Investigating Auditory Concepts in Deep Neural Networks

A thesis submitted in partial fulfillment
of the requirements for the degree of

Master of Science
in
Computational Linguistics
by Research

by

Pratyaksh Gautam
2020114002
pratyaksh.g@research.iiit.ac.in

Advised by Dr. Vinoo Alluri and Dr. Makarand Tapaswi

International Institute of Information Technology Hyderabad
500 032, India

June 2026

International Institute of Information Technology Hyderabad
Hyderabad, India

CERTIFICATE

This is to certify that work presented in this thesis titled ***Investigating Auditory Concepts in Deep Neural Networks*** by *Pratyaksh Gautam* has been carried out under our supervision and is not submitted elsewhere for a degree.

Date	Dr. Vinoo Alluri	Dr. Makarand Tapaswi

To Mom, Dad, and Piya di.

Acknowledgements

This thesis has been a challenging undertaking, one in which I have received the support of many people in my life, more than I can fully acknowledge in this space. I'm grateful to my advisors, Dr. Vinoo Alluri and Dr. Makarand Tapaswi, whose careful consideration and guidance was instrumental in bringing the thesis to its current shape. They have pushed me to think critically and to persevere through challenges, which has led this journey of research to be an invaluable life experience for me. In addition, I would also like to thank all of my lecturers, who have instilled even more curiosity in me, and the college staff whose quiet contributions allow us a college lifestyle we sometimes take for granted.

I have immense gratitude for all of my friends in college; their presence made it so that I truly felt at home in a distant city, and they are the reason I carry a heart full of fond memories with me about a special time in my life wherever I go. My friends Abhinav, Shashwat and Lakshmanan have been true inspirations to me, people I can look up to while also never having to put my guard up around. I have shared incredible experiences with Aneesh, Pranjal, and Manav, who have become like brothers to me. People in my life like Priyanshul and Anirudh have opened my eyes in more ways than one, and been pivotal to my journey not just as a student, but as a human being. I'm grateful to my seniors - Aakash, Raghav and Aditya - who have made me feel like family, as well as my juniors - Nanda, Vedansh and Hardik - to whom I hope I have been able to extend the same feeling. I'm grateful for the vibrant club culture, and would specially like to thank all the members of the Music Club, Theory Group and OSDG.

I've been grateful to have the support of many of my other friends as well - Aaghaz, Sneha, Amrtanshu, Riti, Ananya, Priyanshi, Shubh and Shiksha - who have been there through thick and thin. Finally, I feel blessed to have a family whose support always allowed me to feel reassured, and keep my confidence in my abilities throughout this arduous journey.

Abstract

Deep neural networks have achieved strong performance across a wide range of auditory tasks, from speech recognition to music classification. Despite this success, their internal decision-making processes remain poorly understood, limiting both scientific insight and practical trust. In real-world settings, models that rely on opaque or spurious cues can behave unpredictably, making interpretability a central challenge in modern deep learning. This thesis addresses this gap by asking a central question: do auditory deep neural networks organize sound in a hierarchical manner similar to human auditory perception?

To test this hypothesis, we conduct a large-scale probing analysis of several widely used audio architectures, including three CNN models (VGGish, CLAP, MobileNetV3), and an Audio Spectrogram Transformer (AST). Using linear probes across six tasks of increasing abstraction, we examine how different forms of auditory information are distributed across network depth. Across all convolutional architectures, we observe a clear and consistent hierarchy. Low-level tasks such as note name classification peak in early layers, while higher-level semantic tasks, including genre classification and speaker count estimation, depend on deeper representations. Although this hierarchy is less sharply delineated in the AST due to its global attention mechanism, the same overall progression remains evident. This suggests that hierarchical organization is a robust property of effective auditory models, even when architectural constraints differ.

Beyond identifying this structure, we also show complex auditory concepts emerge within a deep neural network. Focusing on musical audio, we show that an AST trained solely for genre classification develops explicit representations of musical instruments in intermediate layers. Steering vector interventions further show that these instrument representations actively influence genre predictions, demonstrating that they are causally involved in the model's decisions. These findings collectively establish a principled framework for analyzing, comparing, and causally testing internal representations in auditory deep neural networks, while also grounding modern audio models in theories of hierarchical auditory perception.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Deep learning and the need for interpretability

With the advent of deep learning [50], deep neural networks (DNNs) have shown a remarkable ability to perform complex tasks across a wide range of domains. These successes span language modeling [17, 29, 76], image classification [30, 48], and automatic speech transcription [9]. In the domain of non-speech audio specifically, DNNs have been deployed with great success to classify audio instances, identifying sounds, instruments, and musical genres [8, 31, 44]. Furthermore, multi-modal large language models (MLLMs) capable of processing both audio and text [24, 28, 37] have enabled capabilities such as captioning and question-answering for audio inputs. Beyond analysis, DNNs are now used to clone voices [5, 19] and generate music directly as raw waveforms [2, 25, 36, 53]. Collectively, these advances highlight the growing breadth and sophistication of deep learning in processing auditory information.

Despite their empirical success, the interpretability of DNNs remains a significant challenge; the internal mechanisms that give rise to model behavior are not yet fully understood. Because these models build understanding through many layers of non-linear transformations on high-dimensional data, it remains difficult to discern how their internal representations give rise to the behaviors we observe.

Studying how models arrive at their conclusions is often just as revealing as the conclusions themselves. A model might provide the correct output for a given task but rely on flawed reasoning hidden beneath the surface. Consider an image classifier that occasionally erroneously labels huskies as wolves, as shown in Fig. 1.1. At first glance, this seems like an understandable error, given the visual similarity between the two animals. However, a closer analysis shows that the misclassification is not driven by confusion in their physical features at all [64]. Instead, the model has learned to associate snow in the background with the label ‘wolf’. This is a spurious correlation, a coincidental pattern in the training set where wolves tended to appear in snowy environments. Because this cue was statistically reliable during training, the model adopted it as a shortcut, even though it bears no meaningful relation to the actual task of distinguishing between the animals.

While this example is relatively benign, the stakes can be markedly higher for other applications. Consider a medical image classifier that correctly diagnoses diseases, not by detecting patient features,

(a) An image of a husky. An image classification model mistakenly classifies it as an image of a wolf.

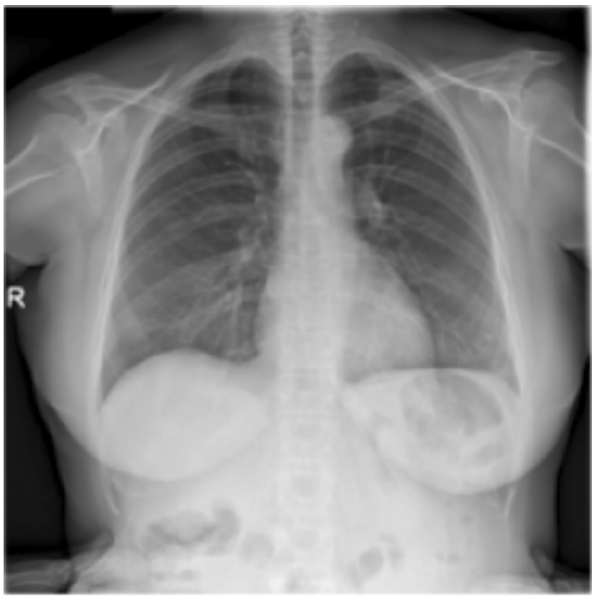
(b) The parts of the image found to be most important for the model's classification.

Figure 1.1: An image classification model might misclassify an image of a husky (1.1a) as that of a wolf, solely on the basis of the presence of snow in the image (1.1b). This is an example of a spurious correlation learnt by the model from the training data. Figure adapted from Ribeiro et al. [64].

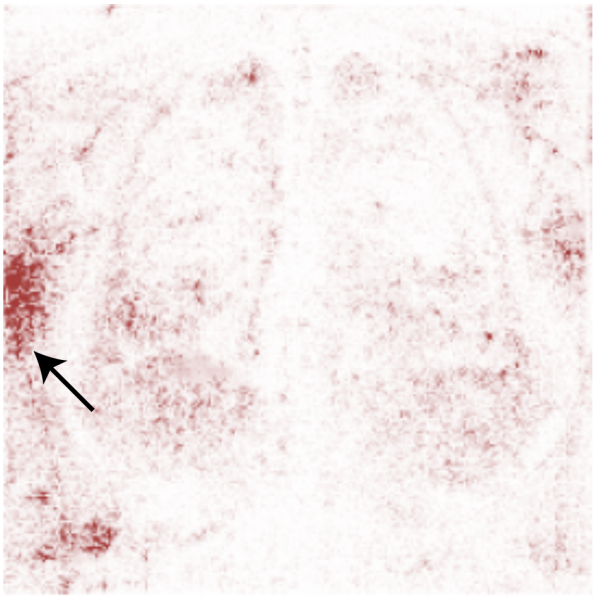
but by identifying orientation markers like 'left' or 'right' on the X-ray scans (see Fig. 1.2). This is yet another spurious correlation, a 'shortcut' learned from the training data [27]. Such success is deceptive; these shortcuts can fail catastrophically when the model is deployed in a new hospital or imaging setup where those specific markers are absent.

Additionally, achieving greater mechanistic insight into DNNs does more than prevent failure modes; it can advance our understanding of the domain itself. For instance, examining the features relied upon by protein-folding models has helped biologists identify previously overlooked structural patterns in real proteins [67]. The researchers originally aimed to apply interpretability tools to better understand the internal representations of protein language models, yet this analysis unexpectedly uncovered coherent biological features that highlighted missing or incomplete annotations in existing protein databases. By applying supervised pruning to retain only the specific subspaces (subsets of features) that best match human similarity ratings for a given category, Bavaresco et al. [10] were able to modify a deep image classification model to significantly improve its performance at identifying AI-generated images that aligned with human aesthetic and semantic preferences. The authors achieved this without any fine-tuning, demonstrating promise in the role of model interpretability in improving both alignment and performance of deep learning models.

Thus, the ability to interpret the models we use is crucial both for trust and for gaining insight into the tasks we use such models for. We see a clear need to understand the internal mechanisms of deep neural

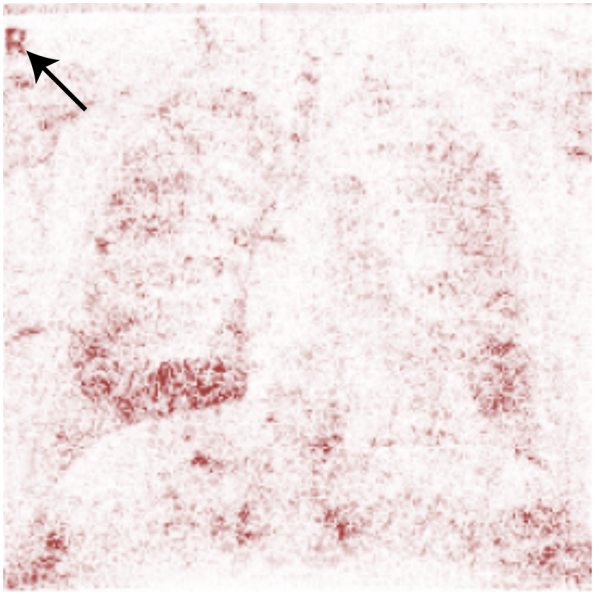


(a) A chest X-ray scan of a COVID-19 positive patient.



(b) Saliency map for the X-ray shown on the left (a).

(c) A chest X-ray scan of a COVID-19 negative patient.



(d) Saliency map for the X-ray shown on the left (c).

Figure 1.2: On the left, we have the chest X-ray scans of two patients, one who tested positive for COVID-19 (top), and one who tested negative (bottom). The orientation marker 'R' is only present to indicate the patient's right side on the X-ray. On the right, we see the corresponding saliency maps for a classifier trained to predict the COVID test result from the X-ray, with the darkest red spots being the most important. We clearly see that in both cases, the orientation marker 'R', (as highlighted with a black arrow on the saliency maps) is influential in predicting the test result. Figures adapted from DeGrave et al. [27].

networks better. Arising from this need, there has been increased interest in work on interpretability methods to explain the behavior of DNNs.

1.2 A representational hierarchy for vision and audio

1.2.1 Vision

Interpretability studies in vision have demonstrated that convolutional neural networks (CNNs) develop structured, human-interpretable representations that mirror a compositional hierarchy. Research into the internal activations of these networks reveals that early layers form simple feature detectors (see Fig. 1.3), sensitive to low-level primitives such as edges and curves [79]. As data progresses through the network, higher layers encode increasingly complex concepts, such as shapes and textures [58].

Strikingly, this emergence of structure often occurs without explicit supervision for those specific concepts. For example, CNNs trained strictly for scene classification, *i.e*. identifying whether an image depicts a bedroom, a hotel lobby, or a theater, have been found to spontaneously develop 'object detector' neurons [81]. A neuron might learn to activate strongly only in the presence of a bed, despite the model never being told what a 'bed' is. The model implicitly learns that a 'bedroom' is a scene composed of objects like beds and lamps, as shown in Fig. 1.4.

Quite notably, this computational hierarchy mirrors the hierarchical processing observed in human vision. Once the information of a visual scene travels from the retina of the observer and through the optic nerve, it eventually reaches the primary visual cortex. From there, processing splits into dorsal and ventral pathways (see Fig. 1.5), dual pathways that are thought to specialize to information about where something is (localization) and what it actually is (recognition and semantics) [57]. Neurons in early areas respond to features relevant to object identification (color, shape), with receptive fields (*i.e*. the region of visual space a neuron responds to) and feature selectivity (*i.e*. the specific visual attributes a neuron is tuned to) progressively increasing along the pathway [34].

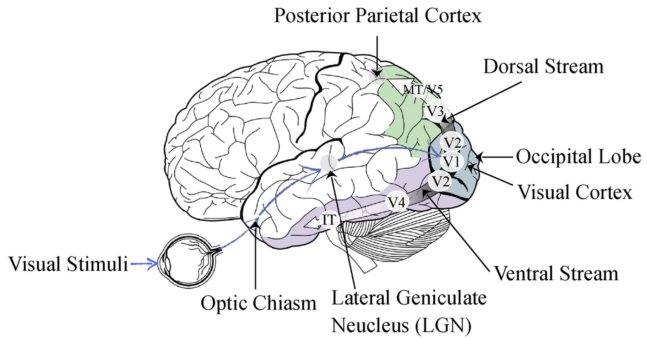
A strikingly similar compositional logic is visible in deep networks. As shown in Fig. 1.3, the earliest CNN layers respond to simple edges and color contrasts; intermediate layers assemble these into curves, textures, and recurring part-like motifs reminiscent of geons; while the deepest layers activate for whole objects such as faces and animals. Prominent theoretical frameworks in human vision are also consistent with this compositional view. Recognition by Components (RBC) [12] proposes that we perceive objects by parsing them into volumetric primitives called "geons". A mug, for instance, is recognized as a cylinder joined to a toroidal handle (see Fig. 1.6).

Human visual processing also involves top-down influences such as feedback connections, attentional modulation, and prior knowledge, but these do not diminish the substantial evidence for hierarchical processing and the important role of compositionality in vision.

Thus, prior work suggests that both deep vision models, much like the human visual system, learn compositional hierarchies: they assemble low-level inputs into intermediate concepts, which in turn inform high-level semantic decisions.

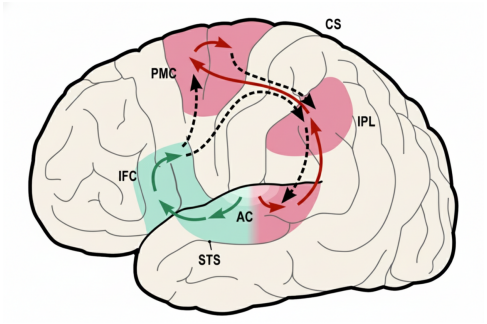
Figure 1.3: Visualization of features learned by a convolutional neural network. Early layers detect simple edges (top left), while deeper layers assemble these into complex geometries and object parts (middle, bottom). Note how these features naturally form a hierarchy of composition. Figure from Zeiler and Fergus [79].

Figure 1.4: Individual neurons from a convolutional neural network trained for scene classification (*i.e.* identifying whether an image is of a bedroom, a movie theater, a zoo *etc.*), often act as object detectors. These neurons learn to fire with high precision for instances of the same object, contributing to the understanding of a larger scene. This phenomenon is observed even with no explicit training for object labels. Figure from Zhou et al. [81].



(a) Dorsal and ventral pathways for visual processing in the human brain. Figure from Zareh et al. [78]. [a]

[a] Sections V2 through V5 are typically referred to by these names, with V simply standing in for 'visual area'.



(b) Dorsal and ventral pathways for auditory processing in the human brain. Figure from Rauschecker [63]. [a]

[a] AC: auditory cortex; STS: superior temporal sulcus; IFC: inferior frontal cortex; PMC: premotor cortex; IPL: inferior parietal lobule; CS: central sulcus.

Figure 1.5: In the human brain, the processing of visual information is supported by two pathways: the *dorsal* or the 'where' pathway, and the *ventral* or 'what' pathway. The processing of auditory information is analogously supported by its own distinct dorsal and ventral pathways. Though these pathways are not completely isolated from each other, each one specializes for its function.

1.2.2 Audio

We may then be tempted to think that such insights can readily carry over to other modalities like that of audio, but we must consider carefully. While the domains of vision and audio share underlying principles of pattern recognition, they differ fundamentally in their input modalities. Vision is primarily spatial, whereas audio is intrinsically spectro-temporal, and the "primitives" of vision do not necessarily translate directly to audio [49]. Biederman suggests that phonemes might serve as auditory primitives analogous to geons [12], but speech represents only a fraction of the auditory experience. We experience audio over multiple temporal scales, with a hierarchy defined by temporal scale as much as spectral content. As noted in Griffiths and Warren [41], the identity of an auditory object depends on the level of analysis. Individual notes or onsets can be our focus, but over wider temporal windows, they cohere into motifs, phrases, and stylistic patterns, as illustrated in Fig. 1.7. Given these differences, seeing a hierarchical structure emerge in vision models does not guarantee the same will occur in audio models. This raises the natural question: *Is there an analogous hierarchy in audio, and do we see evidence for it in DNNs trained for audio tasks?*

Understanding whether such hierarchies emerge in audio models is particularly compelling because human auditory perception itself is largely understood to be hierarchically organized. Sound processing in the brain begins at the cochlea, which creates a spectro-temporal representation [66], and travels to the primary auditory cortex. The cochlea is known to have a *tonotopic map*, *i.e.* areas at various depths within the cochlea specialize and respond to specific frequency regions, as shown in Fig. 1.8. From

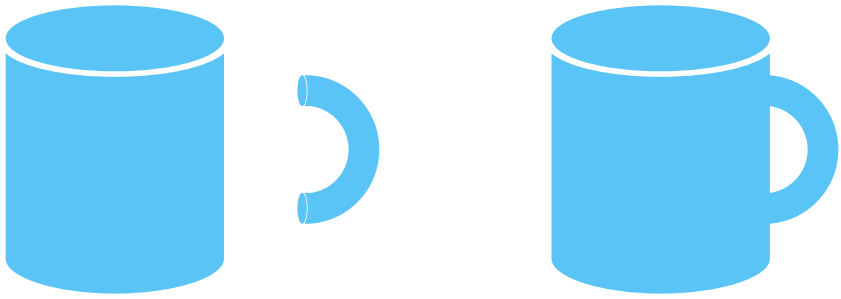


Figure 1.6: The Recognition by Components (RBC) framework [12] suggests that visual objects are recognized by parsing them into component parts, or "geons". This includes objects like a cylinder, and a toroidal segment (or a curved cylinder) as shown on the left, combining to form a mug as shown on the right.

there, processing splits into dorsal and ventral pathways, dual pathways that are thought to specialize to information about where the sound is (localization) and what it actually is (recognition and semantics), much like the analogous pathways for visual processing. The various regions of the brain present in these pathways are shown in Fig. 1.5b. fMRI studies have shown that the ventral pathway implements a progression from acoustic features to perceptual and ultimately object- or category-level representations [13, 18, 45]. Thus, there is evidence of hierarchical processing for auditory information in the human brain.

As we discussed earlier, the perceptual 'primitives' of audio are not as immediately apparent as those for vision. In the visual domain, geons provide a concrete vocabulary of elemental parts from which objects are composed; no equally established set of primitives exists for general audio, though the question has received careful consideration in prior work. The principles of Auditory Scene Analysis (ASA) [15] offer a useful framework for this purpose. ASA describes how 'auditory streams' are formed, *i.e.* distinct sequences of sounds that are perceptually grouped together and maintain a sense of identity over time. Just as geons combine through spatial configuration to form visual objects, auditory elements cohere into streams based on principles such as timbre, spectral proximity, and harmonicity. A series of sounds sharing the same timbre (such as that of a saxophone) would be perceived as one stream. Those with dissimilar timbre separate into different auditory streams, even despite any similarities in pitch (see Fig. 1.9). As multiple auditory streams form and interact, they give rise to higher-level percepts such as musical style or genre. Our understanding of a higher-level scene can then be attributed to the integration of these streams, exhibiting a compositional hierarchy of audio perception analogous to the one described for vision.

If human hearing relies on building up from these streams and temporal patterns to understand a scene, it is plausible to hypothesize that effective deep learning models implicitly learn a similar strategy.

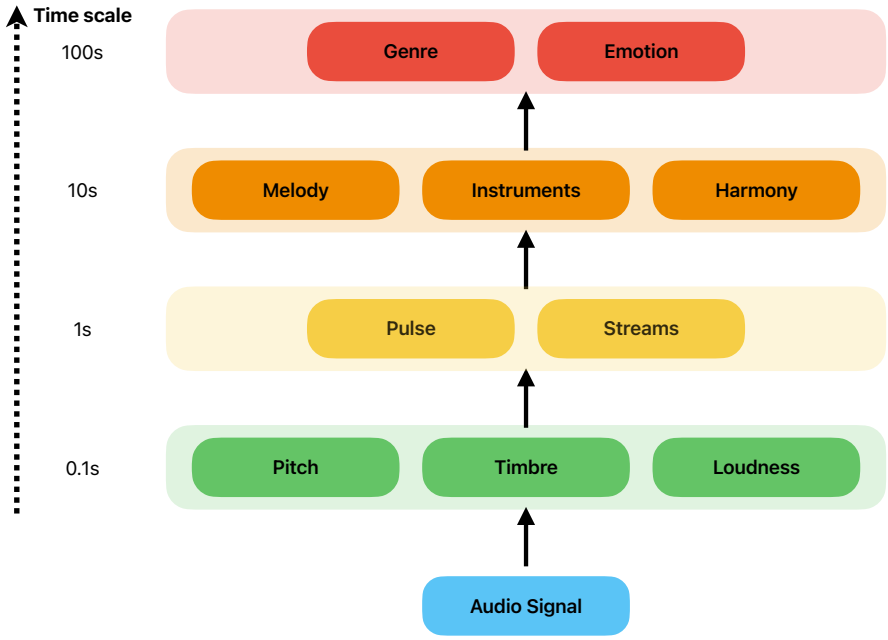


Figure 1.7: Conceptual illustration of temporal integration and hierarchical auditory processing for music. Low-level acoustic features (pitch, timbre) integrate over time to form intermediate objects (melody, instruments), which further combine to form high-level semantic concepts (musical style or genre). We only enlist concepts here directly relevant to our discussion; this is not an exhaustive list.

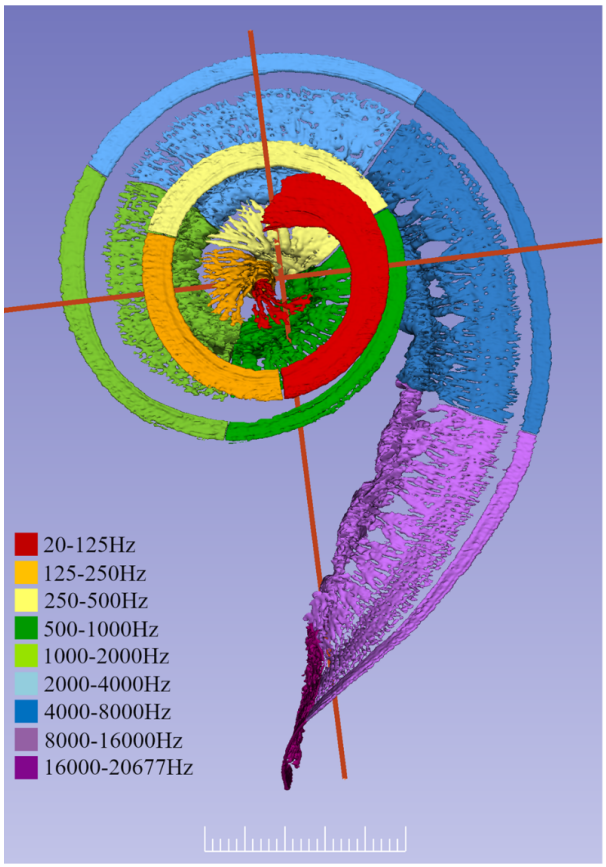


Figure 1.8: An image showing the tonotopic map of the human cochlea. The different regions of the cochlea respond to specific frequency regions, as shown with different colors in the image. Figure from Li et al. [51].

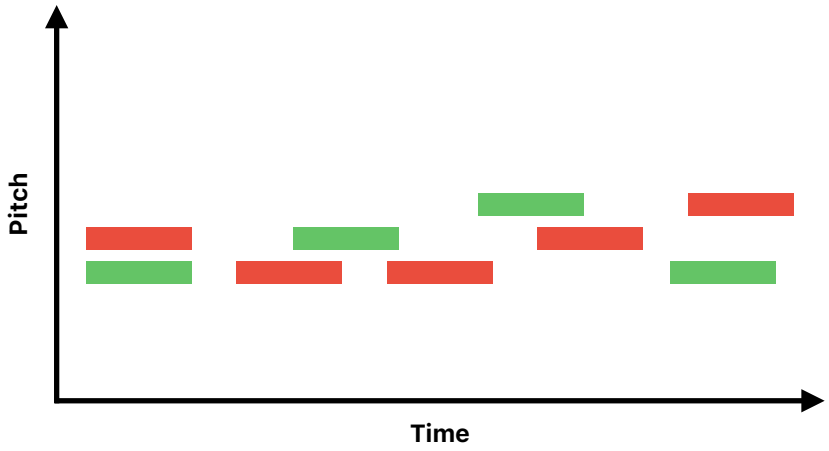


Figure 1.9: Sequences of notes having the same timbre are perceived as belonging to the same auditory stream. Differing timbres are perceived as separating the auditory streams (shown here in two different colors), despite similarities in pitch.

In language, prior work has found evidence of hierarchical processing, with syntactic information present in earlier layers and semantic information in later layers [62, 72]. In vision, the hierarchy from edges to objects is well-documented [79, 81]. However, the internal organization of DNNs trained for general audio tasks remains less explored.

Existing interpretability research in audio has primarily examined local explanations [3, 60] or produced post-hoc rationales for individual decisions [80], leaving open the broader question of whether systematic, concept-level structures emerge in these models. While we know audio models perform well, we lack a comprehensive understanding of whether they learn to disentangle low-level acoustics from higher-order constructs like instrument identity, rhythm, or harmony, and whether these concepts emerge in a specific order. A particularly unexplored question concerns whether instrument concepts emerge within the internals of genre classification models, and whether manipulating these concepts causally affects their predictions.

1.3 Thesis Objectives

The primary objective of this thesis is to investigate the internal representations of deep neural networks trained for audio tasks to determine if they exhibit a human-like hierarchical organization.

1.3.1 Hypothesis

We hypothesize that DNNs trained on high-level audio classification tasks (such as genre recognition) spontaneously learn a compositional hierarchy, where lower layers encode low-level acoustic features and intermediate layers encode "auditory objects" (such as specific instruments) that support the final high-level decision.

1.3.2 Key Contributions

In this thesis, we make the following key contributions -

- We provide a comprehensive layer-wise analysis of multiple CNN architectures trained for general audio classification via different strategies, demonstrating a consistent trend where low-level concepts are resolved in early layers and high-level semantic concepts in deeper layers.
- We extend this analysis to Transformer-based architectures, specifically an audio spectrogram transformer (AST), showing that despite the global attention mechanism, a similar hierarchical emergence of concepts occurs.
- We investigate the specific relationship between musical instruments and musical genres in an AST trained solely for genre classification. By training linear probes on intermediate layers, we show that instrument concepts emerge in genre classifiers without explicit supervision.
- We employ causal intervention techniques (steering vectors) to demonstrate that these emergent instrument concepts are not just present but are causally relevant to the model's genre classification logic.

To our knowledge, this is the first time such a comprehensive study on concept probing in the intermediate layers of DNNs trained for audio tasks has been performed, and the first time the lower-level concept of instruments has been tested for its causal relevance in the classification of the higher-level concept of genre through an intervention strategy.

1.4 Thesis Organization

The remainder of this thesis is organized as follows: **Chapter 2** details our probing experiments on convolutional neural networks (CNNs), analyzing the layer-wise performance of models trained on various audio tasks to establish the existence of a hierarchy. **Chapter 3** focuses on the emergent relationship between instruments and genres. We utilize the audio spectrogram transformer (AST) to investigate whether instrument concepts arise naturally in genre classifiers and use steering vectors to test their causal validity. Finally, **Chapter 4** summarizes our findings, discusses the limitations of our current approach, and outlines directions for future research.

Chapter 2

Probing deep neural networks for auditory task performance

2.1 Prior work in interpretability of deep neural networks for audio

Deep audio networks can be classified into two categories based on their input types: raw audio waveforms or spectrograms. Among the former, interpretability has been studied specifically for speech recognition models pre-trained in a self-supervised manner (*e.g*. Wav2vec 2.0 [8]). Notable contributions from Pasad et al. [61] and Choi et al. [23] conduct layer-wise analyses of DNNs, tracing the evolution of acoustic and linguistic information, and revealing how phonetic and semantic content is progressively organized for individual words across network depth.

For spectrogram-based models, the raw audio waveform is first converted into a (log-Mel) spectrogram, and popular 2D CNN architectures are adopted. In fact, CNNs trained on audio spectrograms have shown state-of-the-art results on auditory tasks (*e.g*. contrastively trained CLAP [31] or the efficient MobileNet architecture [65]).

A key early technique in interpretability of CNNs for vision was *deconvolution*. By sequentially *'inverting'* the operations of layers, we can generate a synthetic input image, allowing us to visualize the model's internal processing at a given layer [79]. In Choi et al. [21], the authors use this technique on spectrogram-based auditory CNN models and further *auralize* the resultant spectrogram. This allows us to hear the processed version of the input audio at a given layer in the model. Using this technique, the authors demonstrate the presence of onset detectors in earlier layers, and more complex detectors of percussive patterns and harmonic textures in later layers. While this hints towards auditory CNNs modeling the hierarchy in audio, it is presented for a single model trained on a specific downstream task on music.

More recently, similar work for Transformer models in the audio domain has emerged as well, offering new frameworks which can generate intuitive explanations have emerged, including listenable interpretation methods for audio classifiers [59, 60]. These approaches generate sample-level (or *local*) explanations by highlighting salient spectrogram regions, demonstrating clear value for improving model transparency and trustworthiness in specific instances. The focus of these works lies more on the technical novelty, and the discussion on the insight gained into the model itself from the application of these techniques is limited.

Despite these advances, such methods face significant scalability challenges: they require careful per-instance analysis, and provide limited insight into the global organizational principles of the network. Additionally, existing approaches remain limited in scope, confined either to a single domain (e.g., music). They fail to reveal whether the hierarchical organization hinted at in prior work holds across different types of sounds and across tasks requiring varying levels of abstraction.

A concrete demonstration of such a hierarchical organization requires an approach that can systematically probe representations across tasks that explicitly target different levels of auditory complexity from low-level acoustic features to high-level semantic concepts, and ask where in the network information relevant to each task resides.

2.2 Probing intermediate layers in audio CNNs

For our work, we aim to show that the intermediate layers of an audio CNN trained for general audio tasks implicitly learn representations that are well suited for different tasks. Additionally, we assess intermediate layer representations on a variety of tasks and note if the representations at different layers are better suited to different tasks.

Representations from intermediate layers may be evaluated on downstream tasks by training additional linear classifiers [4] or using k-Nearest Neighbor (kNN). We perform kNN probing with Euclidean distance, as it is a parameter-free approach that alleviates some challenges with learned probes [11], and allows us to work with small datasets of $\sim$1000-2000 instances.

2.2.1 Tasks and Datasets

We select a total of six tasks (and datasets) for our experiments, three each from the domains of music and speech. These tasks are chosen to cover the range of sounds from low- to high-level in auditory processing. This structure enables us to examine how representations behave as task complexity increases and abstraction deepens. For each level, we choose benchmark datasets from prior literature that are publicly available. Since kNN classification is susceptible to class imbalance, a class-balanced subset of the dataset is considered. Table 2.2 summarizes the dataset size, number of classes, and audio duration.

Tasks in music.

1. *Note name classification*, also known as *chroma classification*, involves correctly classifying a single note into one of 12 pitch classes commonly used in Western music without differentiating between octaves, and is considered a low-level task. We use the NSynth [32] dataset, which contains recordings of single notes played on acoustic, electronic, and synthetic instruments.

2. *Instrument recognition* is considered a mid-level to high-level task. We use the Medley-solos-DB dataset [55], which contains short 3 s excerpts of monophonic instrumental sounds extracted from various songs.

3. *Genre classification* is considered a high-level task, for which we use the GTZAN [75] dataset, consisting of 30 s clips of audio from 10 popular genres.

Tasks in speech.

1. *Consonant classification* is our low- to mid-level task and is evaluated on the PCVC [56] dataset, which consists of 23 Persian consonants spoken with different vowels by various speakers.

2. *Keyword recognition* is a mid-level task. We evaluate on the Speech Commands dataset [77], consisting of 35 class of single word utterances.

3. Finally, *speaker count estimation* is a high-level task, for which we use the LibriCount [70] dataset that consists of 5 s instances of audio with up to 11 distinct speakers speaking within the duration.

2.2.2 Experimental setup

CNN Models

We analyze layer-wise properties of three CNN models trained via different strategies. They include direct supervised training (VGGish), contrastive audio-language pre-training (CLAP), and training via knowledge distillation from complex Transformer models (MobileNetV3). Irrespective of the training strategy, we observe similar trends hinting towards CNNs exposing underlying hierarchies.

VGGish [42] is based on the principles of the VGG CNN [68] (3×3 convolutions), and is pre-trained on the YouTube-100M dataset. We use a PyTorch port of VGGish [71].

CLAP [31] is initialized with pre-trained text and audio encoders. Representations from the two encoders are passed through additional projection layers and the whole model is trained via the contrastive audio-text matching loss on audio-caption pairs. CLAP's audio encoder is based on the CNN14 architecture [46] and pre-trained on AudioSet [38].

MobileNetV3 [43] is an efficient CNN architecture that uses depth-wise separable convolutions and squeeze-and-excite layers. Schmid et al. [65] train the CNN through knowledge distillation from a complex audio transformer teacher on AudioSet (PaSST [47]), which we utilize as our third model. For our experiments, we use the `mn40_as` checkpoint provided by Schmid et al..

Layer	VGGish	CLAP	MobileNetV3
conv1	features.1	conv_block1	features.1
conv2	features.4	conv_block2	features.4
conv3	features.7	conv_block3	features.7
conv4	features.9	conv_block4	features.10
conv5	features.12	conv_block5	features.13
conv6	features.14	conv_block6	features.16
fc1	embeddings.1	fc1	classifier.1

Table 2.1: Specific layers at which the intermediate representations are extracted for each model and the corresponding layer name used in Fig. 2.1. The layers are selected to be equally spread from the model input up to the first fully-connected layer. For CLAP, the prefix `audio_encoder.base` is omitted for brevity.

Extracting Layer-wise Intermediate Representations

For an input audio $\mathbf{x} \in \mathbb{R}^{sr}$, with duration s and sampling rate r, we follow the preprocessing as prescribed for each model to compute a log-Mel spectrogram $\mathbf{s} \in \mathbb{R}^{F \times T}$ with F Mel frequency bands and T time steps. For each layer l, we extract the output from the partial encoder Φ_l to obtain $\hat{\mathbf{h}}_l \in \mathbb{R}^{c_l \times t_l \times f_l}$, a feature map with c_l channels, t_l width (time), and f_l height (frequency). We flatten $\hat{\mathbf{h}}_l$ to obtain $\mathbf{h}_l \in \mathbb{R}^{d_l}$, where $d_l = c_l t_l f_l$. If l is a fully-connected layer, the flattening procedure is not necessary.

We extract intermediate representations $\mathbf{h}_l = \Phi_l(\mathbf{x})$ at six equally spaced convolutional blocks (`conv1-6`), and the first fully-connected layer in the model (`fc1`). For reproducibility, the precise layer names from which we extract features are indicated in Table 2.1.

Further implementational details

We adopt a five-fold stratified cross-validation setup for all our experiments. Specifically, each fold is also constructed to be class-balanced. We present some additional details.

GTZAN audio duration. While the original audio files are $30\,\mathrm{s}$ long, we crop and use the middle $5\,\mathrm{s}$ as we expect the central portion to clearly indicate the genre.

Class-balancing datasets. We report experiments on all datasets after class-balancing. Additionally, Medley-solos-DB contains excerpts of instrumental solos from various songs, with multiple excerpts belonging to the same song. To prevent spurious correlations from affecting results we need to ensure that all excerpts from the same song lie in the same split. This results in approximately balanced classes. Furthermore, the class 'tenor saxophone' has very limited samples and is discarded.

VGGish processing details. While CLAP and MobileNet are able to process audio inputs of $5\,\mathrm{s}$, VGGish requires inputs of $960\,\mathrm{ms}$. We split longer audio files longer into non-overlapping chunks and compute $\mathbf{h}_l^c$, where the superscript denotes the chunk id. If an audio has more than one chunk, the last one is ignored (as it is often small, *e.g.* $40\,\mathrm{ms}$ for audio of $1\,\mathrm{s}$). The other chunks are concatenated in our experiments.

2.2.3 Results and discussion

Layer-wise Performance

The performance of layer-wise representations across all six tasks is presented in Fig. 2.1.

Low-level tasks. On *note name classification* (row 1 left), we observe that representations from the first layer outperform later layers across all 3 models. While models are never trained with such category labels, early layers implicitly learn to distinguish between them. Similar observations hold for *consonant classification* (row 1 right) where both CLAP and MobileNetV3 peak at `conv3`.

Mid-level tasks. *Keyword recognition* (row 2 right) shows peak performance in intermediate layers (`conv4`) for CLAP and MobileNetV3. Interestingly, both early and later layers perform worse. On *instrument recognition* (row 2 left), we see a large performance improvement of 20-40% for the intermediate layers (`conv4` for CLAP, `conv2` for MobileNetV3). However, as the AudioSet dataset has instruments as categories, unlike keyword recognition, later layers preserve this information and highest performance is seen at `fc1`.

High-level tasks. On *genre classification* (row 3 left), we see performance peak in the later layers for all models. This is expected as AudioSet contains music genre classes, but the consistent performance improvement across the layers (especially for CLAP) is indicative of more complex integration enabled by the network with increasing depth. Differently, even though models are not trained on the task of *speaker count estimation* (row 3 right), we see that peak performance on this complex task is achieved in the later layers with early layers performing close to random.

In summary we observe that auditory CNNs (especially CLAP, MobileNetV3) show better performance on low-level tasks in early layers and on high-level tasks in later layers. This hints that the model is able to learn the underlying hierarchy of audio tasks.

VGGish shows trends different from CLAP or MobileNetV3, especially on the intermediate and high-level tasks. We attribute this to the simple architecture where the final convolutional map is flattened to a high-dimensional representation and the input to the `fc1` layer is a vector in $\mathbb{R}^{12,288}$. In fact, the close-to-random performance of early layers on high-level tasks indicates that the final projection (MLP) may be doing much of the heavy-lifting.

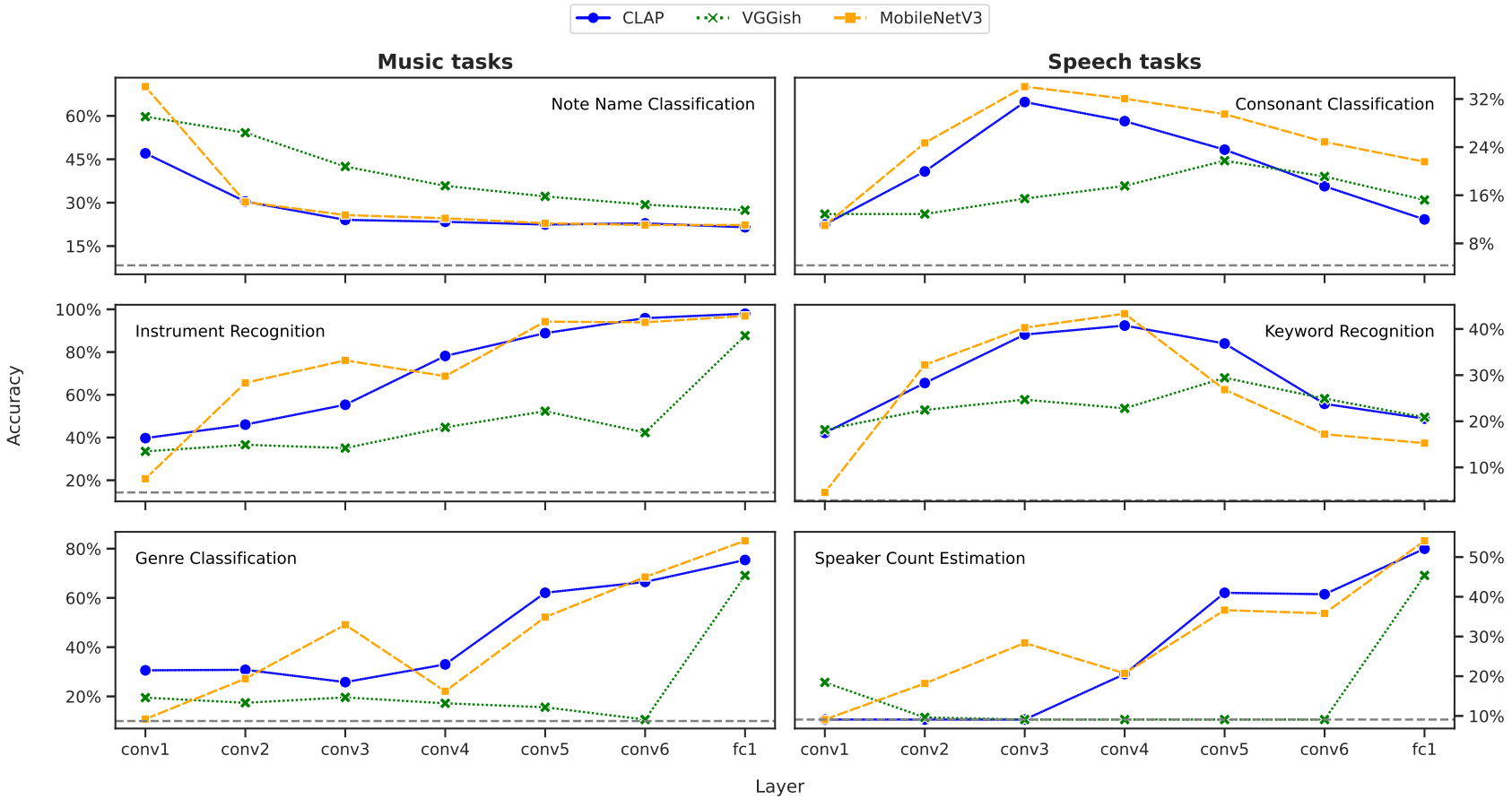


Figure 2.1: Accuracy using intermediate representations extracted from convolutional layers of increasing depth. Gray dashed line represents chance accuracy. Tasks are ordered in increasing abstraction for music (left) and speech (right). We observe that low-level tasks (row 1) are likely to perform well at shallow layers while semantic high-level tasks (row 3) perform well at deeper layers.

Dataset	Type	#Instances	#C	Dur. (s)
NSynth [32]	🎵	1800*	12	4
Medley-solos-DB [55]	🎵	965*	7*	3
GTZAN [75]	🎵	1000	10	5
PCVC [56]	☺	1794	23	2
Speech Commands [77]	☺	1750*	35	1
LibriCount [70]	☺	1100*	11	5

Table 2.2: Datasets used in our work from music (🎵) and human speech (☺) domains. #C is the number of classes and Dur. (s) is the typical audio duration in seconds. *The dataset is modified slightly to be class-balanced when possible.

Ablations

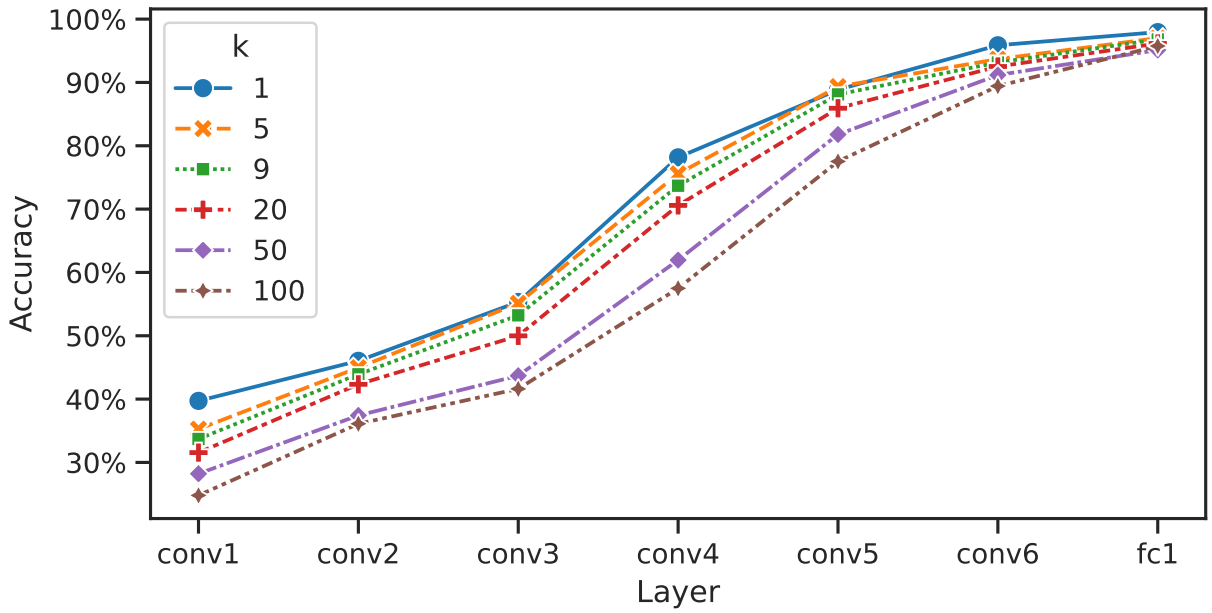


Figure 2.2: Effect of changing the value of k for CLAP on the Medley-solos-DB dataset. In general, changing k does not have a major effect on the overall layer-wise trend, across all models and datasets.

Effect of k in kNN. We vary the value of k in kNN from $\{1, 2, \ldots, 9, 20, 50, 100\}$ assuming each dataset class has at least k samples in the training set. Comparing results for various values of k, Fig. 2.2 shows that the general trends are largely independent of k. Thus, we select the k-value with the best peak accuracy as representative of the layer-wise trends.

Do varying durations matter? Our datasets have differing audio durations (Table 2.2). While this may act as a confounder to layer-wise effects discussed in the previous section, we rule out this possi-

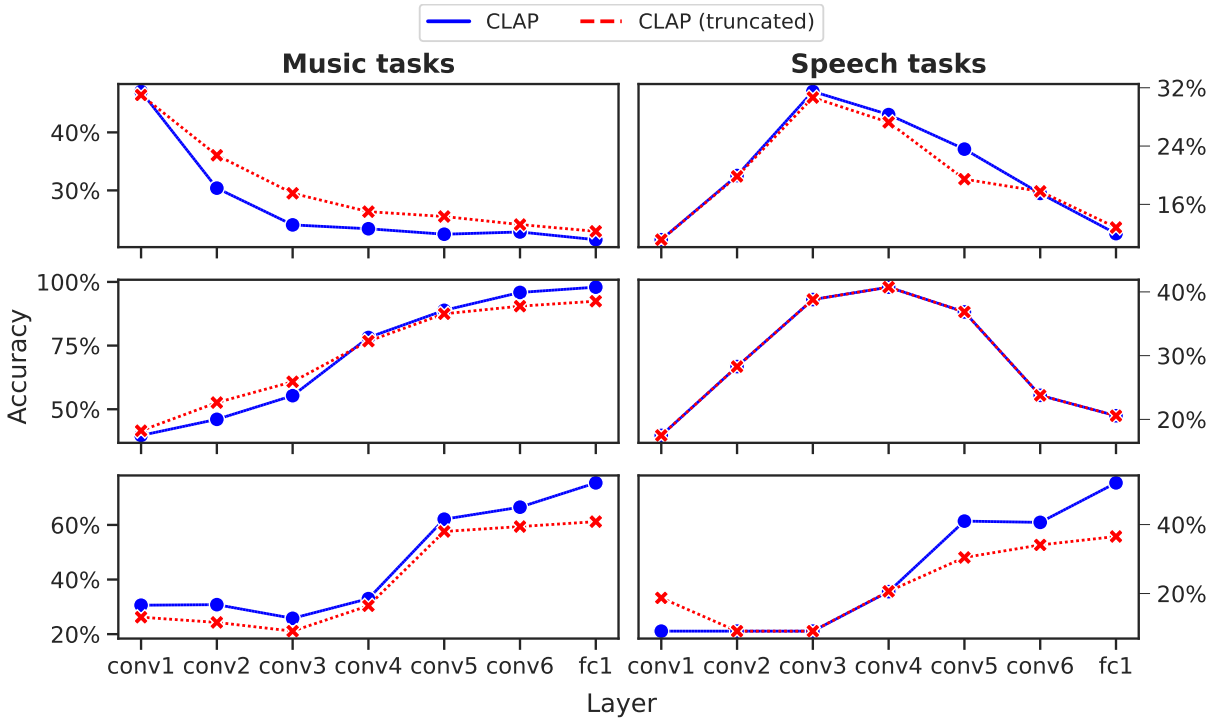


Figure 2.3: Effect of truncating all input audio files to 1 s on CLAP. The high-level tasks display an earlier saturation, but the general trends remain. Similar results are observed for VGGish and MobileNetV3 as well. The six tasks are presented in the same order as Fig. 2.1.

bility by conducting experiments on all tasks with audio files truncated to 1 s. In Fig. 2.3, we see that the performance difference before and after truncation is minimal. The differences are most evident in the later layers of high-level tasks that seem to experience a saturation with shorter audio clips.

2.2.4 Limitations

We note some limitations with our approach, due to the architecture of CNNs and issues with distances in higher-dimensions.

CNNs are built on the principle of aggregating information such that early feature maps have units whose receptive fields are small with respect to the entire image, and subsequent feature maps are generated further from these further maps. As a result subsequent feature maps are able to aggregate information from larger and larger receptive fields. In the case of a CNN operating on a spectrogram, this means integration over a larger spectral and temporal range, which naturally lends itself to better performance on high-level tasks which often require longer temporal ranges to perform.

In higher dimensions, Euclidean distances between points can become less informative, reducing their discriminative value [1]. Additionally, our feature maps have differing dimensionalities at different layers. We note that our k-Nearest Neighbor probes show good performance on many tasks neverthe-

less. This displays that representations and the distances between them retain sufficient discriminatory information, despite these challenges.

2.3 Probing intermediate layers in AST

We address the issue of varying receptive fields in audio CNNs by probing another architecture which is not affected by this limitation, *i.e*. the audio spectrogram transformer (AST).

2.3.1 Audio Spectrogram Transformer (AST)

The **audio spectrogram transformer (AST)** is a model architecture introduced in Gong et al. [40], based on the popular vision transformer architecture [30]. The vision transformer architecture takes an input image and splits it into patches of size 16 pixels $\times$ 16 pixels. After pre-processing, these tokens are converted to embeddings by passing through an multi-layer perceptron (MLP) and then fed as an input sequence to a Transformer encoder. Similar to BERT [29], a special `[CLS]` token is also a part of the sequence, which is used as the input for training the model on a downstream classification task.

In the case of AST specifically, training task is general audio classification on AudioSet [38]. An input audio waveform of t seconds is first converted into a log-Mel spectrogram. This spectrogram is computed with 128-dimensional log-Mel filterbanks, and with $25\,\mathrm{ms}$ Hamming windows $10\,\mathrm{ms}$ apart, resulting in a $128 \times 100t$ log-mel spectrogram as input to the AST. This is then split into a sequence of N 16×16 patches with an overlap of 6 for both axes (time and frequency). Each patch is flattened and passed through an MLP to result in a 768-dimensional embedding, to which positional embeddings are added. The Transformer encoder used in AST embedding dimension of 768, 12 layers, and 12 heads.

AST further utilizes transfer learning by initializing its weights with those of a data-efficient image transformer (DeIT) [73], which was trained on ImageNet [48]. Using a strategy involving cutting and then using bi-linear interpolation on the positional embeddings already trained from ImageNet, these positional embeddings are adapted for use with AST. DeIT also has two `[CLS]` tokens; in AST, these are averaged as a single $[CLS]$ token and then passed to a new classification head for the audio pre-training.

2.3.2 Experimental setup

We perform probing experiments on AST with the same six tasks and datasets specified in 2.2. We follow the pre-processing steps as prescribed in Gong et al. [40], and then extract the `[CLS]` token at various layers, using k-Nearest Neighbor probes with Euclidean distance on these tokens to perform these experiments. We utilize the same dataset splits previously used for the audio CNN experiments.

2.3.3 Results and discussion

The performance of layer-wise representations across all six classification tasks is presented in Fig. 2.4.

Low-level tasks On *note name classification* (row 1 left), we note that the best performance is achieved using representations from the first layer, and the performance steadily decreases as we go to deeper layers. For *consonant classification* (row 1 right), peak performance is close to the middle at layer 6, continues to dip at layers 8 and 9 before increasing once again at the final layers.

Mid-level tasks On *instrument recognition* (row 2 left), the classification accuracy when using representations from the middle layers is high, near the peak accuracy which we observe at the last two layers. There is also a small dip in performance at layers 8 and 9. The performance at different layers for *keyword recognition* follows a pattern nearly identical to that of *consonant classification*, with a peak in performance at layer 6, a dip at layers 8 and 9, and an increase at the final layers.

High-level tasks For *genre classification*, classification accuracy peaks at the final layers, but reaches high performance at the middle layers. Here too, there is a dip in performance at layers 8 and 9. Surprisingly, we see little difference between early and later layers for *speaker count estimation*. The performance remains largely unchanged at most layers, except a small but notable dip at layers 8 and 9.

Overall, the results are not as conclusive or convincing as what we see for the audio CNNs. The line plots for consonant classification and keyword recognition are very similar, and we observe no notable trend at all for speaker count estimation. The fact that we cannot as clearly observe the trends for AST as we had for audio CNNs, implies that the architectural design of CNNs likely played a notable role in the results we had seen previously. CNNs have a more explicit integration of elements across the two axes of the image (in this case frequency and time in a spectro-temporal image), leading to a more direct effect on a specific layer's ability to discriminate between auditory objects.

The classification accuracies for all tasks are well above their chance accuracies, implying that the pre-training still resulted in representations useful for all six tasks that we assess the model on. The dataset used for pre-training (AudioSet), contains no annotated speech data, but does however contain instrument and genre labels.

Additionally, we clearly see a pattern of a dip in performance for most tasks at layers 8 and 9. Although potentially informative, a thorough exploration of this effect would require additional data and methodological extensions that fall outside the scope of this thesis.

2.3.4 Limitations

We note the limitations of our approach due to the nature of the training tasks. The tasks of *genre classification* and *instrument recognition* are already overlapping with the labels present in AudioSet, which causes the later layers of our models to preserve a strong performance on these tasks.

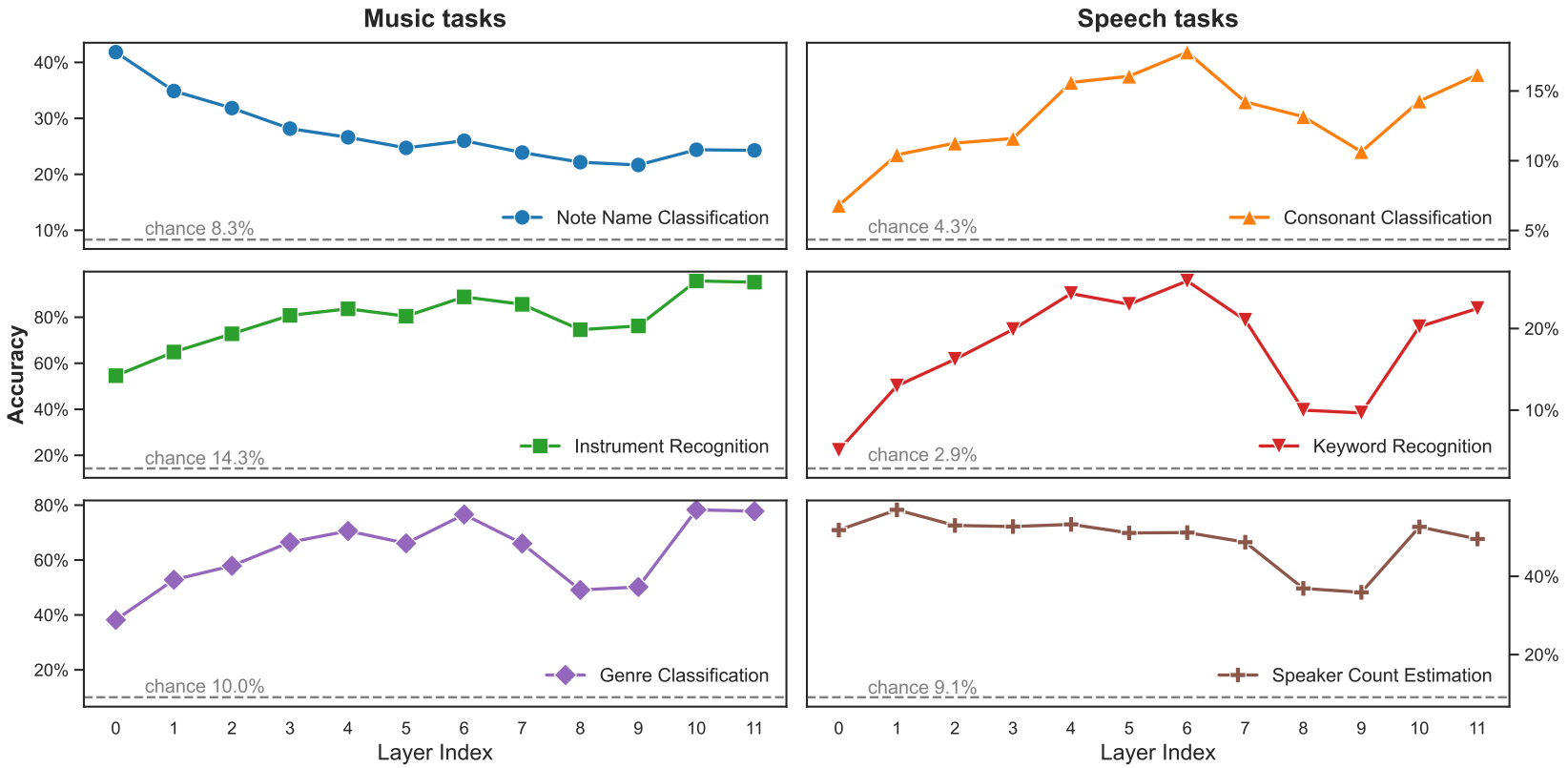


Figure 2.4: Classification accuracy for representations extracted at different layers of AST. We see a less pronounced effect here. The note name classification task peaks at the earliest layers with performance dropping over time, and the low- and mid-level speech tasks also show the best classification accuracy for representations from the middle layers. Notably, the performance on speaker count estimation is comparable regardless of what layer the representation is extracted from. Results shown for k-Nearest Neighbor classification with k which gives maximum peak classification accuracy.

In order to confidently state there is a hierarchical relationship between concepts at different levels in an AST, we therefore need more evidence with another approach. A model trained on *both* instrument and genre labels then is not perhaps the most suitable to demonstrate the relationship between these concepts. Consider however, if we train an AST model on a single higher-level concept. For this higher-level concept, if there is a lower-level concept with a hierarchical and causal relationship, we should be able to observe it as an emergent concept in intermediate layers of the model. The concepts of **instrument** and **genre** are a natural choice for the lower-level and higher-level concepts.

Chapter 3

Instruments as emergent and causally relevant concepts for genre classification

3.1 Musical genre

Musical genre refers to the set of labels used to identify a piece of music as belonging to some shared tradition or set of conventions. One of the main utilities of the notion of a musical genre lies in its ability to convey *expectations* of the nature of a piece of music, and historically served as a practical tool for marketing and selling recordings in the pre-streaming era [7].

At the same time, musical genre is a slippery concept informed by multiple factors. It is not determined solely by the auditory content of an excerpt, but also by cultural context, geographic origin, lyrical content, *etc.* [14]. In our work, the terms *musical genre* and *musical style* are used interchangeably; we focus on genre-characteristic elements present in the audio signal itself, excluding extra-musical factors.

Musical genre classification is a task with some subjectivity: what may be termed as a disco song by some, would be called a techno song by others. Nonetheless, there is often consensus between human listeners on what genre a specific excerpt should be labeled with [26]. In Lippens et al. [52], the authors demonstrate that for a six-genre classification task (*pop, rock, classical, dance, rap* and *other*) with 160 tracks, an average of 20.6 participants out of 27 total voted for the same genre on a given track.

In Gjerdingen and Perrott [39], participants in a genre classification study were able to achieve above chance classification with excerpts as short as $250\,\mathrm{ms}$. This is a duration too short to ascertain melodic or rhythmic motifs, and only allows for the timbre of the excerpt to be perceived clearly. *Timbre* is the same spectro-temporal property of sound which allows us to distinguish one instrument from the other, even when they play the same note at the same amplitude. With multiple instruments, a *polyphonic timbre* emerges [35], which for different genres can form the 'characteristic sound of a genre'. In Tzanetakis et al. [75] as well, a set of feature vectors containing *'musical surface'* information (timbre-related and instrument-related features such as spectral mean centroid and mean rolloff) were found to work nearly as well for genre classification as a larger set including additional rhythm features as well, with a difference of less than 10% between the two sets.

Instrumentation is one of the most salient perceptual cues in these categorizations. Music cognition research highlights the centrality of timbre and instrumentation in genre perception, with both behavioral

and neurocognitive studies showing that listeners rely on instrument cues when categorizing music [6, 16, 35, 69].

Automatic music genre recognition systems have received considerable attention from the research community [69, 75]. More recently, deep neural networks have shown remarkable performance in music genre classification [22, 31, 74]. However, such models are opaque, and their internal mechanisms have not received similar attention. As a result, we lack an understanding of what properties of audio these models are sensitive to, and whether their internal mechanisms are at all similar to how humans perceive sound and classify musical genres.

Previous work aiming to better understand deep neural networks performing music classification is limited. Early work by Choi et al. [21] inspects a CNN trained for genre classification on audio spectrograms and *auralizes* the processing at various intermediate layers for a given input. The authors find interpretable filters are learned: early filters become effective onset detectors and filters in later layers activate for more specific patterns of percussion or harmony, showing some evidence of composable patterns learned for genre classification. However, they only consider four music pieces in their investigation. More recent work from Parekh et al. [60] and Paissan et al. [59] allow for the creation of *'listenable explanations'*, *i.e.* audio clips derived from the input audio which contain only the information most salient to the classification decision. Such approaches provide intuitive feedback for a model's decisions, but are individual explanations isolated to each data point; gleaning general patterns from such *local explanations* is a cumbersome task. To capture a clearer picture of model behavior at a larger scale, we need a concept-driven approach to understanding complex tasks like genre classification.

As we discussed previously, deep neural networks often develop intermediate representations aligned with intuitive sub-tasks: object detectors in scene classifiers in vision [81], phoneme-like units in speech [20]. Extending this idea to music, instrument recognition can be viewed as an intermediate task that supports genre classification. We therefore hypothesize that genre classifiers develop internal representations that correspond to instrument concepts.

To this end, we train a genre classifier, and then further investigate the suitability of its intermediate representations for instrument classification. We expect that such a genre classifier would learn intermediate representations that have strong discriminatory ability on instruments. Further, we design an experiment to identify whether there is a *causal* relationship between such instrument concepts if found, and the downstream task of genre classification. We expect there to be such a relationship for specific pairs of strongly associated instruments and genres, *e.g.* the violin and classical music. Testing such a hypothesis also requires then, that we expose instrument concepts within a model in a manner amenable to some sort of *intervention*, and we describe our methodology for the same in further detail below.

Genre	Count	Instrument	Count
Blues	3,469	Clarinet	1,340
Classical music	4,126	Electric guitar	9,786
Country	4,317	Female singing	6,743
Disco	3,097	Flute	3,599
Hip hop music	5,952	Piano	8,436
Jazz	3,169	Saxophone	1,814
Heavy metal	3,591	Trumpet	2,740
Pop music	6,589	Violin, fiddle	23,175
Reggae	2,331		
Rock music	5,170		
Total	41,811	**Total**	57,633

Table 3.1: Class counts for genres in GTZANLike and instruments in MedleySolosDBLike.

3.2 Training and probing a genre classifier

3.2.1 Training procedure

Model architecture. We adopt the audio spectrogram transformer (AST) [40] architecture, which provides state-of-the-art results on audio classification. Further, we do not use audio pre-trained AST checkpoints as they are typically trained on AudioSet [38] and contain instruments among their label set. This explicit supervision impedes our study of *emergence* of instruments as intermediate concepts in genre classifiers. Instead, following the AST paper's original setup, we initialize with pre-trained DEiT [73] image weights and use AST's cut and bi-linear interpolate method to adapt positional embeddings. The model is only trained for genre classification.

Model training data. We adopt genre labels from GTZAN [75], a popular genre classification dataset with 10 diverse genre labels. However, the dataset features 100 examples per class that are insufficient to train the AST model. Thus, we create **GTZANLike** with the same 10 genre labels but with an order of magnitude higher number of audio samples from AudioSet [38]. We choose only those samples that are tagged with *exactly* one of the 10 genre labels. We obtain 41,811 audio samples and their distribution is given in Table 3.1.

Data pre-processing. We follow a similar pre-processing pipeline as AST [40]. We resample all audio to $22.5\,\mathrm{kHz}$ and compute spectrograms with n_{fft} set to 1024, a window length of $25\,\mathrm{ms}$, and a hop length of $10\,\mathrm{ms}$. These are then log Mel-scaled to 128 Mel bands. Most instances in the dataset are $10\,\mathrm{s}$ long, and we right pad instances with silence such that all computed spectrograms have a duration of 1024 frames, and normalize our data by subtracting μ (mean) and dividing by 2σ (standard deviation).

Transform	Range	p
Noise	scale: [0.01, 0.1]	0.50
Pitch mask	size: [2, 4]	0.25
Time mask	size: [4, 64]	0.25
Pitch shift	steps: [-0.5, 0.5]	0.50

Table 3.2: Data augmentation configuration. size/rate/steps are drawn from a uniform distribution, and then the augmentation is applied with probability p.

Training details. We adopt various data augmentation procedures indicated in Table 3.2. Training hyperparameters are tuned via a mix of random and grid search, and the configuration yielding the best validation macro average F1-score is adopted for all subsequent experiments. We train for 25 epochs using cross-entropy loss, with the AdamW [54] optimizer and a learning rate of 10^{-5}. We use an 80/10/10 split for our train/validation/test sets resulting in unique samples across splits.

3.2.2 Evaluating genre classification

Our best model achieves a macro average F1-score of 58.0% on the validation set, and 58.5% on the test set. From Fig. 3.1, we see that the model achieves strong performance on most genres, though some genres like *jazz* and *blues* exhibit relatively poor performance. Genres like *rock* and *metal* share core instrumentation (distorted guitars, drums, bass) and have overlapping timbral signatures. This makes them perceptually confusing even for humans, and the model reflects that ambiguity. Further, *pop* as a genre is dynamic and continually evolves, often absorbing stylistic features from whichever genres that dominate a given era.

Finally, classifying genres is inherently difficult owing to the fuzziness of genre boundaries as noted in [69] and misclassifications often occur in ways "similar to what a human would do," for example, classifying *disco* tracks as *pop*, or *rock* tracks as *metal* or *blues*. Nonetheless, we see sufficiently high accuracy over chance performance to probe this model for presence of instrument concepts.

3.2.3 Probing for instrument concepts

To investigate whether instrument information is encoded in intermediate layers of the AST, we train linear probes to classify between instruments at each layer of the network.

Probe training dataset. We adopt instrument labels from Medley-solos-DB [55], an instrument classification dataset with 8 labels. However, Medley-solos-DB only contains monophonic recordings resulting in a distribution mismatch between the instrument probe training data and genre classifier data that contains many polyphonic recordings. Thus, we follow a similar strategy as before and create **MedleySolosDBLike**, a new dataset with audio samples from AudioSet that are tagged with the 8 instrument categories defined in Medley-solos-DB. Each instance contains exactly one of the eight instrument la-

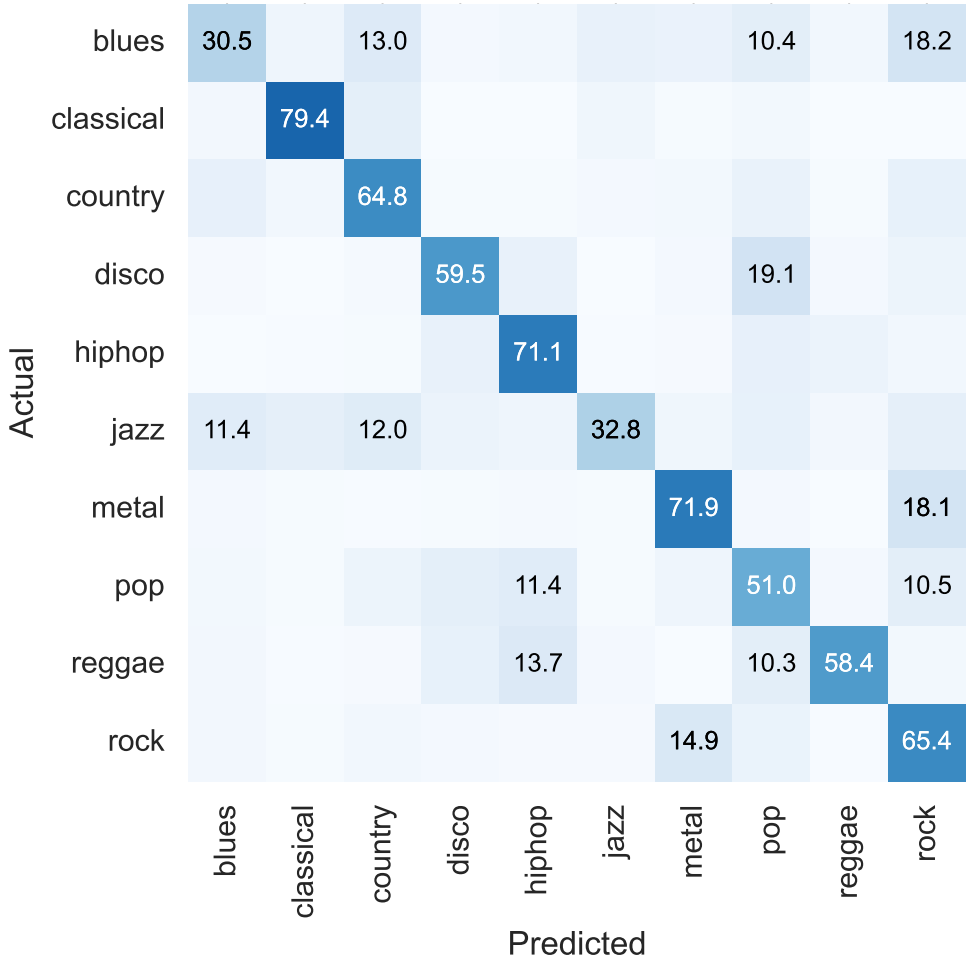


Figure 3.1: Normalized confusion matrix for our AST model on GTZANLike (test set). Model performance is above chance across all genres, though slightly weaker for *blues* and *jazz*. All misclassifications $> 10\%$ are highlighted for brevity and often occur between closely related genres such as *rock* and *metal*.

bels, and we ensure no overlap with audio samples used in GTZANLike to avoid data leakage. The dataset contains 57,633 valid samples with the distribution reported in Table 3.1. This enables systematic evaluation of whether instrument concepts emerge in genre classifiers trained without explicit instrument supervision, and whether such concepts can be isolated through linear probes.

Probe training setup. For an audio input converted to spectrogram x, our AST with L=12 layers computes the audio representation as

$$h(x) = h_{11}(h_{10}(\cdots(h_0(x))\cdots))\,. \tag{3.1}$$

We train layer-wise instrument probes using $h_{l_{[\mathrm{CLS}]}}$, the `[CLS]` token's representation at the output of each intermediate layer l.

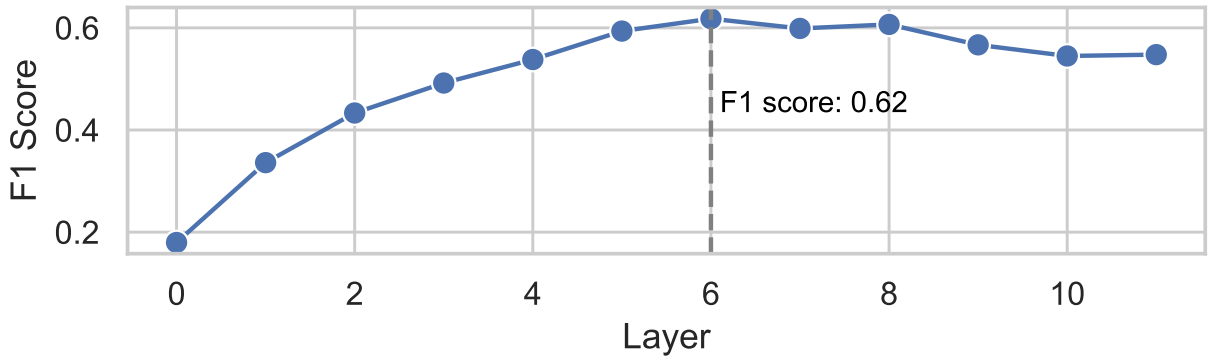


Figure 3.2: Macro average F1-score on MedleySolosDBLike (test set) for multi-class instrument probes. Probes are trained on intermediate layer representations with balanced sampling. Performance peaks at the middle layers.

3.2.4 Results and Discussion

We present the layer-wise linear probe Macro-average F1 scores for multi-class instrument classification in Fig. 3.2. The peak in the middle indicates that intermediate layers are indeed better at encoding instrument information. This suggests that while mid-level layers capture instrument features in a form suitable to recognize them, later layers transform them into higher-level abstractions that emphasize genre-relevant information without any instrument training.

3.3 Instruments as steering vectors

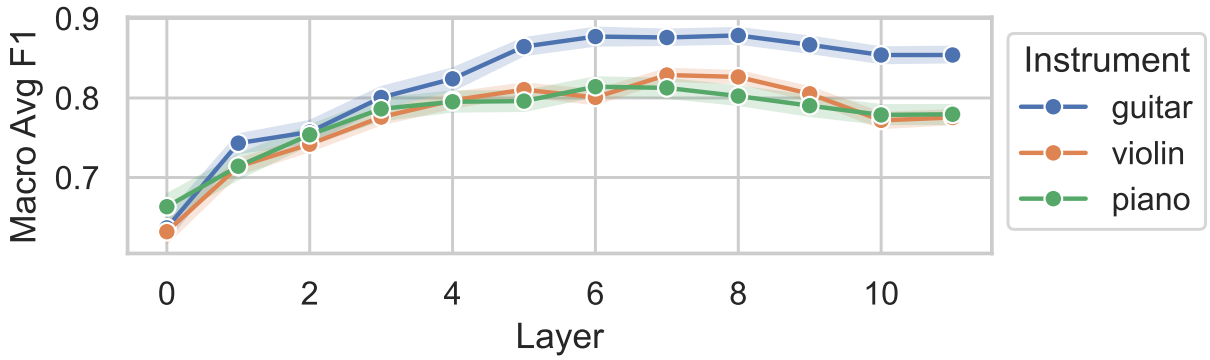


Figure 3.3: Macro average F1-score on MedleySolosDBLike (test set) for **binary** instrument probes. Probes are trained on intermediate layer representations with 100 random subsamples (250 positives, 250 negatives); error bands show 95% confidence intervals. Performance rises from early layers and peaks at intermediate layers. Results are shown for *guitar*, *violin*, and *piano*; similar results are observed for other instrument probes.

Are the emergent instrument concepts identified in intermediate layers causally influential for genre classification? From the genres learned by our model with a high F1-score, we specifically consider

those which have strong associations with certain instruments. *Rock* and *metal* are both strongly associated with the *electric guitar*, and *classical music* with the sound of the *violin*. Similarly, the sound of *saxophone* is a mainstay in *jazz music*.

We hypothesize that these instruments strongly influence their corresponding genres, *i.e.* *guitar* has an influence on *rock* and *metal*. We also note that some instruments, are associated with multiple genres. For example, *piano*, which includes electric keyboard is commonly found in *classical*, *pop*, *jazz* and *rock*. As a result, we hypothesize that *piano* would not have a strong influence on *classical*, *jazz*, *rock* or *metal*.

To test our hypotheses, we design an experiment to steer model (genre) representations along instrument vectors to see if predictions move toward the corresponding genres.

3.3.1 Experimental setup

Instrument vectors.

For a given layer, we train several **binary** linear probes (using $h_{l_{[\text{CLS}]}}$) with varying random seeds and extract their weights. These weights correspond to the vector normal to the hyperplane separating the instrument's positive and negative samples, and thus encode the *direction* of the instrument in the representation space. We denote the mean vector across seeds as v_l, the instrument vector at layer l. Using multiple seeds (100 in our case) helps identify statistically robust directions for instrument vectors and also deal with the class imbalance in MedleySolosDBLike. Note that all vectors within an instrument class are well aligned with an average pairwise cosine similarity of > 0.9 in all cases. Fig. 3.3 shows similar trends for binary classification as the multi-class instrument probes (Fig. 3.2), with peak performance at the middle layers.

ions.

Intervention experiments

We perturb $h_{l_{[\text{CLS}]}}$, the genre representation at the `[CLS]` token, as

$$\hat{h}_{l_{[\text{CLS}]}} = h_{l_{[\text{CLS}]}} + \lambda \cdot \frac{v}{\|v\|} \cdot \|h_{l_{[\text{CLS}]}}\|\,, \tag{3.2}$$

where $l \in \{0, \ldots, L-1\}$ and $\lambda \in [-1, 1]$ is a scalar that decreases or increases the contribution of the instrument representation.

Similar to [33], we intervene on **all layers** of the model and consider two intervention strategies based on the choice of the instrument vector v used to perturb model representations:

1. **one-to-one** uses the instrument vector v_l computed from the **same** layer to affect the model representation at each layer l.
2. **one-to-all** uses an instrument vector v_k from a **fixed** layer k to affect the model representation at each layer l.

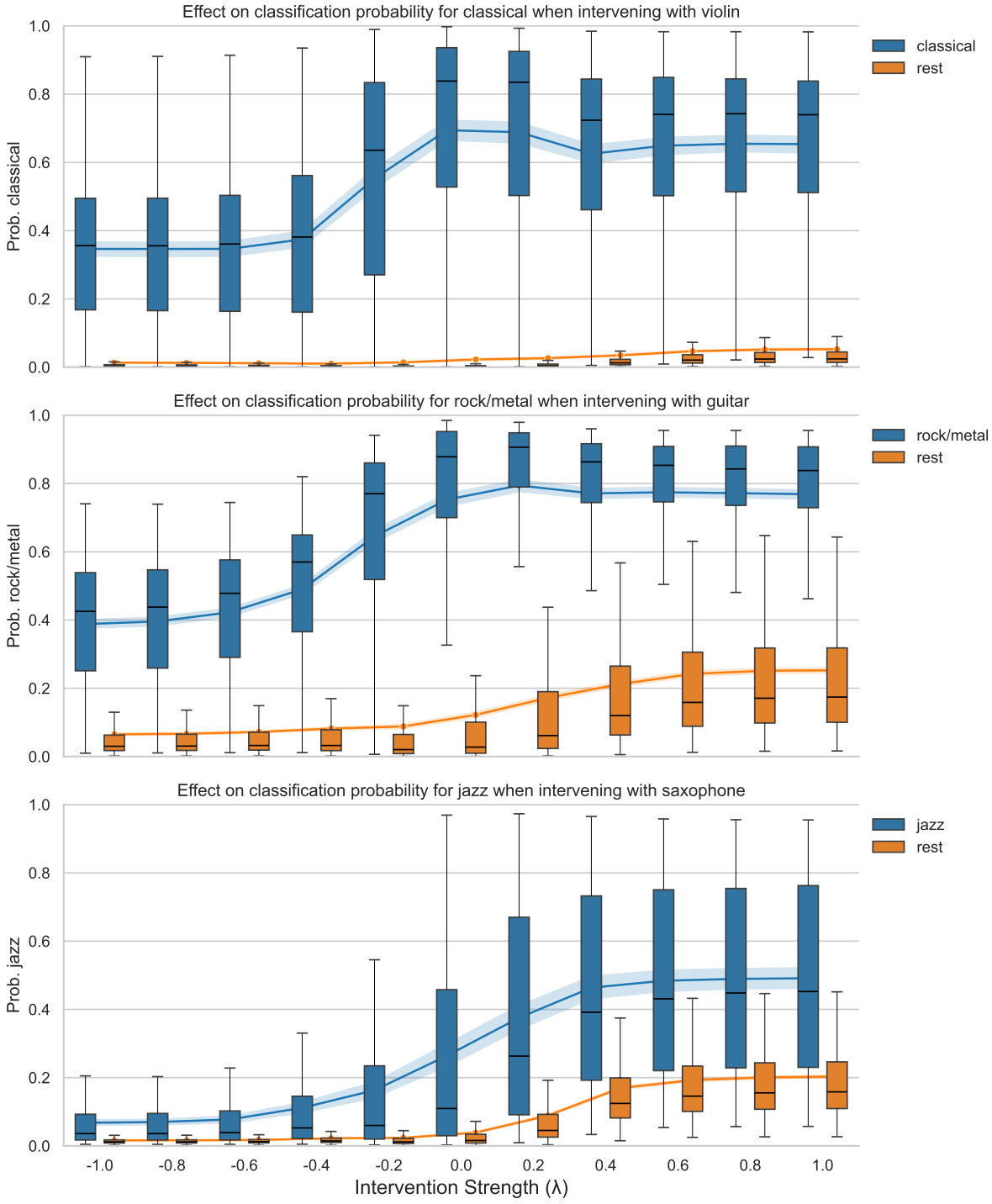


Figure 3.4: Effect on the probability of being classified as a particular genre when intervened on with the corresponding instrument. We intervene on instances from GTZANLike (test) labeled with the genre (blue) as well as instances labeled with any of the remaining genres (orange). In all cases, we see that adding the instrument vector has stronger effects on instances which are **not** from the corresponding genre, and subtracting the instrument vector has stronger effects on instances which are from the corresponding genre. We use the one-to-one steering strategy for these instruments.

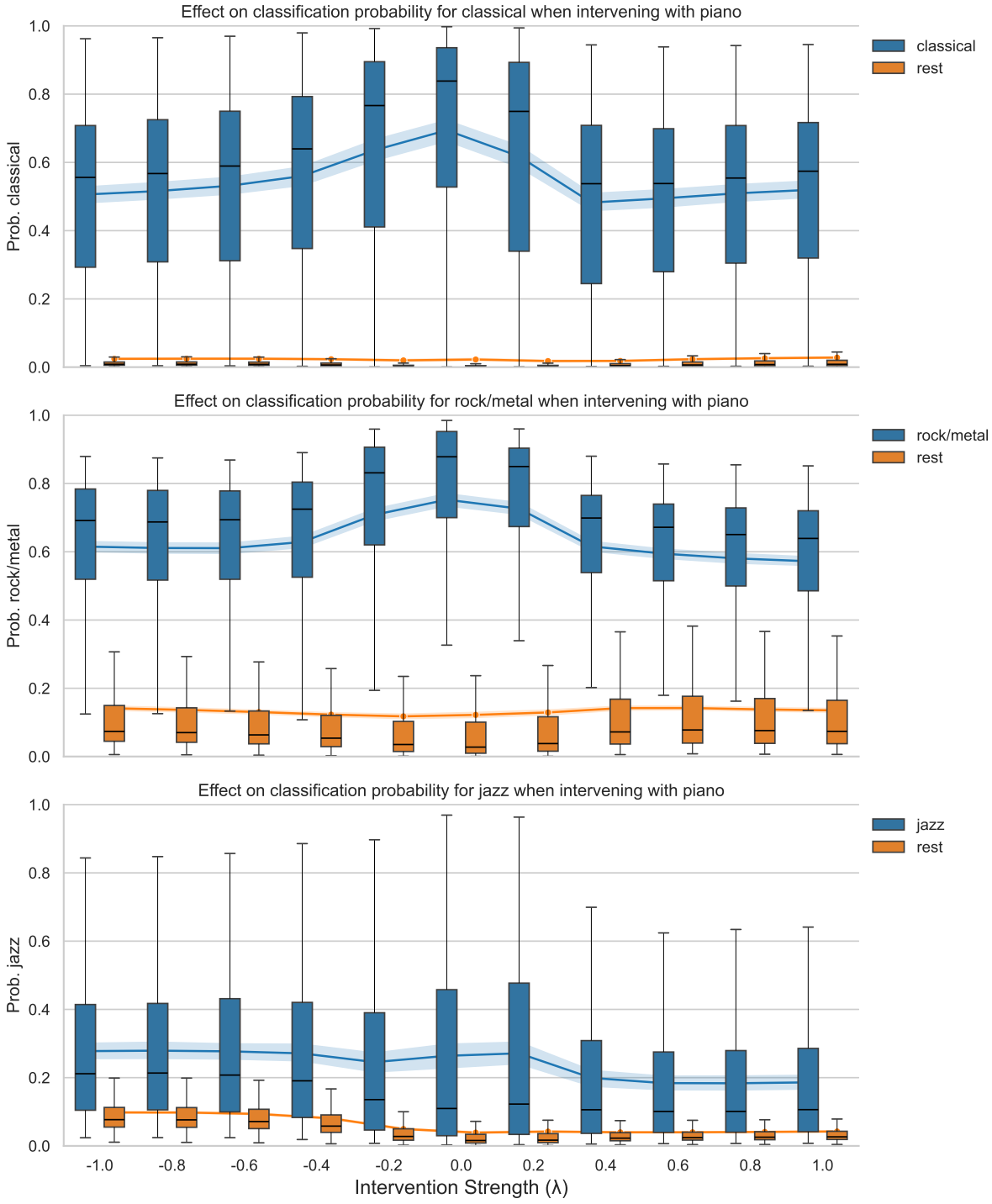


Figure 3.5: Effect on the probability of being classified as a particular genre when intervened on with piano, an instrument not strongly associated with any one genre. We see classification probabilities for classical (top), rock-metal (middle), and jazz (bottom). We intervene on instances from GTZANLike (test) labeled with the genre (blue) as well as instances labeled with any of the remaining genres (orange). In all cases, no strong trend is observed in terms of the changes in the probability on intervening. We use the one-to-one steering strategy for these instruments.

We demonstrate the effects of such interventions further with the classification task where we take instances from each genre, focusing on the associated genre as one class, and all of the other genres as the other class. For *e.g.*, we take instances from *classical* and instances from the *rest* of the genres, and intervene with the *violin* concept vector for both of them. We also show how *piano*, not associated with any one genre exclusively, does not have strong effects when intervening on instances of various genres.

3.3.2 Results and discussion

We observe that both intervention strategies are effective at steering genre predict

One-to-one steering results are presented in Fig. 3.4. We observe that adding the *guitar* vector *increases* the probability of an instance being classified as *rock-metal*, while subtracting the *guitar* vector *reduces* the probability of an instance being classified as *rock-metal*.

When we use instances that already belong to the *rock-metal* genre, there is little effect of adding the vector, and a stronger effect of subtracting it. We note that when using instances that do not belong to the *rock-metal*, it is adding the vector that has a stronger effect. This makes sense; if the instance is already being classified as *rock-metal* with high probability, adding the *guitar* concept cannot increase the probability by much before saturating. We also note that intervening on instances from other genres by adding the *guitar* vector has a relatively weaker effect. We see similar trends in the effects for *violin* vectors on *classical*, and *saxophone* vectors on *jazz* as well.

For instruments that are not strongly associated with a single genre, *e.g.* the *piano*, we also see that the interventions in either direction (add/subtract) have limited effect on genre classification, as seen from Fig. 3.5. We largely observe no directional effect, and only a small decrease in some cases on intervening, which we attribute to a loss in information fidelity from the interventions. There is a small negative effect of *piano* on *jazz*, leading to a slight increase in the probability of being classified as *jazz* if the *piano* vector is subtracted. This however, is negligible compared to the effects we see for instruments and genres which we hypothesized to be correlated, such as *violin* and its effect on *classical*.

We see similar results when we extend the same experiments to the GTZAN dataset, which shares the set of genre labels used in our pre-training dataset (by construction), but is completely unseen to the model during the training phase. GTZAN has 30 second clips, from which we select the middle 10 seconds to maintain compatibility with our model's pre-training on 10 second clips from GTZANLike.

The same trends are found, even slightly more pronounced for GTZAN. The probability of being classified as a certain genre showing an effect corresponding to the instrument vector with which we intervene on the model (Fig. 3.6). Here too, we see that *piano* does not have a strong directional effect (Fig. 3.5). These results show our findings are robust, preserving trends even when applied to another dataset.

One-to-all steering allows us to study the effectiveness of intervening with instrument vectors computed from different layers. Fig. 3.8 shows that instrument vectors extracted from later layers are more

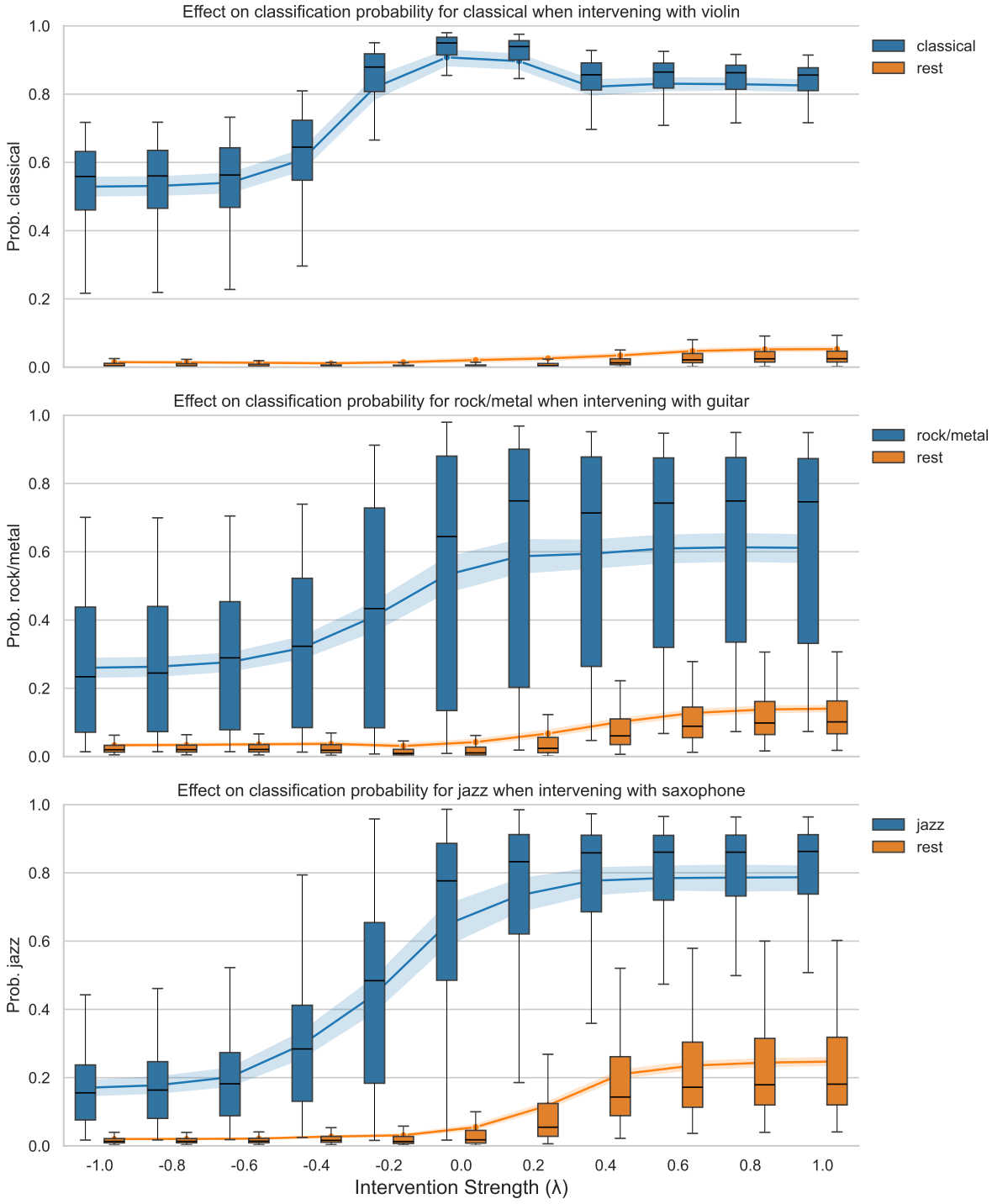


Figure 3.6: Effect on the probability of being classified as a particular genre when intervened on with the corresponding instrument. We intervene on instances from GTZAN labeled with the genre (blue) as well as instances labeled with any of the remaining genres (orange). In all cases, we see that adding the instrument vector has stronger effects on instances which are **not** from the corresponding genre, and subtracting the instrument vector has stronger effects on instances which are from the corresponding genre. We use the one-to-one steering strategy for these instruments.

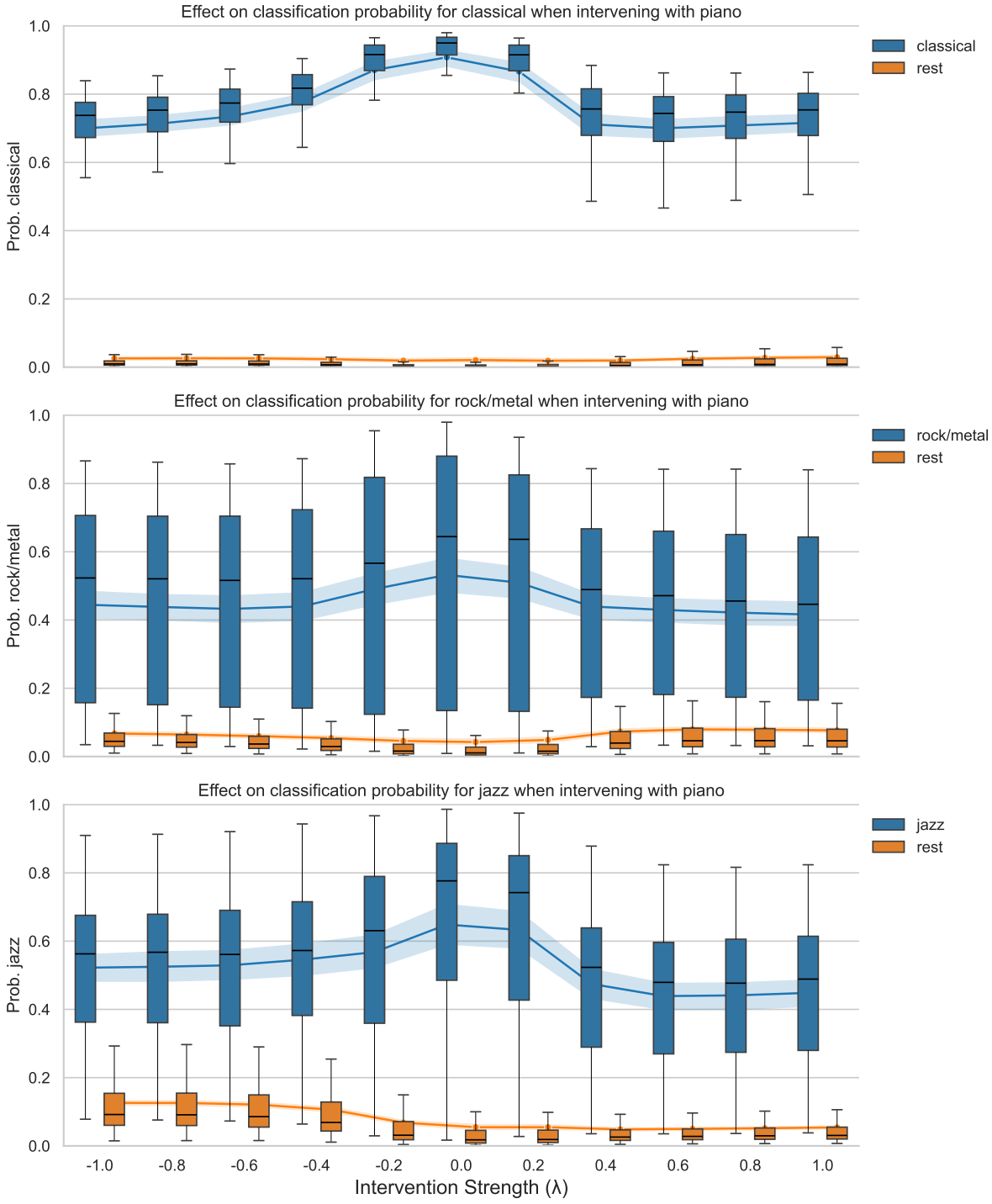


Figure 3.7: Effect on the probability of being classified as a particular genre when intervened on with piano, an instrument not strongly associated with any one genre. We see classification probabilities for classical (top), rock-metal (middle), and jazz (bottom). We intervene on instances from GTZAN labeled with the genre (blue) as well as instances labeled with any of the remaining genres (orange). In all cases, no strong trend is observed in terms of the changes in the probability on intervening. We use the one-to-one steering strategy for these instruments.

effective at steering the corresponding genre's classification probability. Specifically, adding the *guitar* vector increases probability of *rock-metal* for *classical* instances. Similar results are seen for the other instrument-genre pairs as well. Thus, we show that vectors learned from emergent instrument concepts can indeed be used to steer genre classification.

Are instrument vectors same as genre vectors? Lastly, a simple explanation for the observed steering effects is that instrument vectors are aligned with genre vectors (*e.g.* *guitar* with *rock-metal*.) We refute this by first computing *genre probe vectors* from GTZANLike (val set), and comparing their similarity to instrument vectors. From Fig. 3.9, we note that the instrument and genre vectors are somewhat similar at early-middle layers. However, intervening with instrument vectors computed from these layers has limited effect on genre classification as discussed earlier (Fig. 3.8). The strongest steering effects are observed at later layers, where the instrument vectors and genre vectors are **not** similar. This suggests that, even though instrument vectors are dissimilar to genre vectors, they provide the model with genre-relevant information that allows its predictions to be steered.

3.3.3 Limitations

Our analysis was limited to one architecture (AST) and a restricted set of instruments and genres; further work is needed to establish how general these findings are across models. Also, we focused on genres strongly associated with specific instruments, but other genres may be better characterized by some other sonic properties such as rhythmic motifs or harmonic structure. Future research could broaden the scope of probing and steering to capture these additional factors and to develop controllable models that make such internal structure usable.

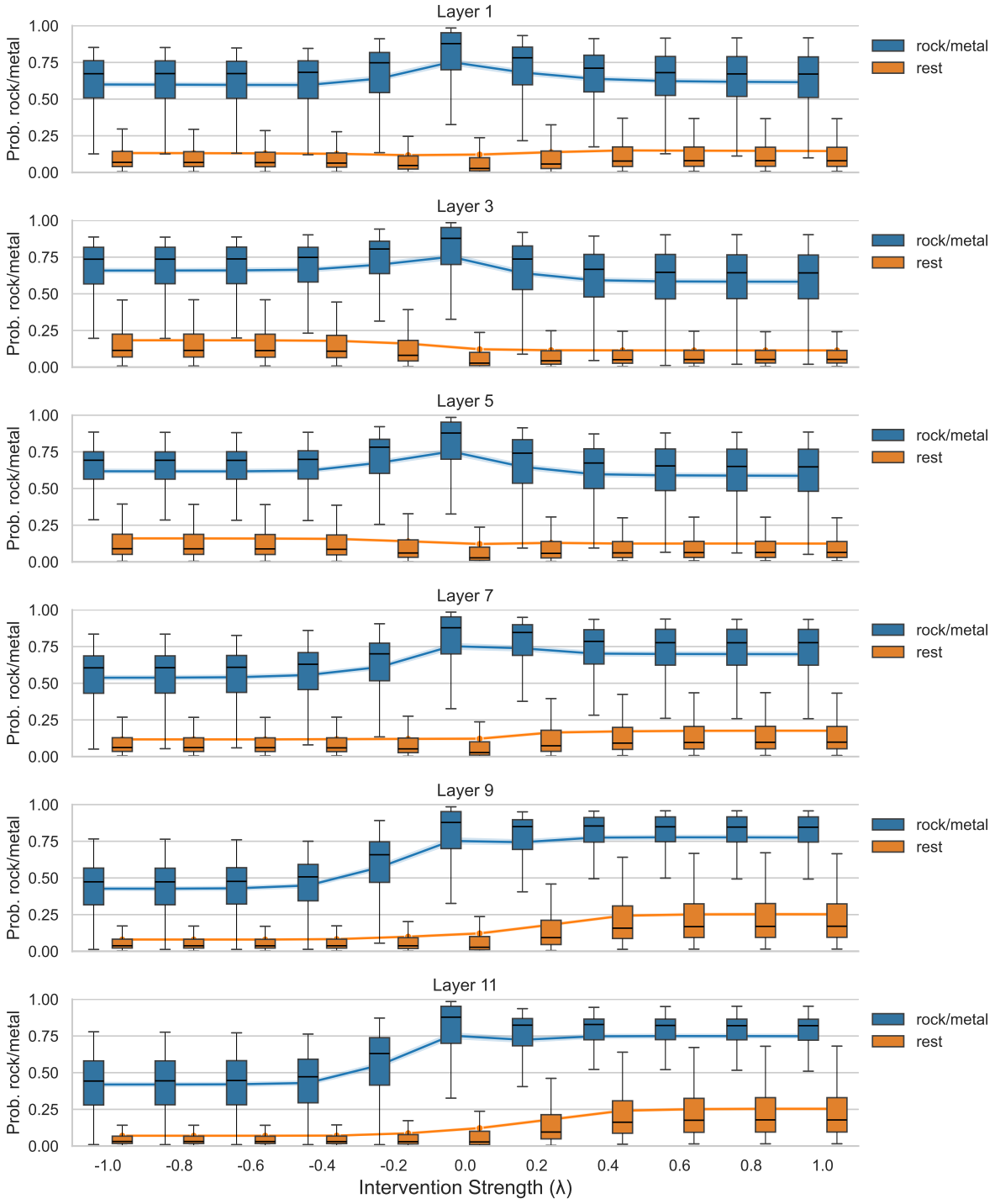


Figure 3.8: The effect of intervening with the *guitar* vector on *rock-metal* using the **one-to-all** strategy. We show results for *guitar* vectors extracted from different layers. Interventions using instrument vectors from early layers have a negligible effect. In contrast, adding the *guitar* vector from later layers increases the probability of classification as *rock-metal*, while subtracting it decreases this probability. This behavior mirrors the results observed with the **one-to-one** strategy. Similar trends are observed for other instrument–genre pairs.

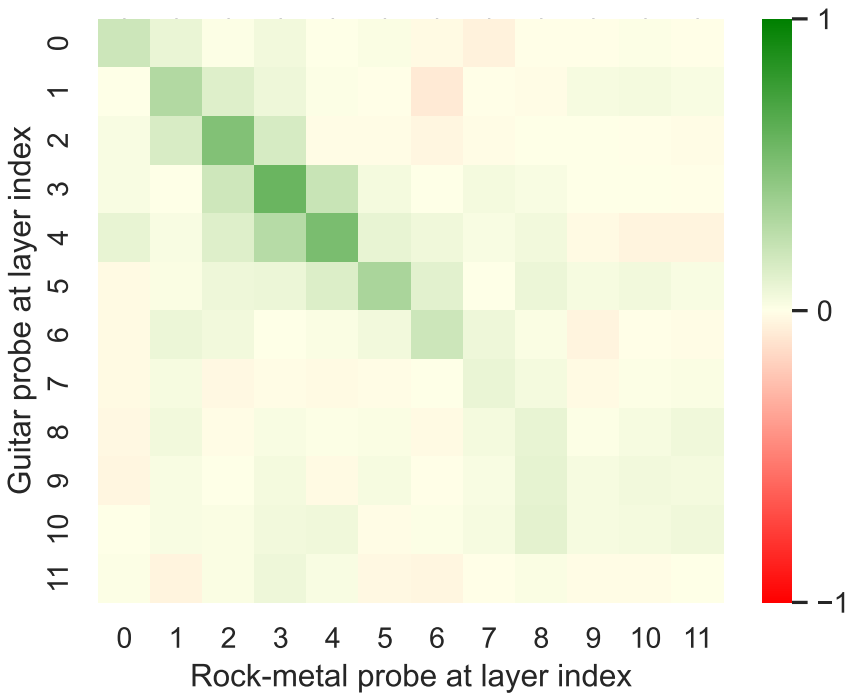


Figure 3.9: Cosine similarities between the vectors for *guitar* and *rock-metal* at different layers. Note how the similarity is greatest at the earlier layers, but we see the effect of the intervention is strongest at later layers (Fig. 3.8).

Chapter 4

Conclusion

In this thesis, we posed the question: do DNNs trained on audio tasks develop a compositional hierarchy analogous to the one observed in human auditory perception, progressing from low-level acoustic primitives to high-level semantic representations? Drawing on prior work in vision, where convolutional neural networks are known to develop structured representations mirroring the progression from edges to objects, and on auditory scene analysis, which describes how listeners organize sound into coherent streams and objects, we hypothesized that audio DNNs would exhibit a similar layered organization. The results presented in this thesis provide strong evidence that they do.

Across multiple convolutional architectures, including VGGish, CLAP, and MobileNetV3, we observed a consistent layer-wise pattern. Tasks grounded in basic acoustic properties, such as *note name classification*, achieved peak performance in early layers, whereas higher-level tasks, including *genre classification* and *speaker count estimation*, depended on the integrative capacity of deeper layers. This mirrors the compositional logic discussed in the introduction: just as human listeners are understood to build up from spectral primitives to auditory streams and ultimately to scene-level understanding, these networks resolve low-level features before assembling them into the abstract categories required by their training objectives.

We extended this analysis to the Audio Spectrogram Transformer (AST), comparing the results against those seen for CNN architectures, and discovered that while architectural choices shape how the hierarchy is expressed, the transition from acoustic primitives to semantic representations is a robust property of effective audio models, not an artifact of any single architecture.

We also investigated the emergent relationship between musical instruments and genres. In the introduction, we highlighted how vision models trained solely for scene classification spontaneously develop neurons that act as object detectors, without explicit supervision for those categories. We observed a notably similar phenomenon in the auditory domain. By training an AST model exclusively for genre classification, we showed that representations corresponding to musical instruments emerge naturally in intermediate layers, despite the absence of instrument-level supervision. Linear probing analyses indicated that these layers are well suited for encoding auditory objects such as instruments, which in turn support high-level genre decisions.

Importantly, we demonstrated through causal intervention experiments that these emergent representations are not merely correlational. Injecting instrument-specific steering vectors into the model's internal representations reliably shifted its predictions toward associated genres, for example steering a sample toward jazz using a saxophone vector. These effects were strongest in later layers, where instrument and genre representations are most clearly differentiated.

A compositional, hierarchical view of auditory processing provides a principled basis for understanding how complex sound is organized. Within this framework, it is worth noting that human auditory perception is not purely bottom-up: top-down influences, including prior knowledge, attentional modulation, and schema-driven expectations, play a role in how we process sound. Crucially, however, these top-down effects operate over, and depend upon, an underlying hierarchy of representations, rather than replacing it; the hierarchy is thus not a rigid pipeline but a dynamically modulated structure. In this light, the feedforward models we study, despite lacking explicit top-down or recurrent connections, nonetheless learn a clear compositional hierarchy. This supports the view that bottom-up structure is a foundational organizational principle, and that a top-down analytical perspective remains both appropriate and informative.

In summary, this thesis demonstrates both the emergence of meaningful concepts and their causal role in the intermediate representations of audio deep neural networks. We asked whether these models organize sound in a way that parallels human auditory perception, and the results across multiple architectures and experiments support this view. Together, the findings suggest that audio DNNs organize sound into structured and hierarchical representations of the acoustic world.

These insights have implications across audio applications. In music, the finding that genre judgments build on instrument-level representations can inform the design of more interpretable systems for music analysis and generation. More broadly, the framework developed here provides a way to study whether similar hierarchies appear in other domains such as speech and language processing. While this work focuses on specific architectures and musical concepts, it establishes a framework for studying interpretability in auditory models. Future work could compare these representational hierarchies with those observed in the human brain, extend the methods to broader audio domains, and explore practical applications such as controllable audio generation through manipulation of internal representations.

Related Publications

[P1] Pratyaksh Gautam, Makarand Tapaswi, and Vinoo Alluri, **"Localizing Auditory Concepts in CNNs"**, accepted at the *Mechanistic Interpretability Workshop, 41st International Conference on Machine Learning (ICML)*, 2024.

[P2] Pratyaksh Gautam, Makarand Tapaswi, and Vinoo Alluri, **"Auditory CNN Analysis: What Do Layers Encode?"**, accepted at the *18th International Conference on Music Perception and Cognition (ICMPC)*, 2025.

[P3] Pratyaksh Gautam, Makarand Tapaswi, and Vinoo Alluri, **"Instrument Detectors Emerge in Genre Classifiers - and They Can Steer Genre Classification"**, *to be submitted at* the *The 27th International Society for Music Information Retrieval Conference*, 2026.

Bibliography

[1] Charu C Aggarwal, Alexander Hinneburg, and Daniel A Keim. On the surprising behavior of distance metrics in high dimensional space. In *International Conference on Database Theory*, pages 420–434. Springer, 2001.

[2] Andrea Agostinelli, Timo I Denk, Zalán Borsos, Jesse Engel, Mauro Verzetti, Antoine Caillon, Qingqing Huang, Aren Jansen, Adam Roberts, Marco Tagliasacchi, et al. MusicLM: Generating music from text. *arXiv preprint arXiv:2301.11325*, 2023.

[3] Alican Akman, Qiyang Sun, and Björn W Schuller. Audio explanation synthesis with generative foundation models. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2025.

[4] Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. In *International Conference on Learning Representations (ICLR), Workshop Track Proceedings*, 2017.

[5] Sercan Arik, Jitong Chen, Kainan Peng, Wei Ping, and Yanqi Zhou. Neural voice cloning with a few samples. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, 2018.

[6] Jean-Julien Aucouturier. Sounds like teen spirit: Computational insights into the grounding of everyday musical terms. *Language, evolution and the brain*, pages 35–64, 2009.

[7] Jean-Julien Aucouturier and Francois Pachet. Representing musical genre: A state of the art. *Journal of new music research*, 32(1):83–93, 2003.

[8] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[9] Max Bain, Jaesung Huh, Tengda Han, and Andrew Zisserman. WhisperX: Time-Accurate Speech Transcription of Long-Form Audio. In *Interspeech*, 2023.

[10] Anna Bavaresco, Nhut Truong, and Uri Hasson. Modeling human concepts with subspaces in deep vision models. *ACM Transactions on Interactive Intelligent Systems*, 15(4):1–25, 2025.

[11] Yonatan Belinkov. Probing Classifiers: Promises, Shortcomings, and Advances. *Association of Computational Linguistics (ACL)*, 2022.

[12] Irving Biederman. Recognition-by-components: a theory of human image understanding. *Psychological review*, 94(2):115, 1987.

[13] Jennifer K Bizley and Yale E Cohen. The what, where and how of auditory-object perception. *Nature Reviews Neuroscience*, 14(10):693–707, 2013.

[14] Constant Bonard. *What is a Musical Genre and What is its Use?* PhD thesis, University of Geneva, 2014.

[15] Albert S Bregman. *Auditory scene analysis: The perceptual organization of sound*. 1990.

[16] Sofia Brené and Carl Thome. How musical instrumentation affects perceptual identification of musical genres. *Bachelor's thesis at KTH Royal Institute of Technology*, 2014.

[17] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[18] Michael A Casey. Music of the 7ts: Predicting and decoding multivoxel fmri responses with acoustic, schematic, and categorical music features. *Frontiers in psychology*, 8:1179, 2017.

[19] Bo Chen, Chenpeng Du, and Kai Yu. Neural fusion for voice cloning. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:1993–2001, 2022.

[20] Cheol Jun Cho, Abdelrahman Mohamed, Shang-Wen Li, Alan W Black, and Gopala K. Anumanchipalli. SD-HuBERT: Sentence-Level Self-Distillation Induces Syllabic Organization in Hubert. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024.

[21] Keunwoo Choi, George Fazekas, and Mark Sandler. Explaining Deep Convolutional Neural Networks on Music Classification. *arXiv preprint arXiv:1607.02444*, 2016.

[22] Keunwoo Choi, György Fazekas, Mark Sandler, and Kyunghyun Cho. Convolutional recurrent neural networks for music classification. In *2017 IEEE International conference on acoustics, speech and signal processing (ICASSP)*, pages 2392–2396. IEEE, 2017.

[23] Kwanghee Choi, Ankita Pasad, Tomohiko Nakamura, Satoru Fukayama, Karen Livescu, and Shinji Watanabe. Self-supervised speech representations are more phonetic than semantic. In *Interspeech*, 2024.

[24] Yunfei Chu, Jin Xu, Xiaohuan Zhou, Qian Yang, Shiliang Zhang, Zhijie Yan, Chang Zhou, and Jingren Zhou. Qwen-audio: Advancing universal audio understanding via unified large-scale audio-language models. *arXiv preprint arXiv:2311.07919*, 2023.

[25] Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Défossez. Simple and controllable music generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[26] Alastair JD Craft, Geraint A Wiggins, and Tim Crawford. How many beans make five? the consensus problem in music-genre classification and a new evaluation method for single-genre categorisation systems. In *International Society of Music Information Retreival (ISMIR)*, 2007.

[27] Alex J DeGrave, Joseph D Janizek, and Su-In Lee. AI for radiographic COVID-19 detection selects shortcuts over signal. *Nature Machine Intelligence*, 3(7):610–619, 2021.

[28] Soham Deshmukh, Benjamin Elizalde, Rita Singh, and Huaming Wang. Pengi: An audio language model for audio tasks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[29] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *North American Chapter of Association of Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 2019.

[30] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, G Heigold, S Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*, 2021.

[31] Benjamin Elizalde, Soham Deshmukh, Mahmoud Al Ismail, and Huaming Wang. CLAP: Learning Audio Concepts from Natural Language Supervision. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023.

[32] Jesse Engel, Cinjon Resnick, Adam Roberts, Sander Dieleman, Mohammad Norouzi, Douglas Eck, and Karen Simonyan. Neural audio synthesis of musical notes with wavenet autoencoders. In *International Conference on Machine Learning (ICML)*, 2017.

[33] Simone Facchiano, Giorgio Strano, Donato Crisostomi, Irene Tallini, Tommaso Mencattini, Fabio Galasso, and Emanuele Rodolà. Activation Patching for Interpretable Steering in Music Generation. *arXiv preprint arXiv:2504.04479*, 2025.

[34] Daniel J Felleman and David C Van Essen. Distributed hierarchical processing in the primate cerebral cortex. *Cerebral cortex (New York, NY: 1991)*, 1(1):1–47, 1991.

[35] Rafael Ferrer. *The socially distributed cognition of musical timbre: a convergence of semantic, perceptual, and acoustic aspects*. University of Jyväskylä, 2012.

[36] Seth* Forsgren and Hayk* Martiros. Riffusion - Stable diffusion for real-time music generation, 2022. URL `https://riffusion.com/about`.

[37] Josh Gardner, Simon Durand, Daniel Stoller, and Rachel Bittner. Llark: a multimodal instruction-following language model for music. In *International Conference on Machine Learning (ICML)*, 2024.

[38] Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter. Audio Set: An ontology and human-labeled dataset for audio events. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017.

[39] Robert O Gjerdingen and David Perrott. Scanning the dial: The rapid recognition of music genres. *Journal of new music research*, 37(2):93–100, 2008.

[40] Yuan Gong, Yu-An Chung, and James Glass. AST: Audio Spectrogram Transformer. In *Interspeech*, 2021.

[41] Timothy D Griffiths and Jason D Warren. What is an auditory object? *Nature Reviews Neuroscience*, 5(11):887–892, 2004.

[42] Shawn Hershey, Sourish Chaudhuri, Daniel P. W. Ellis, Jort F. Gemmeke, Aren Jansen, R. Channing Moore, Manoj Plakal, Devin Platt, Rif A. Saurous, Bryan Seybold, Malcolm Slaney, Ron J. Weiss, and Kevin Wilson. CNN Architectures for Large-Scale Audio Classification. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017.

[43] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, Quoc V. Le, and Hartwig Adam. Searching for MobileNetV3. In *International Conference on Computer Vision (ICCV)*, 2019.

[44] Po-Yao Huang, Hu Xu, Juncheng Li, Alexei Baevski, Michael Auli, Wojciech Galuba, Florian Metze, and Christoph Feichtenhofer. Masked autoencoders that listen. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[45] Fatima T Husain, M-A Tagamets, Stephen J Fromm, Allen R Braun, and Barry Horwitz. Relating neuronal dynamics for auditory object processing to neuroimaging activity: a computational modeling and an fMRI study. *Neuroimage*, 21(4):1701–1720, 2004.

[46] Qiuqiang Kong, Yin Cao, Turab Iqbal, Yuxuan Wang, Wenwu Wang, and Mark D. Plumbley. PANNs: Large-Scale Pretrained Audio Neural Networks for Audio Pattern Recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:2880–2894, 2020.

[47] Khaled Koutini, Jan Schlüter, Hamid Eghbal-zadeh, and Gerhard Widmer. Efficient Training of Audio Transformers with Patchout. In *Interspeech*, 2022.

[48] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2012.

[49] Michael Kubovy and David Van Valkenburg. Auditory and visual objects. *Cognition*, 80(1-2): 97–126, 2001.

[50] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[51] Hao Li, Luke Helpard, Jonas Ekeroot, Seyed Alireza Rohani, Ning Zhu, Helge Rask-Andersen, Hanif M Ladak, and Sumit Agrawal. Three-dimensional tonotopic mapping of the human cochlea based on synchrotron radiation phase-contrast imaging. *Scientific reports*, 11(1):4437, 2021.

[52] Stefaan Lippens, Jean-Pierre Martens, and Tom De Mulder. A comparison of human and automatic musical genre classification. In *2004 IEEE international conference on acoustics, speech, and signal processing*, volume 4, pages iv–iv. IEEE, 2004.

[53] Haohe Liu, Zehua Chen, Yi Yuan, Xinhao Mei, Xubo Liu, Danilo Mandic, Wenwu Wang, and Mark D Plumbley. AudioLDM: Text-to-audio generation with latent diffusion models. In *International Conference on Machine Learning (ICML)*, 2023.

[54] Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations (ICLR)*, 2019.

[55] Vincent Lostanlen, Carmine-Emanuele Cella, Rachel Bittner, and Slim Essid. Medley-solos-DB: a cross-collection dataset for musical instrument recognition, 2019.

[56] Saber Malekzadeh, Mohammad Hossein Gholizadeh, Seyed Naser Razavi, and Hossein Ghayoumi Zadeh. The recognition of Persian phonemes using PPNet. *Journal of Medical Signals & Sensors*, 10(2):86–93, 2020.

[57] Mortimer Mishkin, Leslie G Ungerleider, and Kathleen A Macko. Object vision and spatial vision: two cortical pathways. *Trends in neurosciences*, 6:414–417, 1983.

[58] Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. An Overview of Early Vision in InceptionV1. *Distill*, 2020. doi: 10.23915/distill.00024.002. https://distill.pub/2020/circuits/early-vision.

[59] Francesco Paissan, Mirco Ravanelli, and Cem Subakan. Listenable Maps for Audio Classifiers. In *International Conference on Machine Learning (ICML)*, 2024.

[60] Jayneel Parekh, Sanjeel Parekh, Pavlo Mozharovskyi, Florence d'Alché-Buc, and Gaël Richard. Listen to Interpret: Post-hoc Interpretability for Audio Networks with NMF. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[61] Ankita Pasad, Ju-Chieh Chou, and Karen Livescu. Layer-wise analysis of a self-supervised speech representation model. In *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2021.

[62] Alessandro Raganato and Jörg Tiedemann. An analysis of encoder representations in transformer-based machine translation. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, 2018.

[63] Josef P Rauschecker. An expanded role for the dorsal auditory pathway in sensorimotor control and integration. *Hearing research*, 271(1-2):16–25, 2011.

[64] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016.

[65] Florian Schmid, Khaled Koutini, and Gerhard Widmer. Efficient Large-Scale Audio Tagging Via Transformer-to-CNN Knowledge Distillation. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023.

[66] Shihab Shamma. On the role of space and time in auditory processing. *Trends in cognitive sciences*, 5(8):340–348, 2001.

[67] Elana Simon and James Zou. Interplm: discovering interpretable features in protein language models via sparse autoencoders. *Nature methods*, 22(10):2107–2117, 2025.

[68] K Simonyan and A Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *International Conference on Learning Representations (ICLR)*, 2015.

[69] Bob L Sturm. Classification accuracy is not enough: On the evaluation of music genre recognition systems. *Journal of Intelligent Information Systems*, 41:371–406, 2013.

[70] Fabian-Robert Stöter, Soumitro Chakrabarty, Emanuël Habets, and Bernd Edler. Libricount, a dataset for speaker count estimation, 2018.

[71] Harri Taylor. torchvggish: Pytorch port of Google Research's VGGish model used for extracting audio features. — github.com. `https://github.com/harritaylor/torchvggish`. [Accessed 08-09-2024].

[72] Ian Tenney, Dipanjan Das, and Ellie Pavlick. BERT Rediscovers the Classical NLP Pipeline. In *Association of Computational Linguistics (ACL)*, 2019.

[73] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Herve Jegou. Training Data-efficient Image Transformers & Distillation Through Attention. In *International Conference on Machine Learning (ICML)*, 2021.

[74] Joseph Turian, Jordie Shier, Humair Raj Khan, Bhiksha Raj, Björn W Schuller, Christian J Steinmetz, Colin Malloy, George Tzanetakis, Gissel Velarde, Kirk McNally, et al. HEAR: Holistic Evaluation of Audio Representations. In *NeurIPS 2021 Competitions and Demonstrations Track*, pages 125–145. PMLR, 2022.

[75] George Tzanetakis, Georg Essl, and Perry Cook. Automatic Musical Genre Classification of Audio Signals. In *International Society of Music Information Retreival (ISMIR)*, 2001.

[76] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[77] Pete Warden. Speech commands: A dataset for limited-vocabulary speech recognition. *arXiv preprint arXiv:1804.03209*, 2018.

[78] Masoumeh Zareh, Elaheh Toulabinejad, Mohammad Hossein Manshaei, and Sayed Jalal Zahabi. A deep learning model of dorsal and ventral visual streams for dvsd. *Scientific Reports*, 14(1): 27464, 2024.

[79] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision (ECCV)*, 2014.

[80] Alice Zhang, Edison Thomaz, and Lie Lu. Transformation of audio embeddings into interpretable, concept-based representations. *International Joint Conference on Neural Networks (IJCNN)*, 2025.

[81] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Object Detectors Emerge in Deep Scene CNNs. In *International Conference on Learning Representations (ICLR)*, 2015.