Disentangling Polysemantic Channels in Convolutional Neural Networks

Robin Hesse[1] Jonas Fischer[2] Simone Schaub-Meyer[1,3] Stefan Roth[1,3]

1 Technische Universität Darmstadt
2 Max Planck Institute for Informatics
3 hessian.AI

Paper & Code

Motivation

Channels in convolutional neural networks can be polysemantic, *i.e.*, they encode multiple concepts [1].

Channel #1660 in the last layer of a ResNet-50 trained on ImageNet

MACO [2] feature visualization + most activating patches

Since polysemanticity impairs interpretability, our goal is to **disentangle polysemantic channels.**

High-level Idea

Different channels in $L-1$ are important for different concepts encoded in the same polysemantic channel in L.
→ The first channel in $L-1$ is important to detect bird tails while the second channel is important to detect car wheels.

After **identifying** a polysemantic channel in L, we can **disentangle** it by restructuring the weights from $L-1$ based on the respective concept.

Identifying Polysemantic Channels

Find out if a channel in L is relevant for class t by computing the attribution of the respective output logit w.r.t. L for images from t.

If a channel is relevant for two classes t_1, t_2 it is **potentially** polysemantic.

Option 1: Channel activates for a single concept shared by classes, *e.g.*, different dog species sharing the same snout.
→ **No polysemanticity**

Option 2: The channel activates for multiple concepts.
→ **Polysemanticity**

→ Distinguish options 1 and 2 by computing the average attribution of $L-1$ for images from t_1 and t_2 and measure the cosine similarity.

γ-polysemanticity: If the cosine similarity of the average $L-1$ attributions for t_1, t_2 is below γ, the channel is affected by γ-polysemanticity.

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No. 866008). The project has also been funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – project number 529680848. Further, the project has been supported by the State of Hesse, through the cluster projects "The Third Wave of Artificial Intelligence (3AI)" and "The Adaptive Mind (TAM)," and by the Hessian Ministry of Higher Education, Research, Science and the Arts and its LOEWE research priority program "WhiteBox" under grant LOEWE/2/13/519/03/06.001(0010)/77.

Disentangling Polysemantic Channels

1. Create copy of L and split polysemantic channel into two; connect edges important for t_1/t_2 with first/second channel.

Whether an edge is important for a class is determined based on the average attribution.

2. Connect unimportant edges from $L-1$ with both disentangled channels d_1, d_2 in L'.

This helps to better recover the original activation. However, unimportant edges are now weighted twice.

3. Create residual channel; connect with unimportant edges from $L-1$ and a negative weight into the original polysemantic channel in L.

The function of the CNN is **unchanged** but the polysemantic channel is **disentangled** in L'.

Results

Qualitative result from ResNet-50 trained on ImageNet:

Disentangled channel #1660

Feature visualizations of the disentangled channels are more monosemantic.

Quantitative result from ResNet-50 trained on ImageNet:

Images	First (d_1)	Second (d_2)	Residual
Class t_1	114	-9	0
Class t_2	-21	125	0

Average activations of the two disentangled and residual channels relative to the original polysemantic channel (in %) for images from class t_1 and t_2.

Quantitative results confirm that the disentangled channels are more monosemantic.

[1] Elhage et al. (2022). "Toy Models of Superposition." In: Transformer Circuits Thread (accessed May 2025)
[2] Fel et al. (2023). "Unlocking Feature Visualization for Deep Network with Magnitude Constrained Optimization." In: NeurIPS 2023