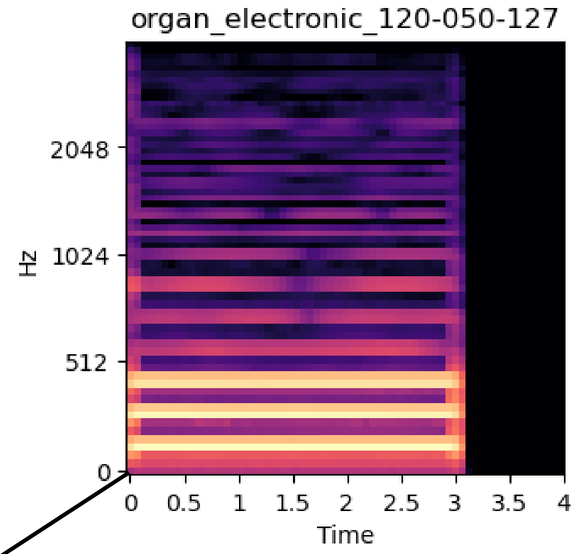
Input channels

Output channels

Kernel size

```
class CNN(nn.Module):
    """A basic convolutional neural network."""
    def __init__(self):
        super().__init__()
        self.conv1 = nn.Conv2d(1, 16, 3, padding="same")
        self.conv2 = nn.Conv2d(16, 32, 3, padding="same")
        self.conv3 = nn.Conv2d(32, 64, 3, padding="same")
        self.conv4 = nn.Conv2d(64, 128, 3, padding="same")
        self.pool = nn.MaxPool2d(2, 2)
        self.fc = nn.Linear(128 * 4 * 4, n_classes)

    def forward(self, x):
        x = self.pool(F.relu(self.conv1(x)))
        x = self.pool(F.relu(self.conv2(x)))
        x = self.pool(F.relu(self.conv3(x)))
        x = self.pool(F.relu(self.conv4(x)))
        x = torch.flatten(x, 1)
        x = self.fc(x)
        return x
```



Input: 1 x 64 x 64

16 x 32 x 32 = 16384

32 x 16 x 16 = 8192

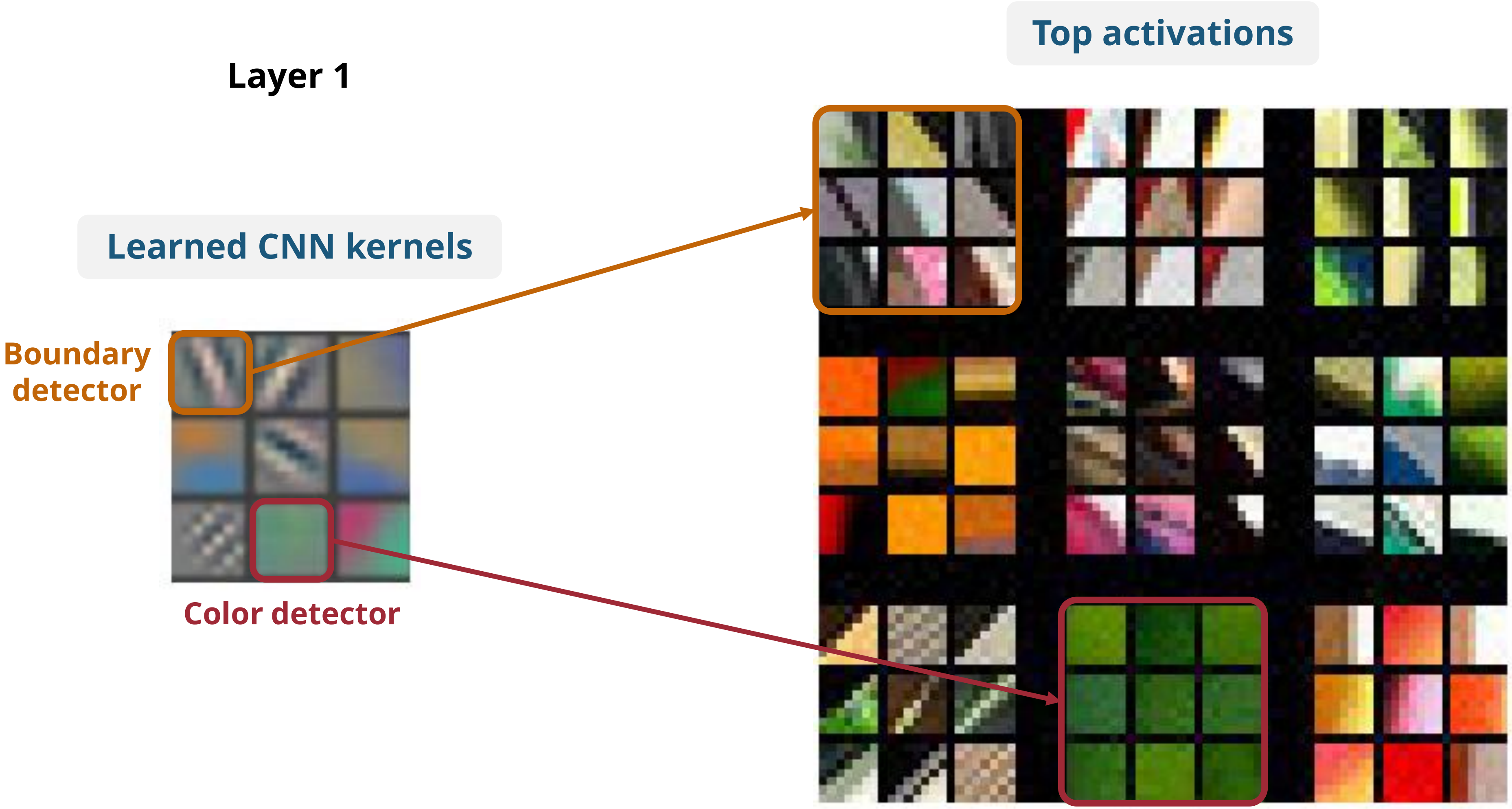
64 x 8 x 8 = 4096

128 x 4 x 4 = 2048

More channels, lower resolution as we go deeper

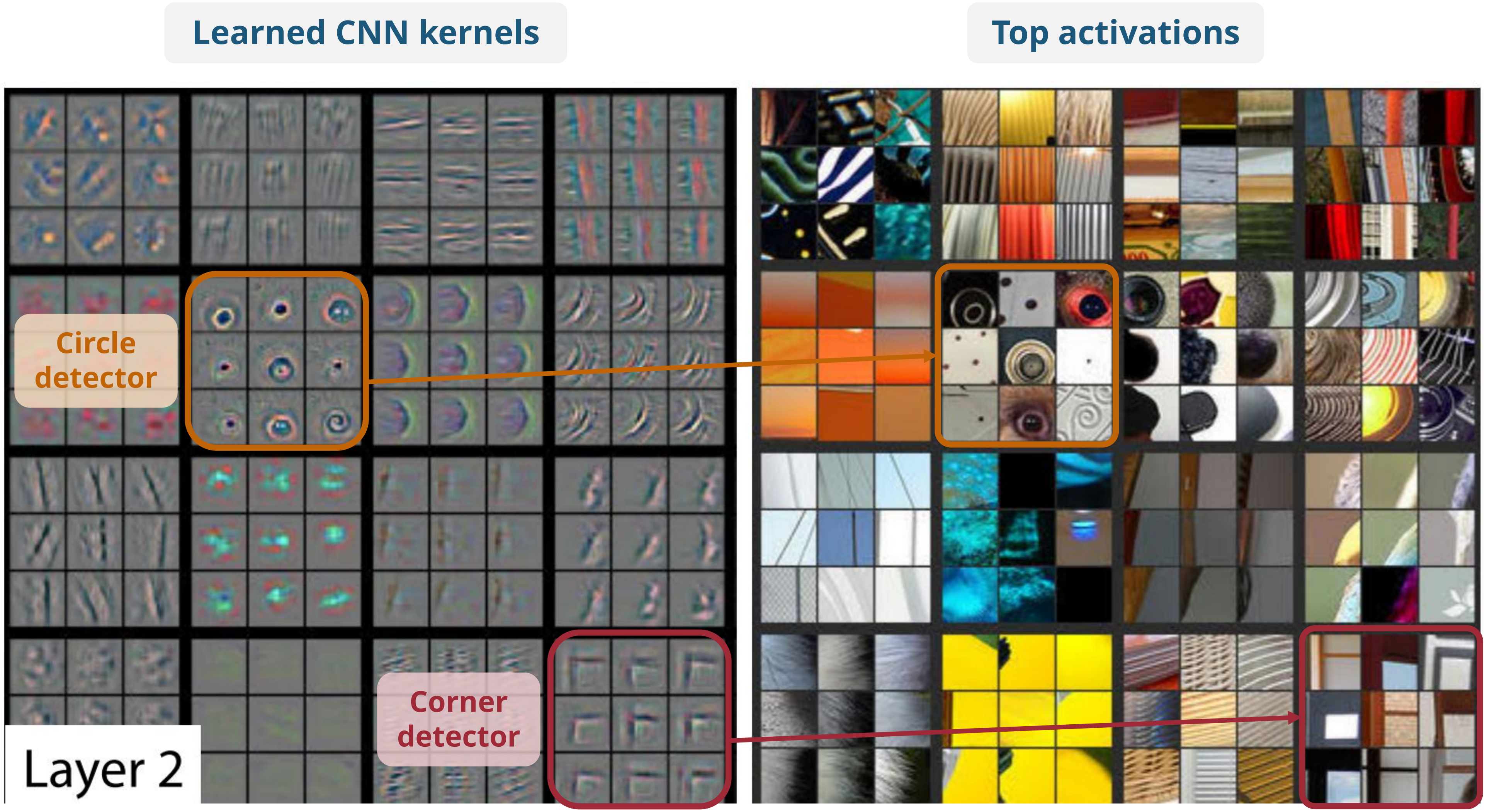
Total number of features decrease as we go deeper

(Recap) Learned CNN Kernels in a Trained AlexNet



Matthew D. Zeiler and Rob Fergus, "Visualizing and Understanding Convolutional Networks," *ECCV*, 2014.

(Recap) Learned CNN Kernels in a Trained AlexNet

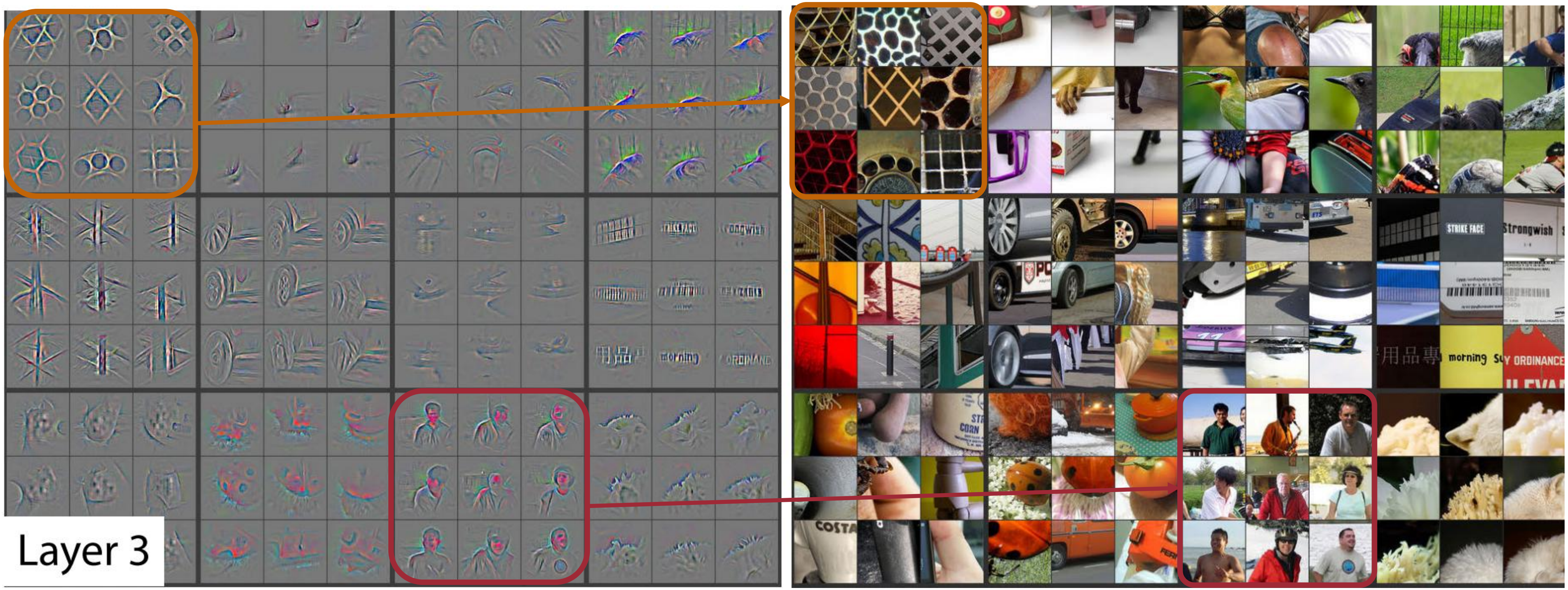


Matthew D. Zeiler and Rob Fergus, "Visualizing and Understanding Convolutional Networks," *ECCV*, 2014.

(Recap) Learned CNN Kernels in a Trained AlexNet

Learned CNN kernels **Top activations**

Grid detector

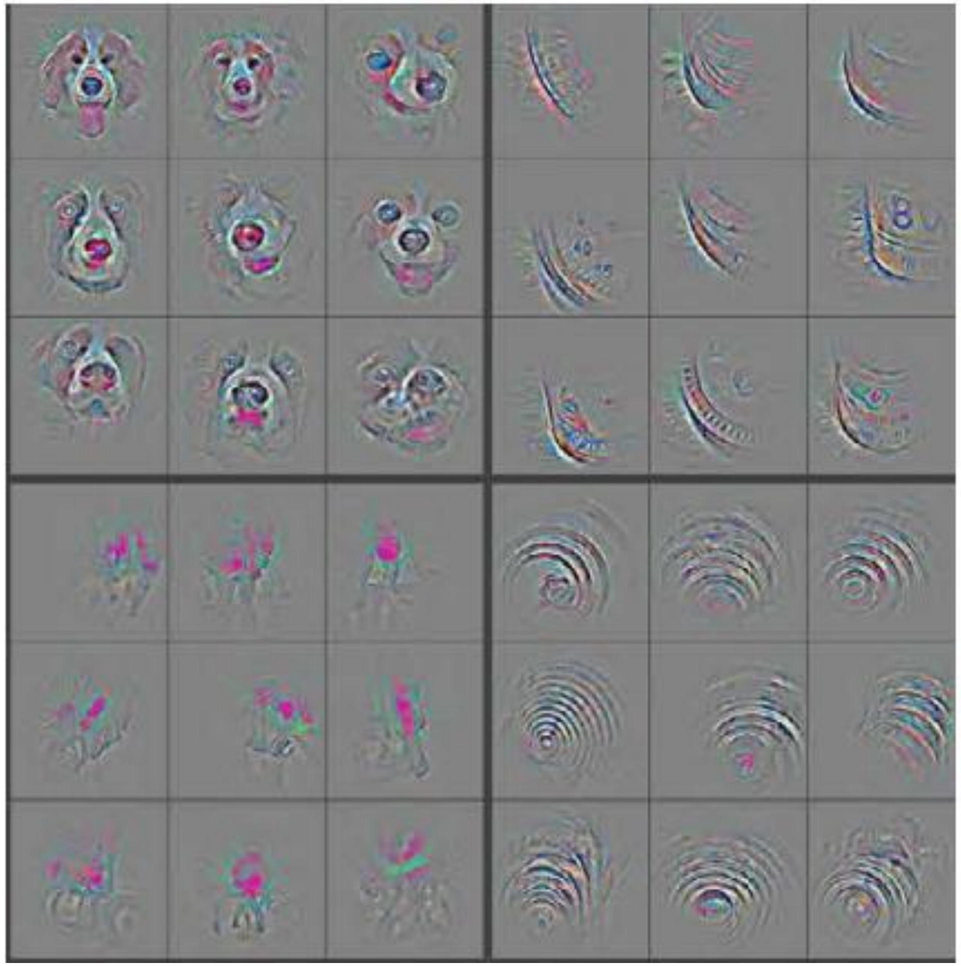


Human detector

Matthew D. Zeiler and Rob Fergus, "Visualizing and Understanding Convolutional Networks," *ECCV*, 2014.

(Recap) Learned CNN Kernels in a Trained AlexNet

Learned CNN kernels

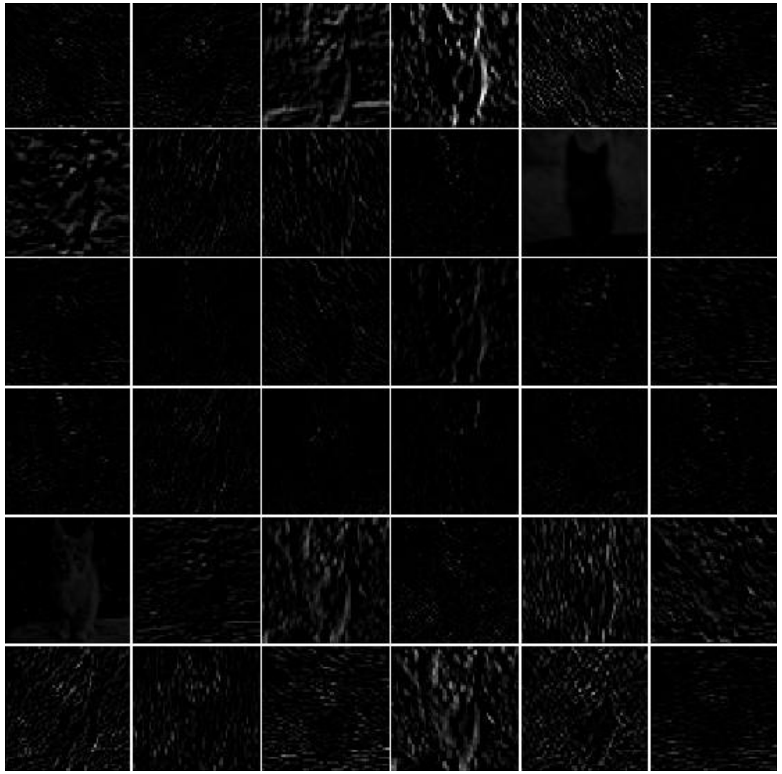


Top activations

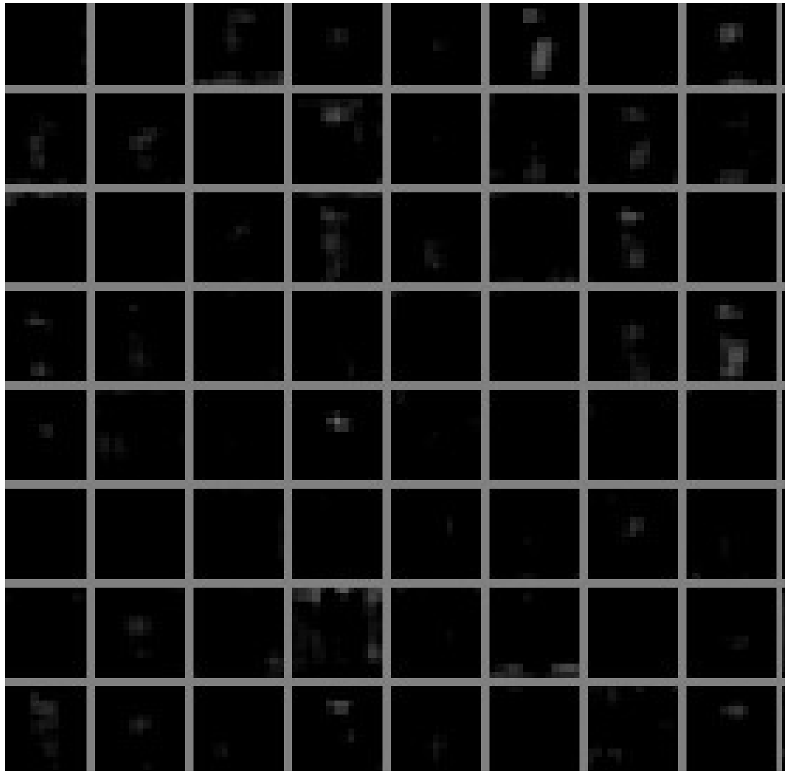
Matthew D. Zeiler and Rob Fergus, "Visualizing and Understanding Convolutional Networks," *ECCV*, 2014.

(Recap) Activations in a Trained AlexNet

1st convolutional layer



5th convolutional layer



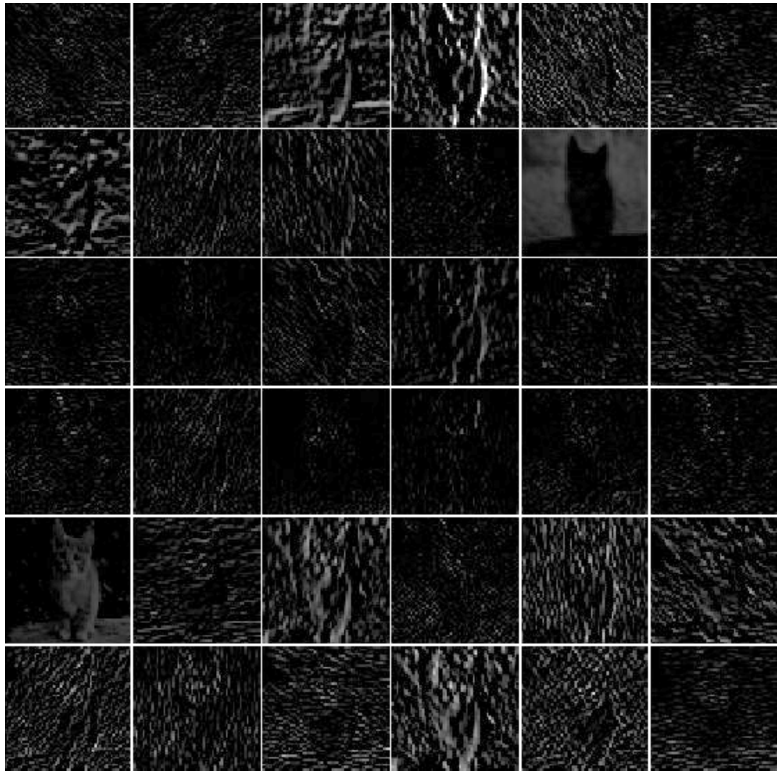
cs231n.github.io/understanding-cnn/

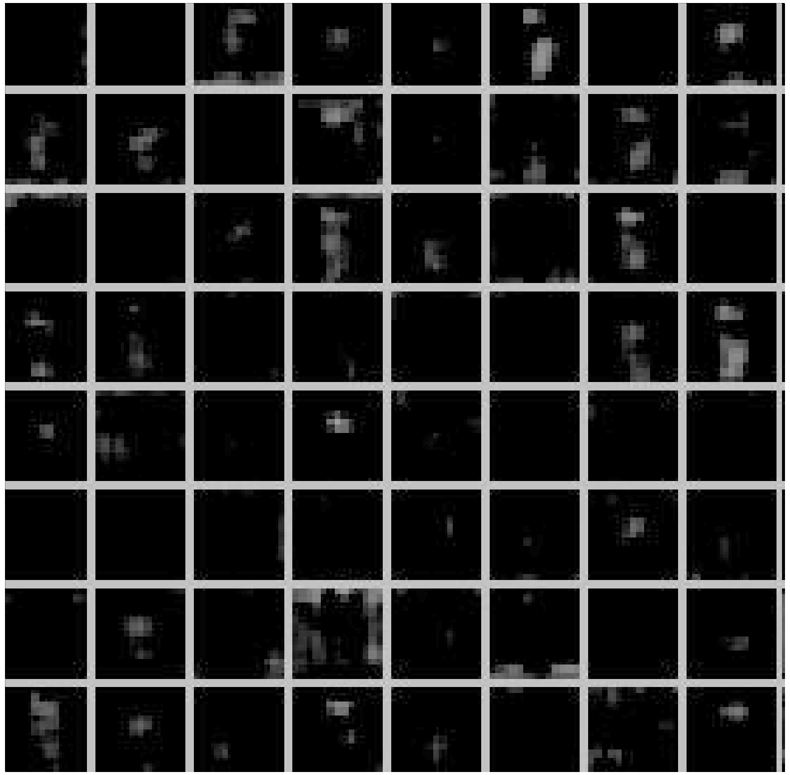
(Recap) Activations in a Trained AlexNet

1st convolutional layer

5th convolutional layer

Brightness+
Contrast+



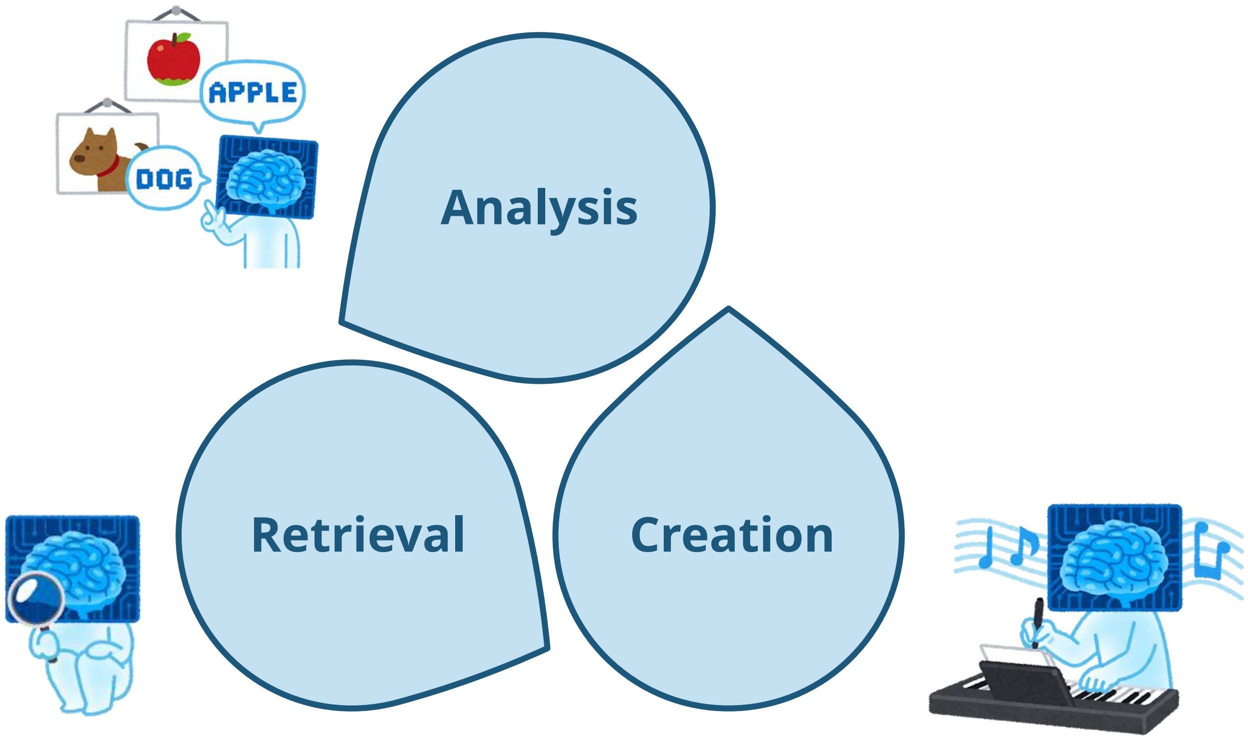


Brightness+
Contrast+

cs231n.github.io/understanding-cnn/

(Recap) Music Information Research (MIR)

- "Intelligent ways to analyze, retrieve and create music" (Yang 2018)



Yang, "Music Information Research," SNHCC, TIGP, lecture notes, April 2018.

Music Classification

Music Classification Tasks

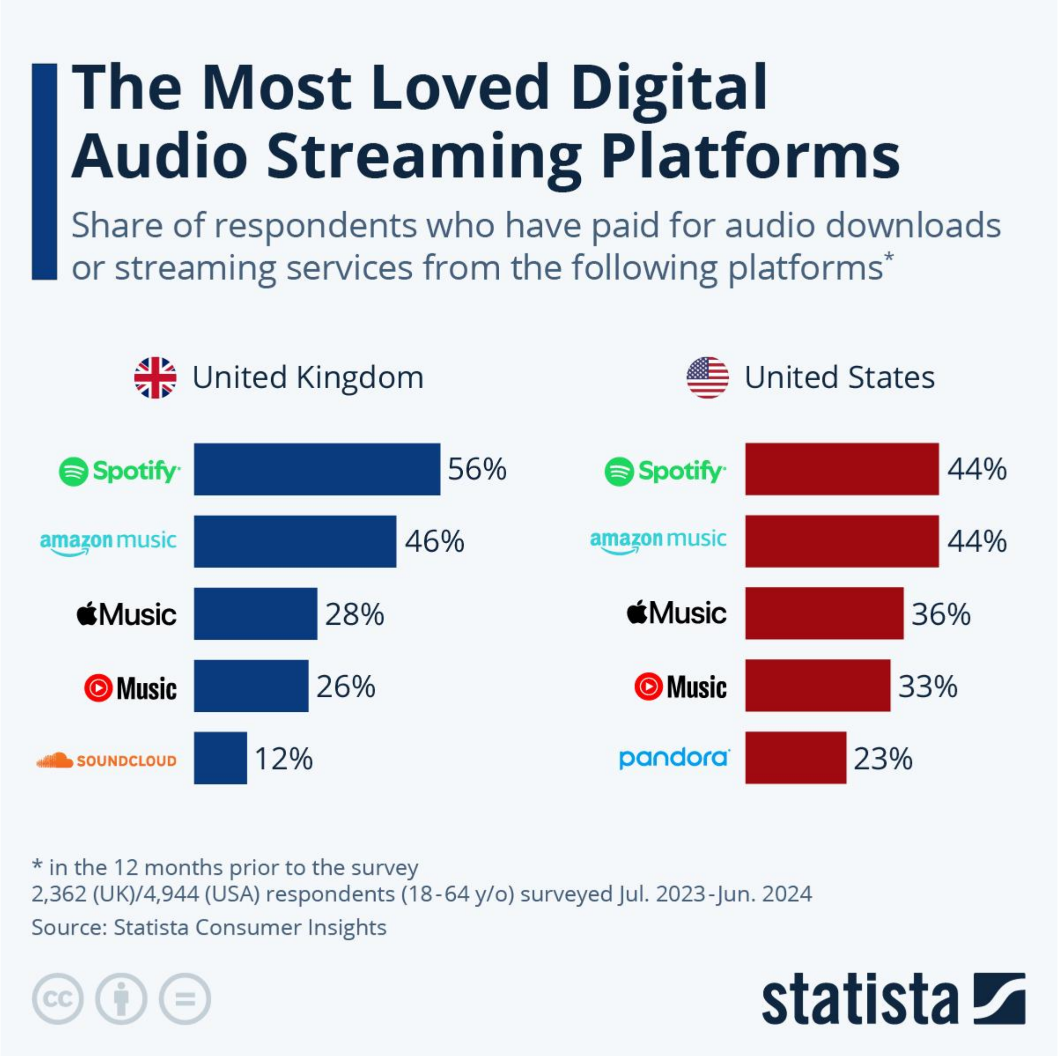
- **Genre** classification (pop, rock, r&b, jazz, hip-hop, classical, etc.)
- **Mood** classification (happy, sad, calm, aggressive, cheerful, etc.)
- **Instrument** recognition
- **Composer** identification
- **Key** detection
- **Chord** estimation
- **Music tagging** → **Can cover everything above!**

Applications of Music Classification Models

- Recommendation
- Curation
- Playlist generation
- Listening behavior analysis
- Musicology research

Minz Won, Janne Spijkervet, and Keunwoo Choi, "Music Classification: Beyond Supervised Learning, Towards Real-world Applications," *Tutorials of ISMIR*, 2021.

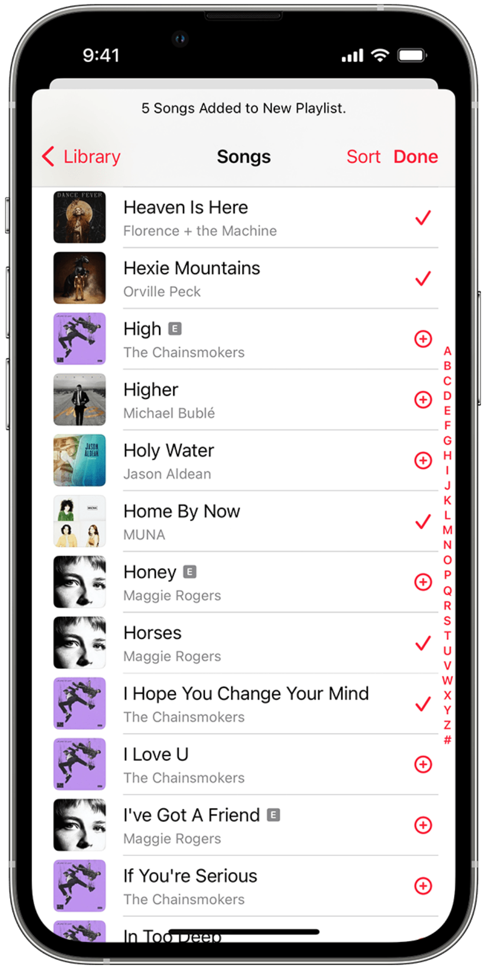
Music Classification for Recommendation

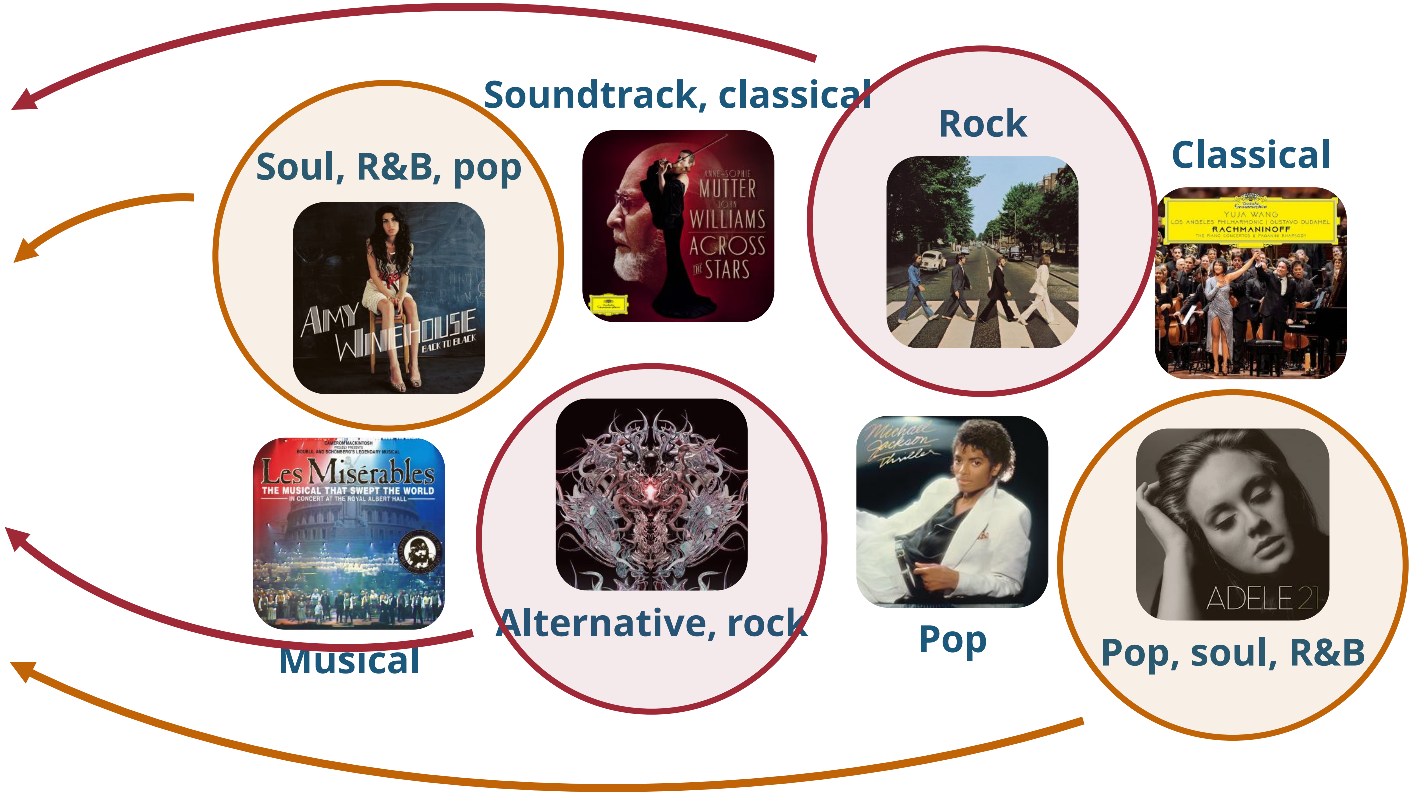


Anna Fleck, "The Most Loved Digital Audio Streaming Platforms," *Statista*, September 30, 2024.

Music Classification for Recommendation

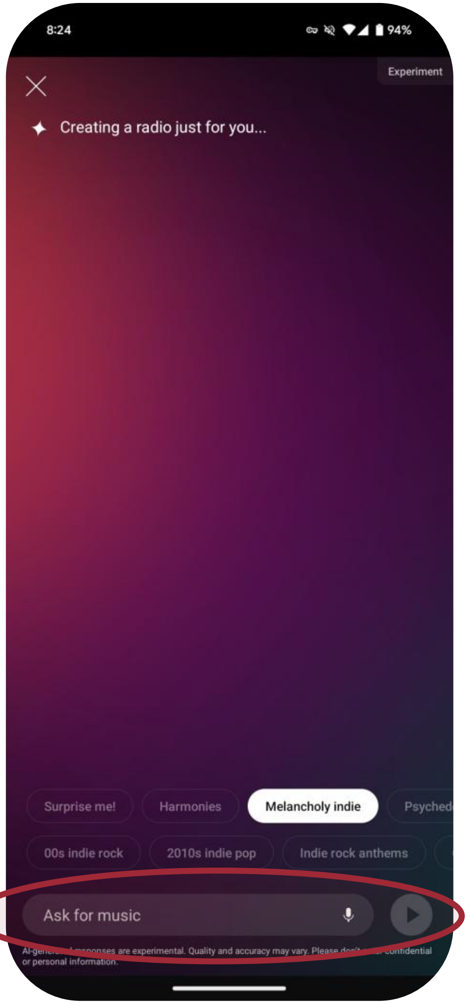
Music Classification for Playlist Generation

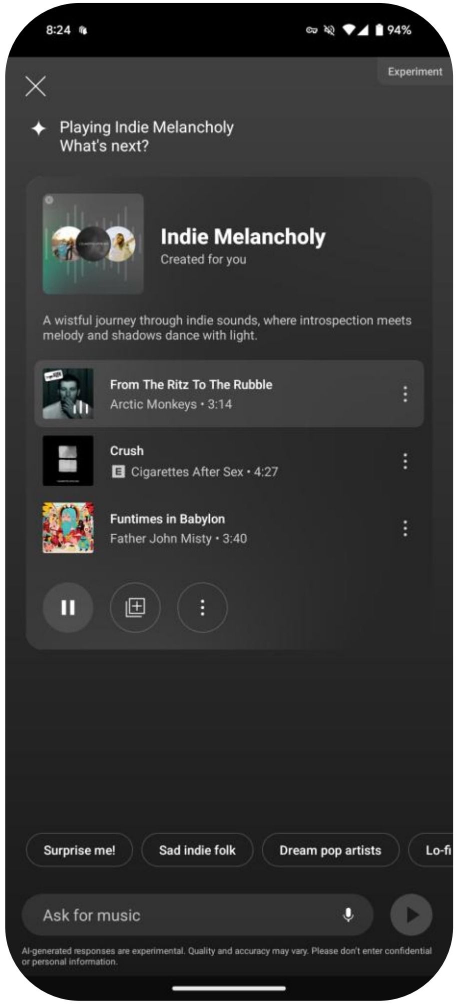




Ask Music (YouTube Music)

Text prompt

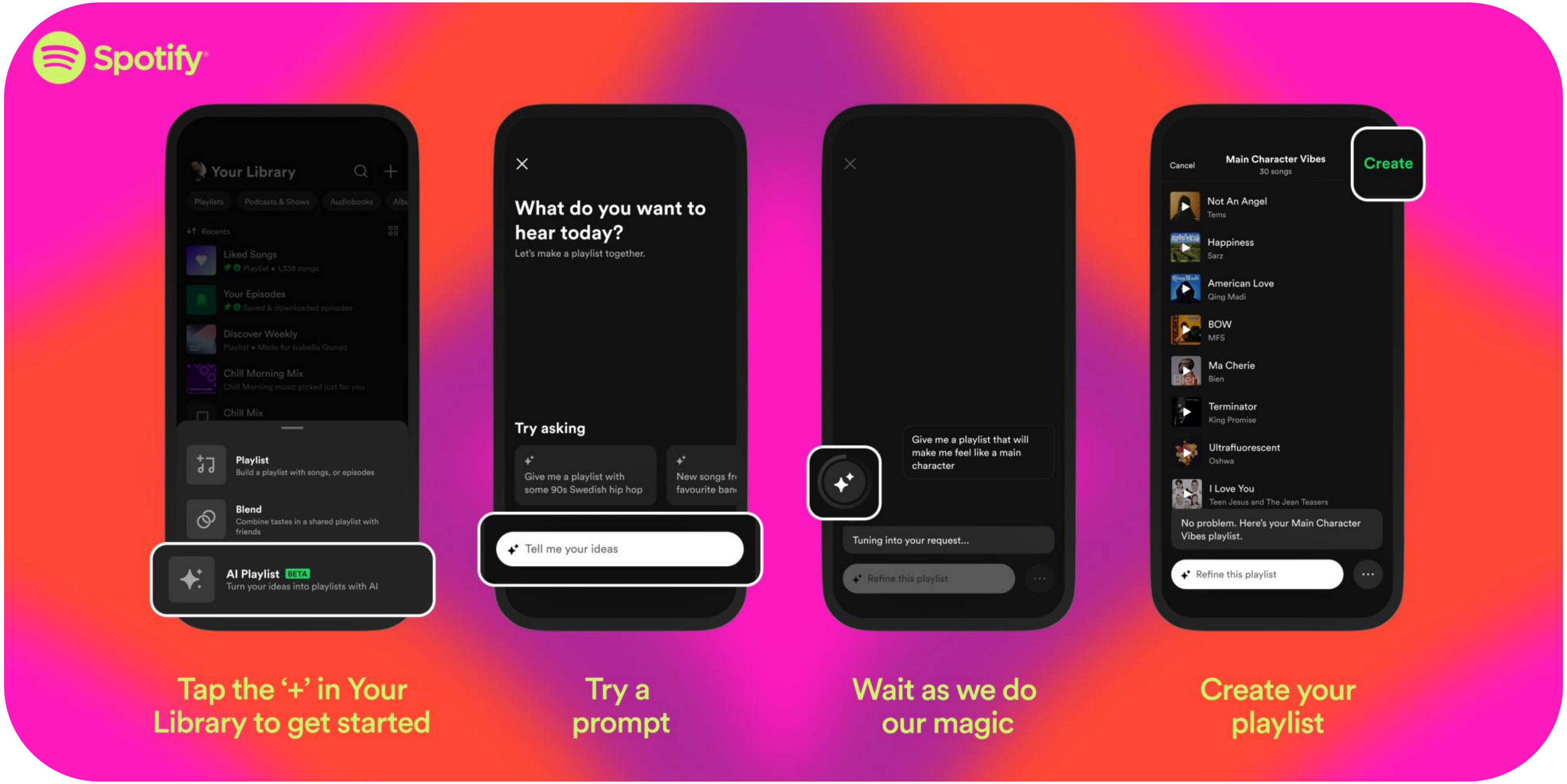




(Source: Android Police)

androidpolice.com/youtube-music-ask-music-ai-playlist-assistant/

AI Playlist (Spotify)

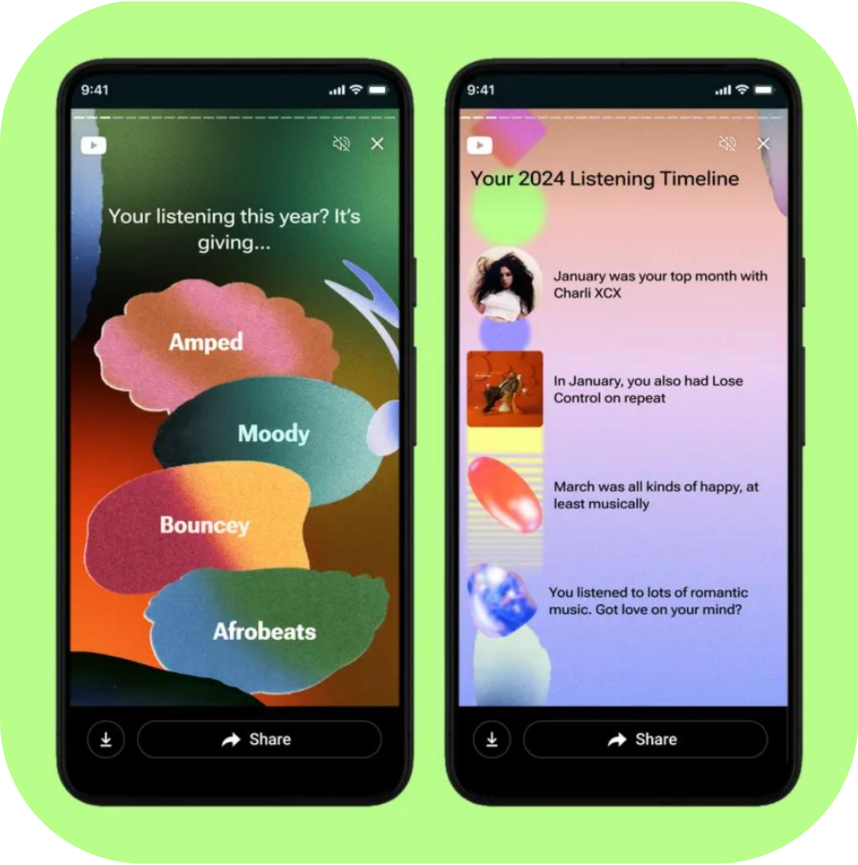


(Source: Spotify)

newsroom.spotify.com/2024-04-07/spotify-premium-users-can-now-turn-any-idea-into-a-personalized-playlist-with-ai-playlist-in-beta/

Music Classification for Listening Behavior Analysis

YouTube's Music Recap



(Source: YouTube)

Spotify's Listening Personality

(Source: Spotify)

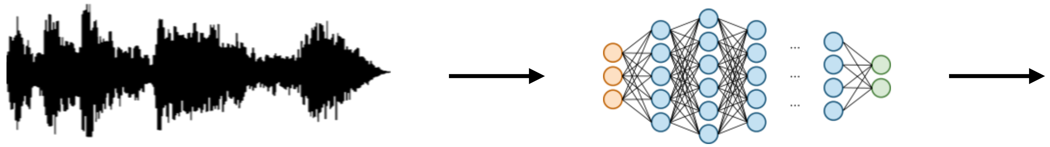
blog.youtube/news-and-events/2024-music-recap-youtube/
engineering.atspotify.com/2023/01/whats-a-listening-personality/

Types of Classification Tasks

Types of Classification Tasks

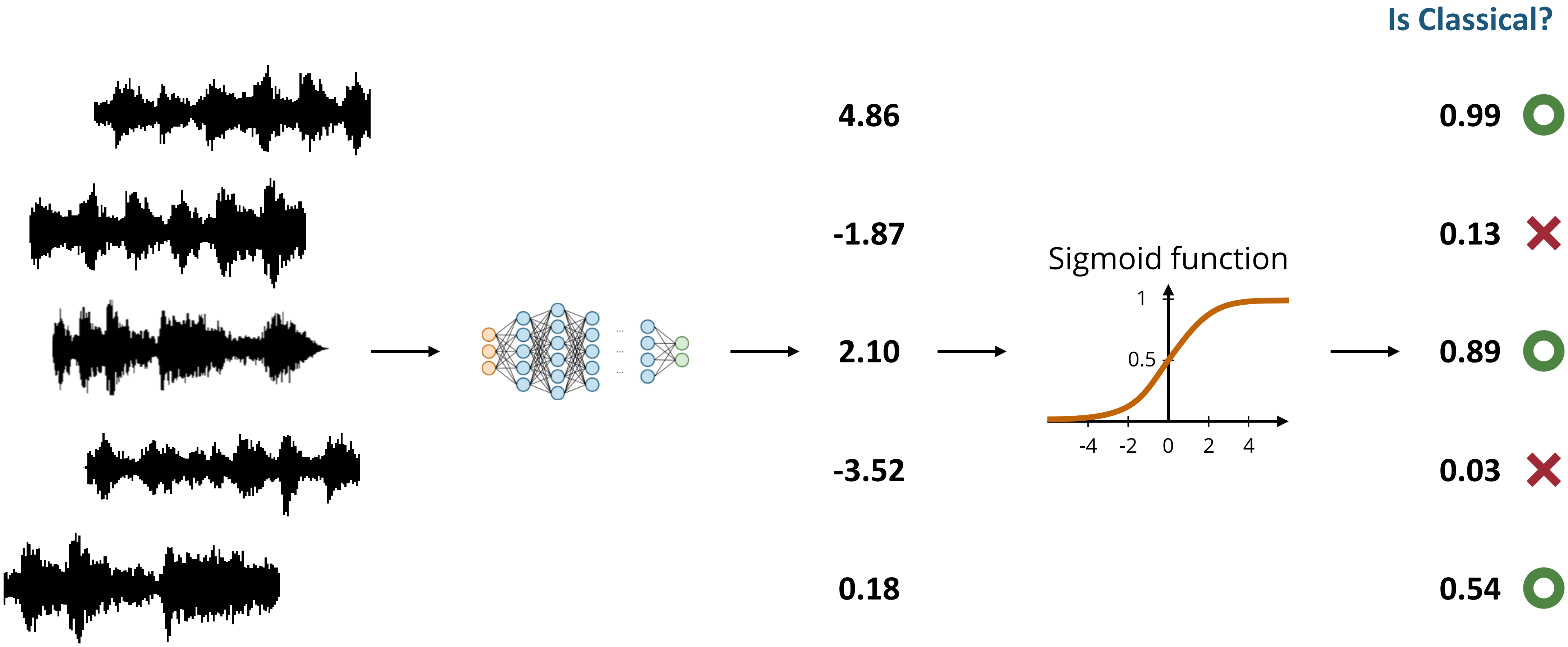
- **Binary** classification
- **Multiclass** classification
- **Multi-label** classification

Binary Classification

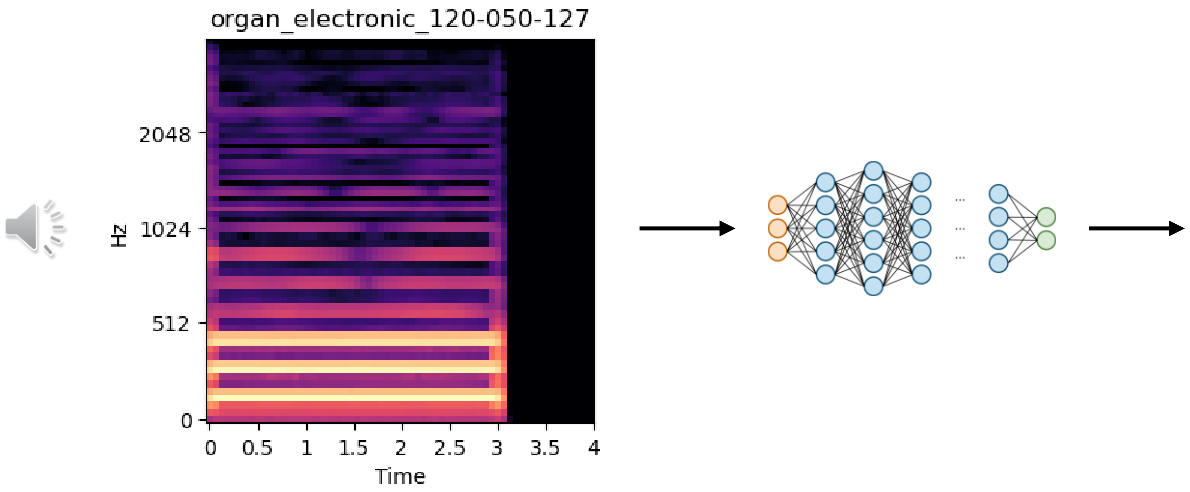


Is Classical?

Binary Classification

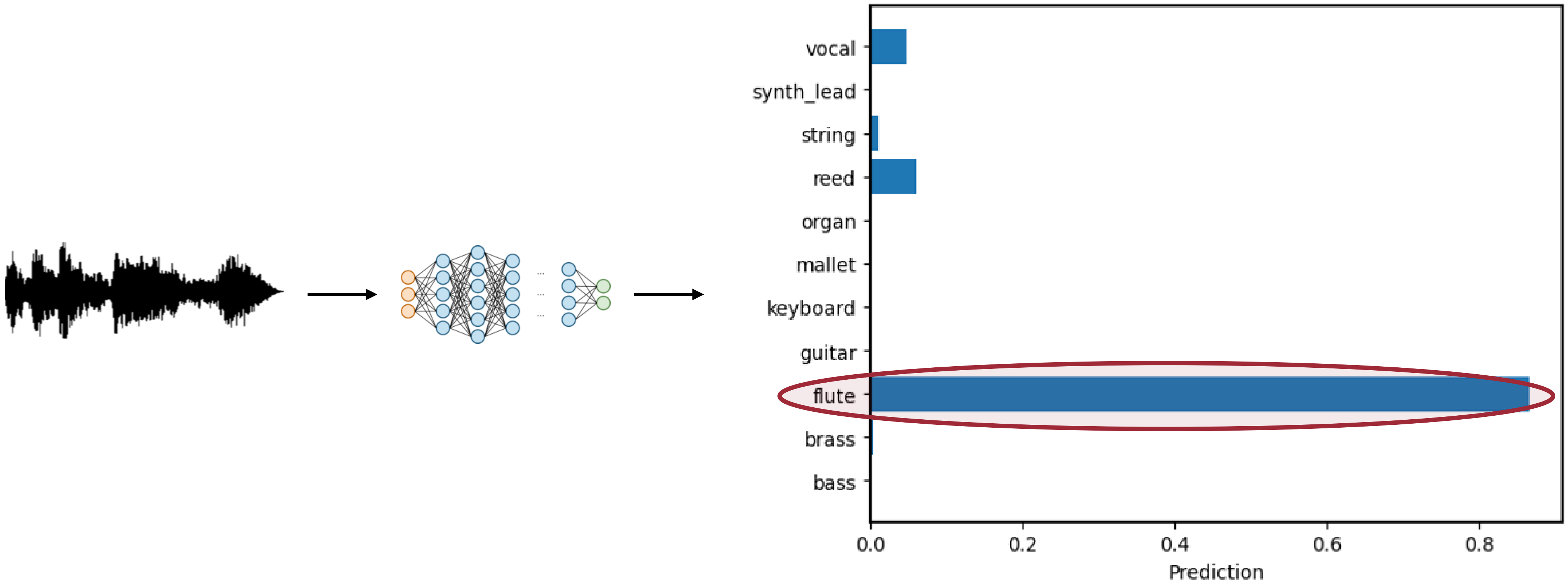


Multiclass Classification



Bass
Brass
Flute
Guitar
Keyboard
Mallet
Organ
Reed
String
Synth
Lead
Vocal

Multiclass Classification



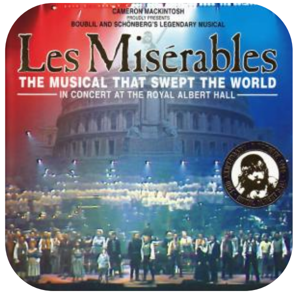
Multi-label Classification

Soul, R&B, pop

Soundtrack, classical

Rock

Classical



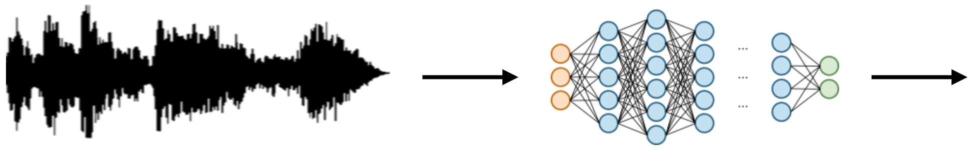
Musical

Alternative, rock

Pop

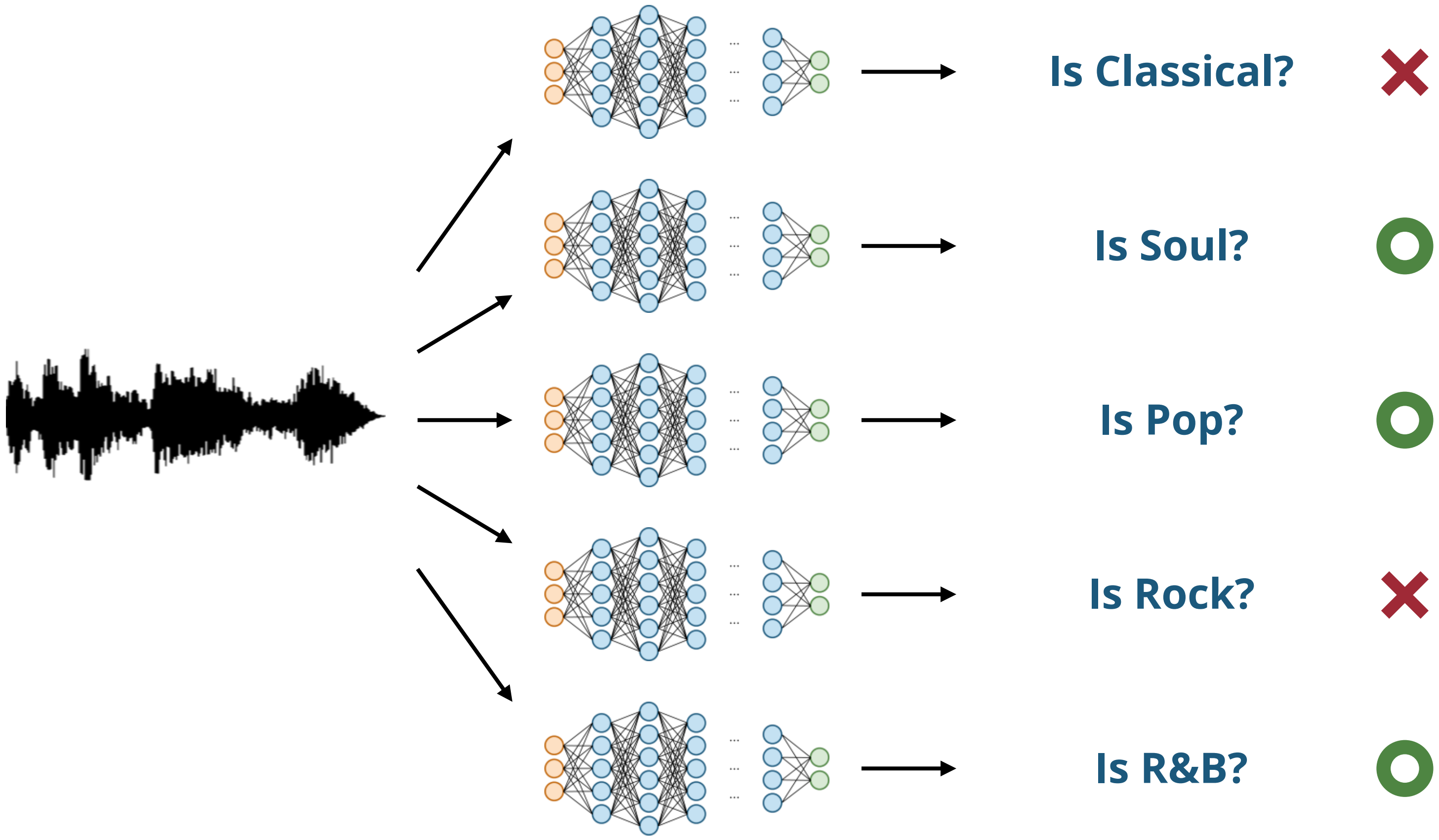
Pop, soul, R&B

Multi-label Classification

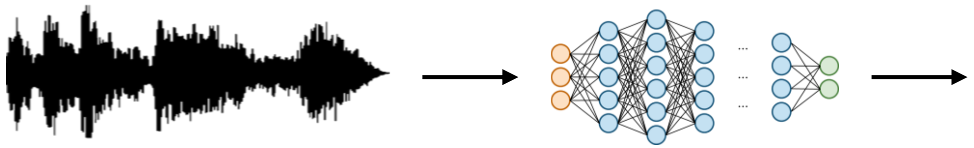


Soul, R&B, pop

Multi-label Classification as Binary Classification



Multi-label Classification as Multi-class Classification



{Pop}

{R&B}

{Soul}

{Classical}

{Pop, R&B}

{Pop, Soul}

…

{Pop, Classical}

{Pop, R&B, Soul}

…

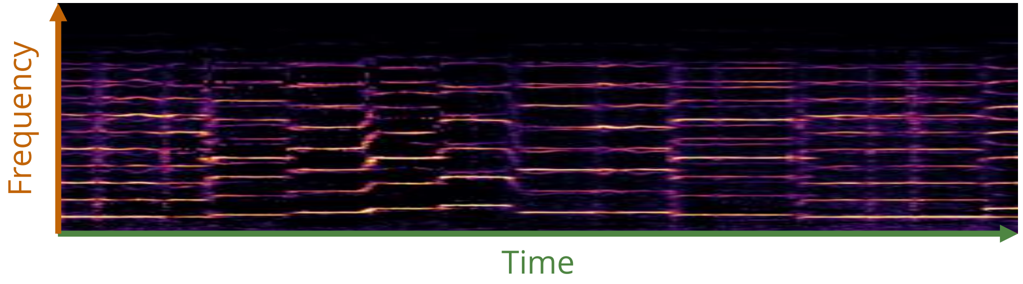
{R&B, Soul, Classical}

{Pop, R&B, Soul, Classical}

Label powerset

Input Features

- **Waveform**
- Time-frequency representation (**spectrograms**)
- **Hand-crafted features** or **features provided in metadata**
  - Acoustic: loudness, pitch, timbre
  - Rhythmic: beat, tempo, time signature
  - Tonal: key, scale, chords
  - Instrumentation, expressions, structures, etc.



Common Datasets

- **GTZAN**: 1,000 30-sec songs, 10 genres
- **MagnaTagATune**: 5,405 29-sec songs, 188 tags, 230 artists
- **Million Song Dataset (MSD)**: **1M** 30-sec songs, >500K tags, **tricky to access**
- **Free Music Archive (FMA)**: >10K **full** songs, 163 genres
- **MTG-Jamendo**: 55K **full** songs, 195 tags
- **AudioSet**: **1M** songs, YouTube URLs, **low-quality audio**
- **NSynth**: ~306K 4-sec instrument sounds

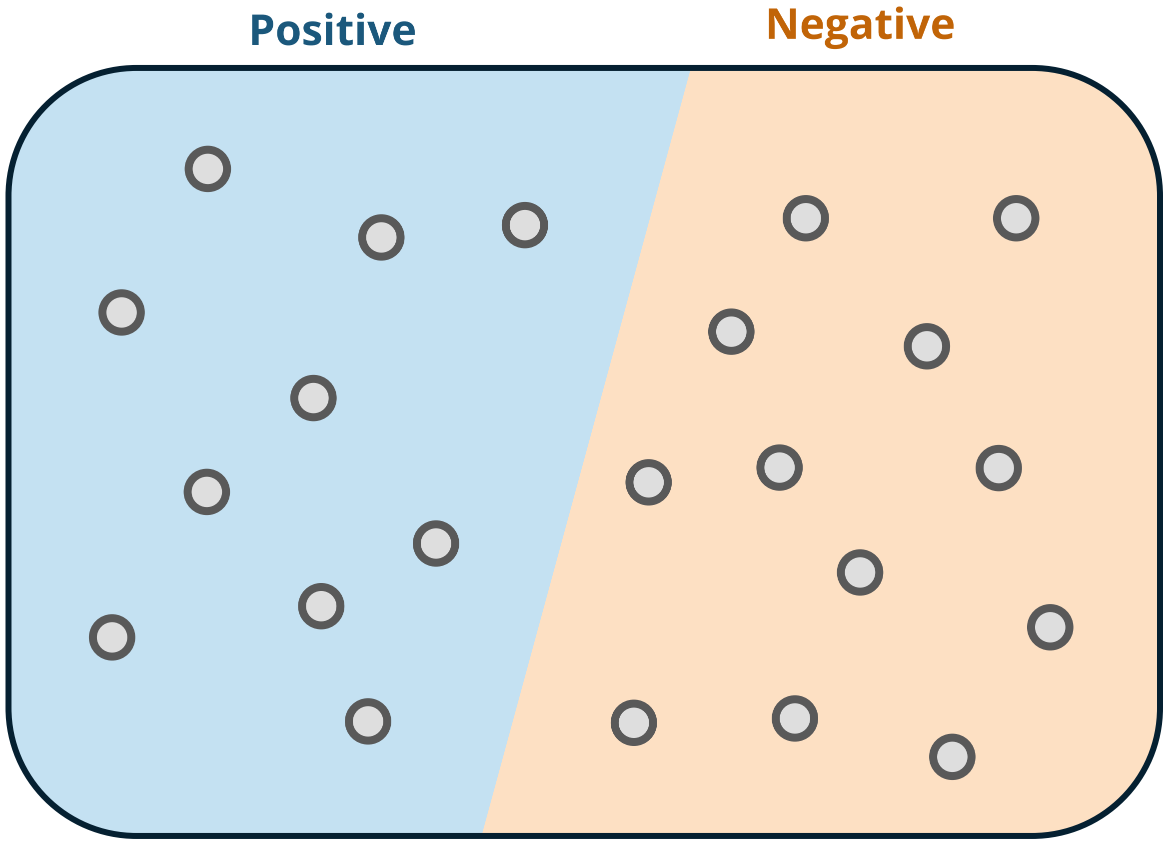
Minz Won, Janne Spijkervet, and Keunwoo Choi, "Music Classification: Beyond Supervised Learning, Towards Real-world Applications," *Tutorials of ISMIR*, 2021.

Evaluation Metrics

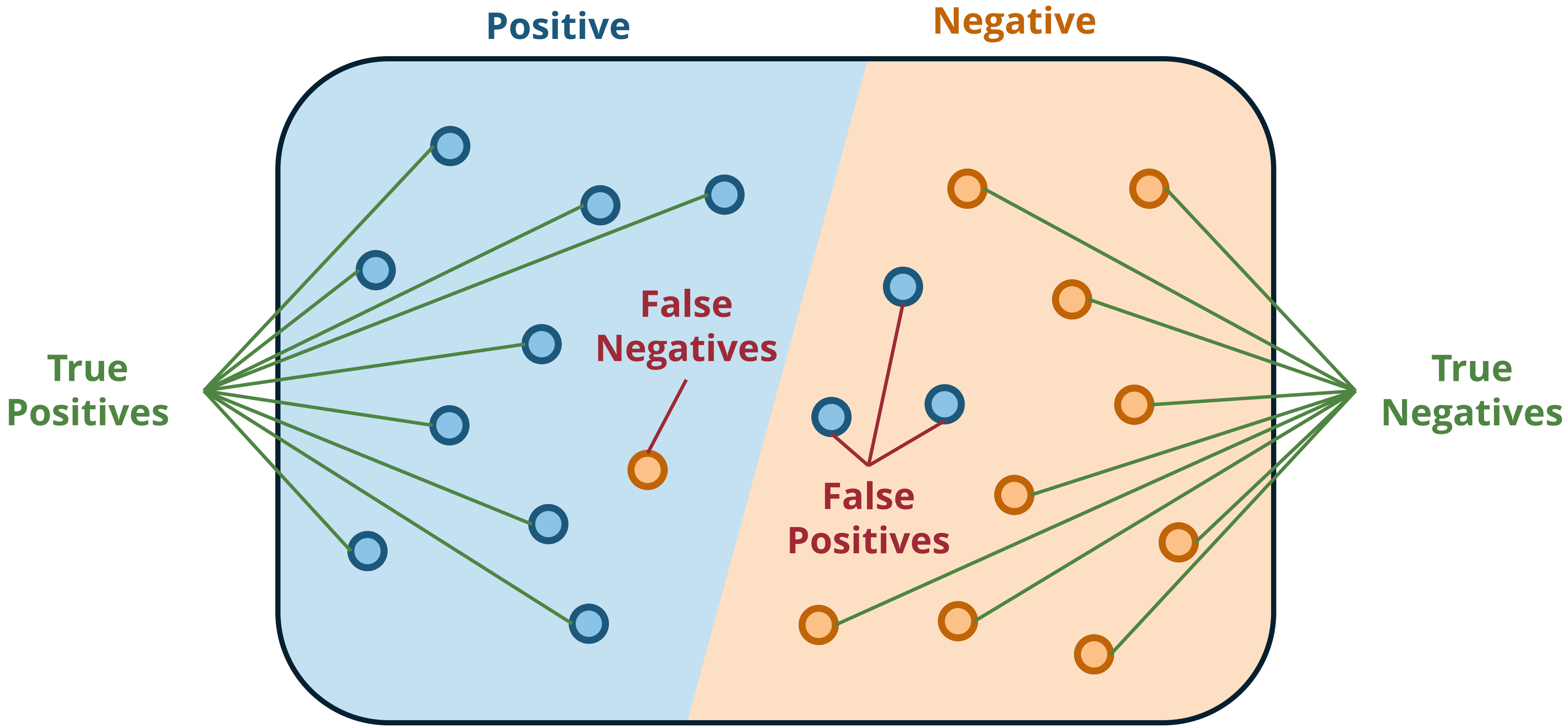
Evaluation Metrics

- **Key**: Capture what you care the most!
- The best evaluation metric depends on the actual use case
- Best to use several evaluation metrics to obtain a holistic view of your model's performance

Toy Example: Binary Classification

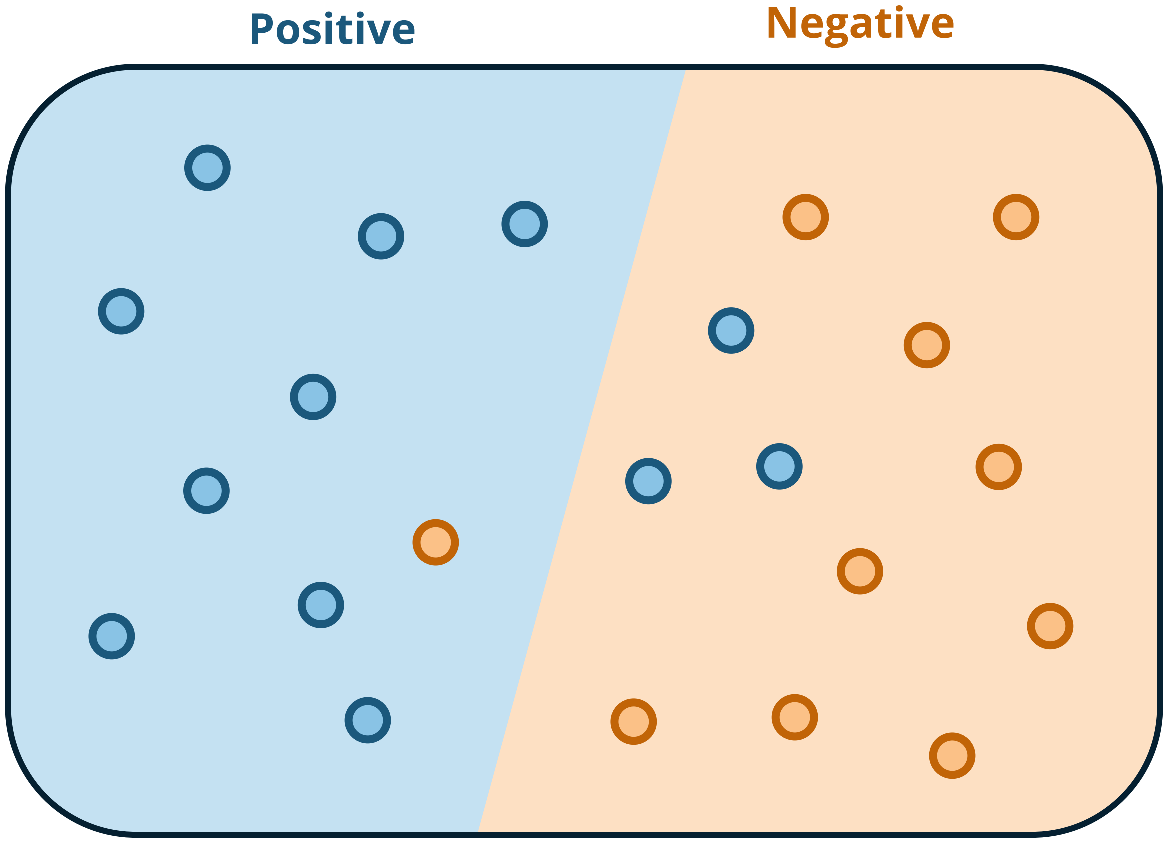


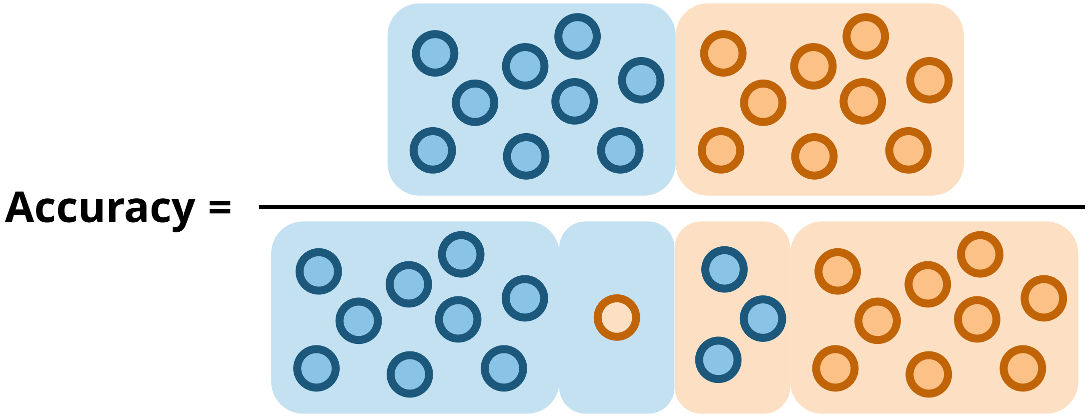
Toy Example: Binary Classification



Accuracy

- **Definition:** Percentage of correct predictions across all classes





= 0.82

Confusion Matrix for Binary Classification

		Prediction	
		Positive	**Negative**
Ground truth	**Positive**	True Positives (TP)	False Negatives (FN)
	Negative	False Positives (FP)	True Negatives (TN)

Confusion Matrix for Binary Classification

		Prediction	
		Positive	**Negative**
Ground truth	**Positive**	True Positives (TP)  **Cat**	False Negatives (FN)  **Not Cat**
	Negative	False Positives (FP)  **Cat**	True Negatives (TN)  **Not Cat**

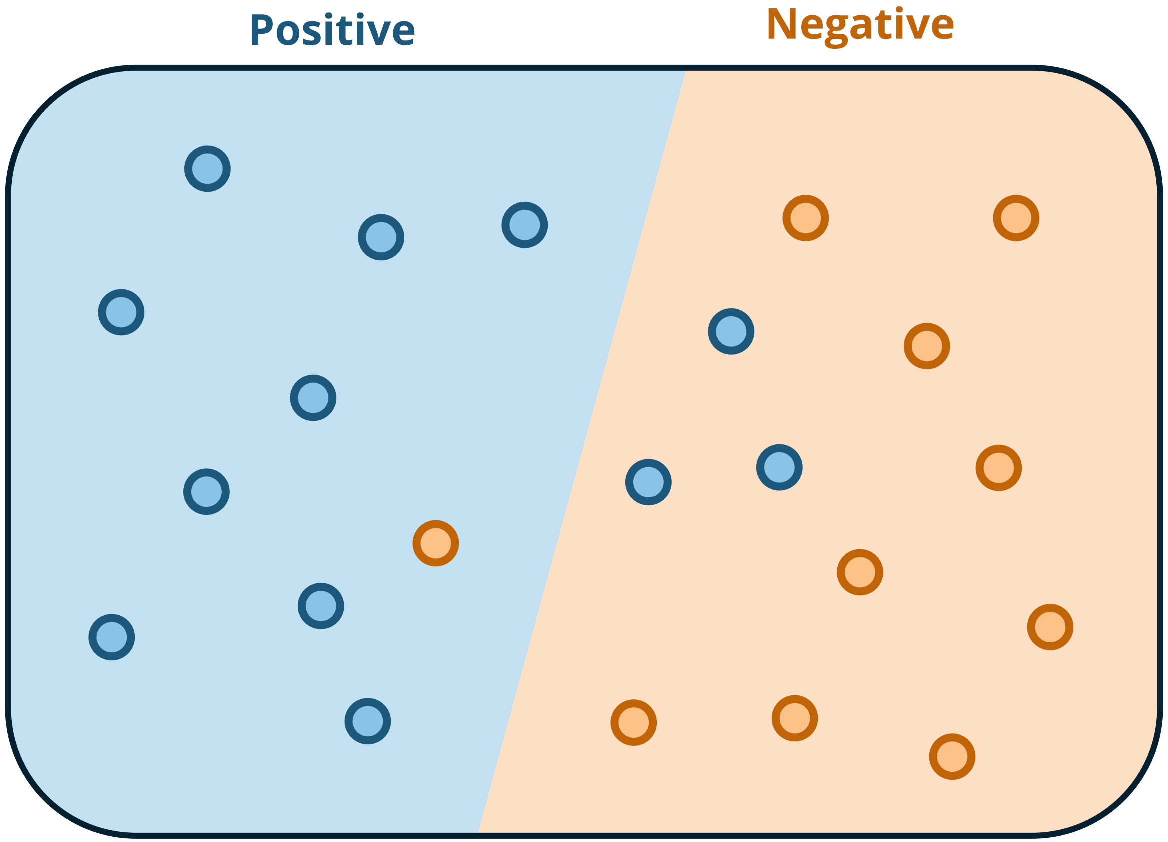
Accuracy on Imbalanced Datasets

- Accuracy does ***not*** work well on **imbalanced dataset**
- Take **a disease with a 1% prevalence** for example:
  - What if we simply say **negative to all diagnoses**?

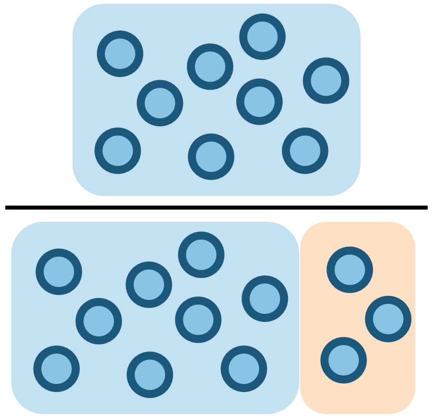
		Prediction	
		Positive	**Negative**
Ground truth	**Positive**	True Positives (TP) 0	False Negatives (FN) 1
	Negative	False Positives (FP) 0	True Negatives (TN) 99

Accuracy = 0.99

Precision

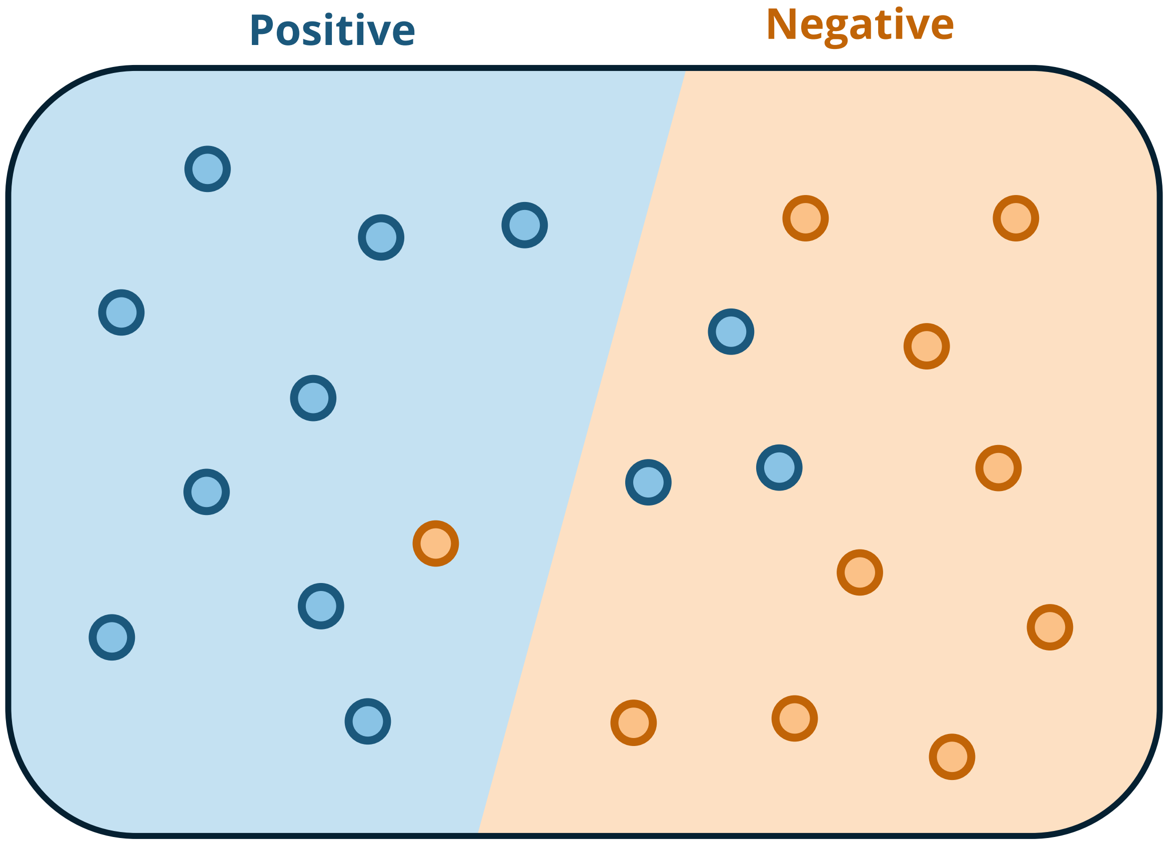


$$\text{Precision} = \frac{TP}{TP + FP} = \frac{\text{[image]}}{\text{[image]}} = 0.75$$

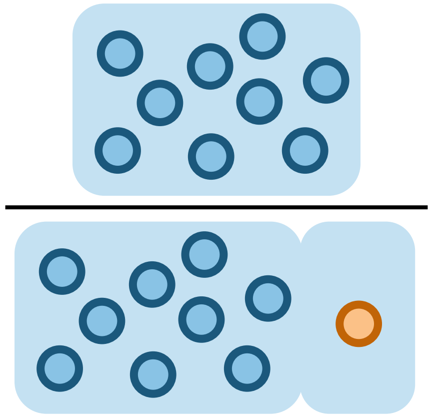


How often predictions for the positive are correct

Recall



$$\text{Recall} = \frac{TP}{TP + FN} =$$



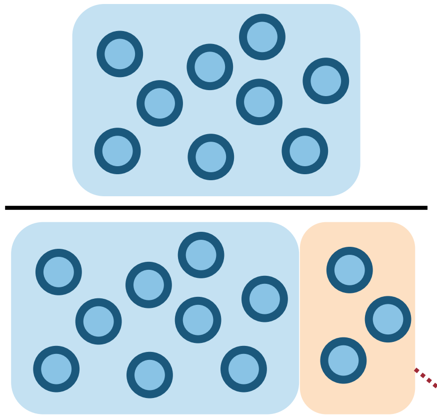
$= 0.9$

How well the model finds all positive instances in the dataset

Precision vs Recall

Precision

$$\frac{TP}{TP + FP} =$$



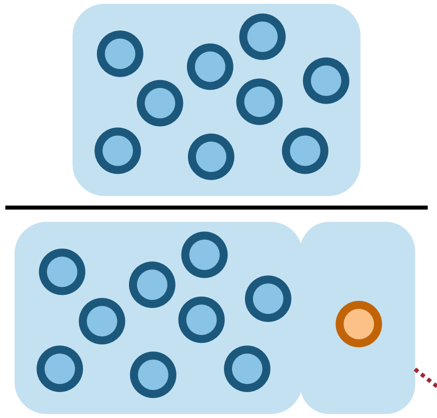
$= 0.75$

False alarms

How often predictions for the positive are correct

Recall

$$\frac{TP}{TP + FN} =$$



$= 0.9$

Missed detection

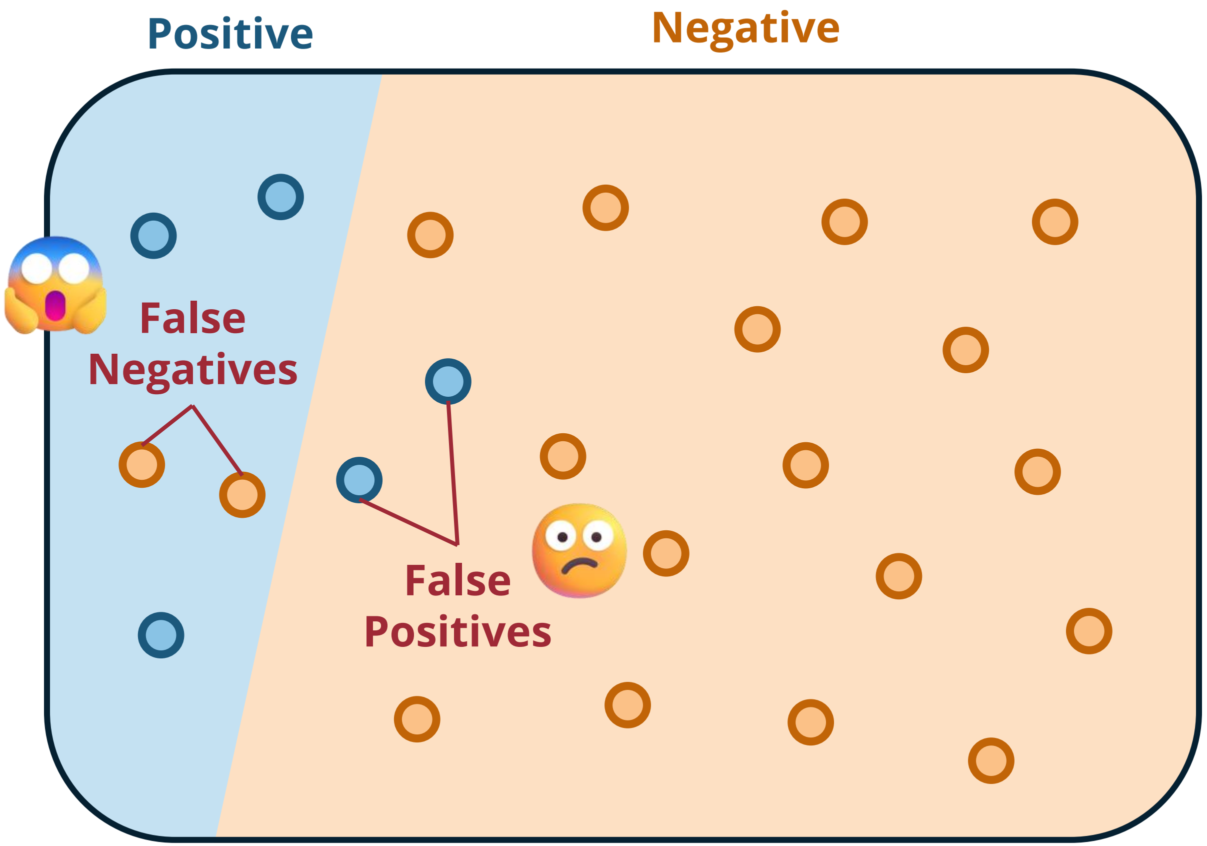
How well the model finds all positive instances in the dataset

When should we care about Precision & Recall?

Rare cancer detection

Aim for high precision or high recall?

High recall ensures most cancer cases are identified.



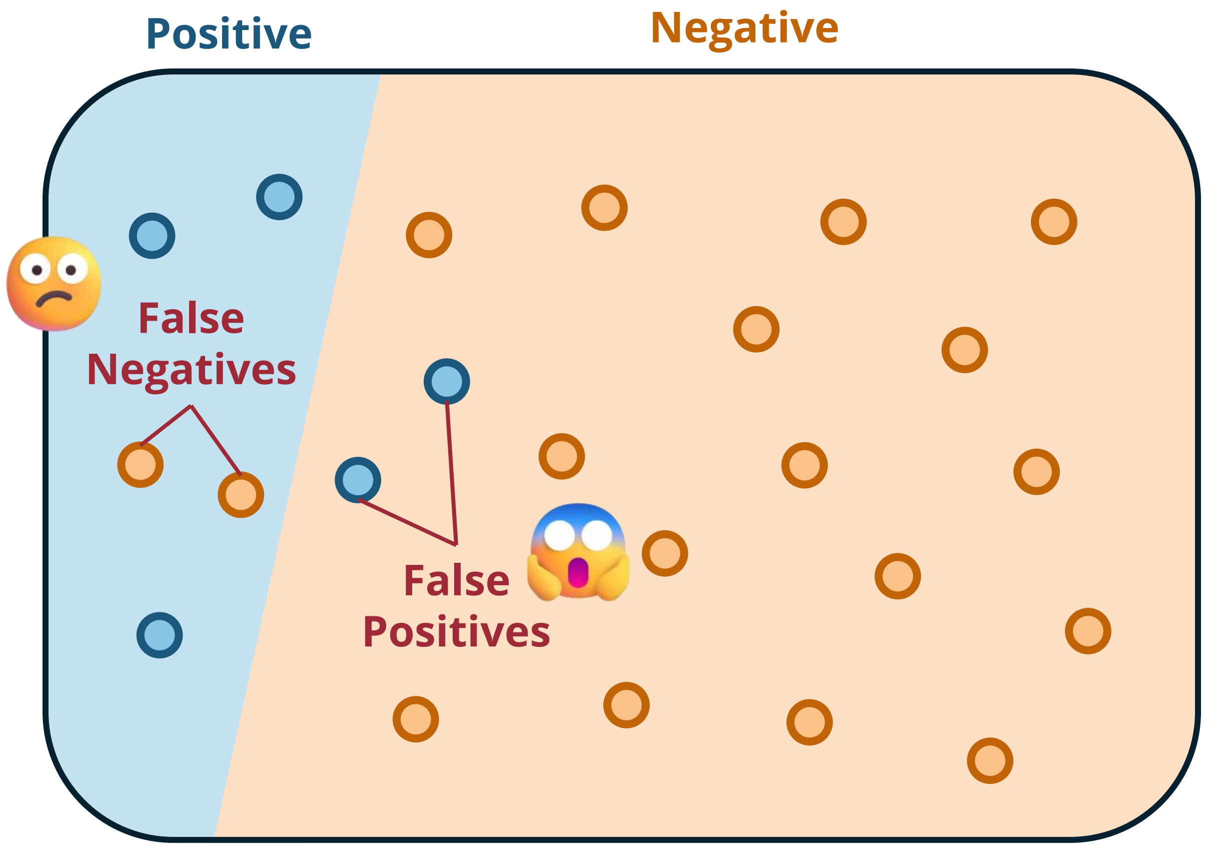
False alarms vs Missed detections

When should we care about Precision & Recall?

Music recommendation

Aim for high precision or high recall?

High precision ensures that the model won't recommend irrelevant items.



False alarms vs Missed recommendations

F1 Score: Considering both Precision & Recall

- Particularly useful for **imbalanced datasets**
  - Work better than accuracy when the dataset is imbalanced
  - For example, music search, retrieval, and recommendation

$$F_1 = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}}$$

$$= \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$$

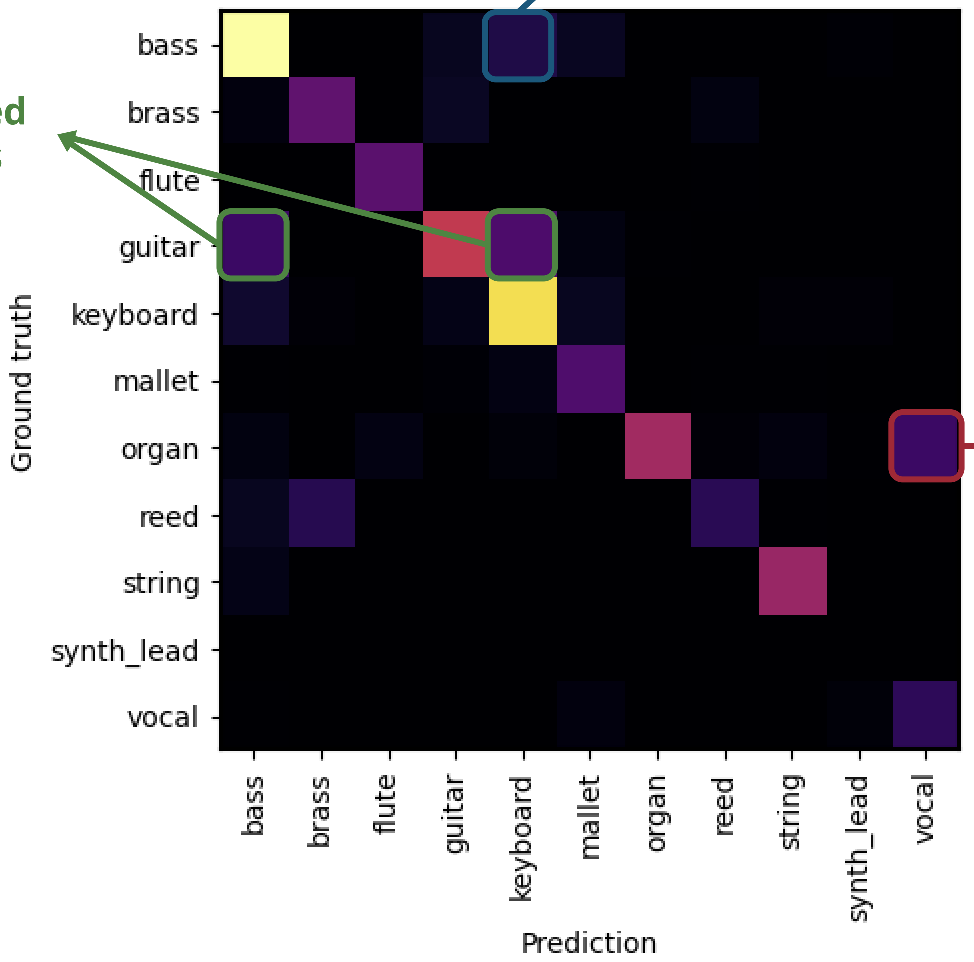
Confusion Matrix for Binary Classification

		Prediction	
		Positive	Negative
Ground truth	Positive	True Positives (TP)  Cat	False Negatives (FN)  Not Cat
	Negative	False Positives (FP)  Cat	True Negatives (TN)  Not Cat

Confusion Matrix for Multiclass Classification

Bass sounds got classified as keyboard sounds

Guitar sounds got classified as bass/keyboard sounds



Organ sounds got classified as vocals

Optional Reading

- Minz Won, Janne Spijkervet, and Keunwoo Choi, "Music Classification: Beyond Supervised Learning, Towards Real-world Applications," *Tutorials of ISMIR*, 2021.

Open Source Music Classification Models

- github.com/minzwon/sota-music-tagging-models
- github.com/jordipons/musicnn