Extended Abstract Track

Extracting Reliable Concept Signals from Just a Handful of Superdetector Tokens

Editors: List of editors' names

Abstract

Concept vectors aim to connect model representations with human-interpretable semantics, but their signals are often noisy and inconsistent, limiting their reliability. In this work, we discover a new property: across labeled concept regions, concept information is concentrated in a small fraction of highly activated tokens, which we call *Superdetector Tokens*. We demonstrate that Superdetector Tokens provide more reliable concept signals than traditional concept vector and prompting methods, and enable more faithful attributions. Our results suggest that this behavior reflects a general mechanism by which transformers encode semantics, holding across image and text modalities, model families, and supervised and unsupervised extraction methods.

Keywords: Concept Vectors, Superdetector Tokens, Interpretability, Transformers, Attribution Methods, Representation Learning

1. Introduction

Modern transformer-based models, while increasingly powerful and ubiquitous, remain opaque and can behave in ways that are unpredictable or harmful. This opacity hinders our ability to identify and debug undesirable representations—such as spurious correlations, biases, or fragile reasoning—or to intervene when models produce undesirable outputs.

Concept vectors (Kim et al., 2018), or semantically meaningful directions in a model's latent space, provide a lightweight tool for examining and influencing internal representations. They have been used to uncover hidden model failures (Abid et al., 2022), and to steer model behavior away from hallucinations (Rimsky et al., 2023), unsafe responses (Liu et al., 2023), and toxic language (Turner et al., 2024). Unsupervised concept extraction is especially powerful, since labeled data is costly and such methods can uncover new scientific discoveries (Ghorbani et al., 2019).

Despite their promise, concept-based approaches face key limitations. We lack a mechanistic understanding of how models organize concept signals, and global vectors often obscure local context, introduce spurious correlations, and yield noisy activations. Evaluation is also difficult, as activation strength is hard to interpret and common validation methods (e.g., attribution, top examples) can be misleading.

In this work, we discover that transformers concentrate the bulk of concept signals in a small set of highly activated tokens, which we term *Superdetector Tokens*. We **(1)** show that these tokens signal concept presence more consistently than traditional concept vector and prompting methods, **(2)** demonstrate that they arise across image and text modalities, model families, and concept extraction techniques (supervised and unsupervised), and **(3)** leverage them to obtain more accurate and faithful attributions.

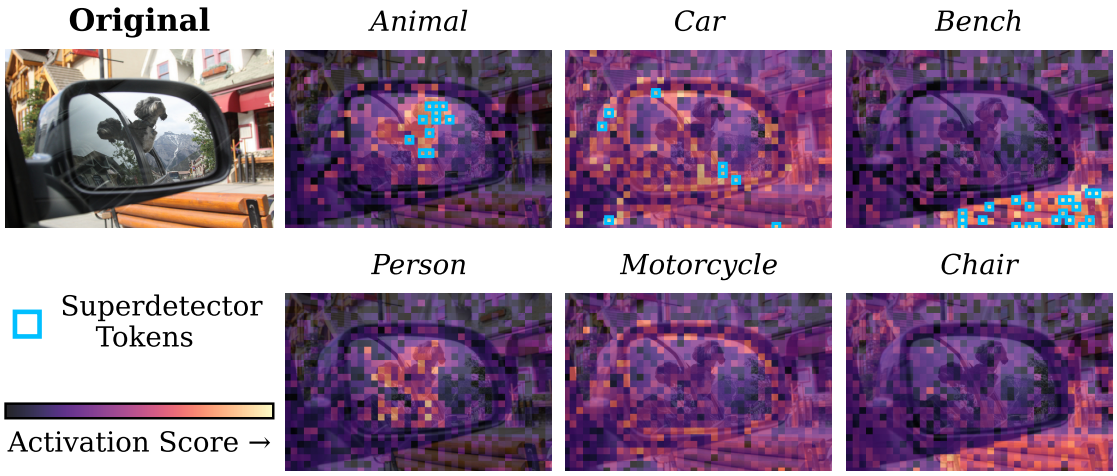

Figure 1: **Superdetector Tokens**, a sparse subset of the most highly activated tokens, correctly distinguish the true concepts in a COCO image even when other concepts appear spuriously activated.

2. Concept Vectors Are Noisy and Inconsistent

Concept vectors are directions in a model's embedding space corresponding to human-interpretable properties. For a pre-trained model f, an input x maps to $z(x) = f(x) \in \mathbb{R}^d$. In transformers, embeddings may be global (CLS/pooled) or local (patch/text). A concept c is represented by $v_c \in \mathbb{R}^d$, obtained via supervised or unsupervised methods (Appendix A.3). Alignment is given by the *activation score* $s_c(x) = \langle z(x), v_c \rangle$, with concept presence predicted as $\hat{y}_c(x) = \mathbf{1}[\, s_c(x) > \tau_c \,]$, where τ_c is tuned on validation data.

Despite their practical utility for interpretation and steering (Abid et al., 2022; Rimsky et al., 2023), concept vectors remain inconsistent and unreliable. Global concept vectors often blur context, amplify spurious correlations, and yield overlapping activations (Goh et al., 2021; Olah et al., 2020; Bricken et al., 2023). In Appendix B, we show empirically that global thresholds cannot cleanly separate in- and out-of-concept activations.

Amid this ambiguity, we find that true concept regions consistently contain a sparse set of strongly activated tokens. This motivates our central question: can ignoring weaker signals and focusing on these extremes yield more reliable concept signals?

2.1. Introducing *Superdetector Tokens*

We define *Superdetector Tokens* as the tokens whose activation scores fall within the top $N\%$ of the true concept region activations. Let $\mathcal{S}_c^+ = \{s_c(x) \mid x \text{ is a token from samples containing } c\}$ be the set of scores from true in-concept tokens in a validation set. Then

$$\mathcal{T}_c^{\text{super}} = \{x \mid s_c(x) \geq Q_{1-N}(\mathcal{S}_c^+)\},$$

where Q_{1-N} is the $(1-N)$ quantile of $\mathcal{S}_c^+$, i.e., the threshold above which the top $N\%$ of in-concept scores lie. In our experiments, N is drawn from a fixed grid (e.g., 2%–95%).

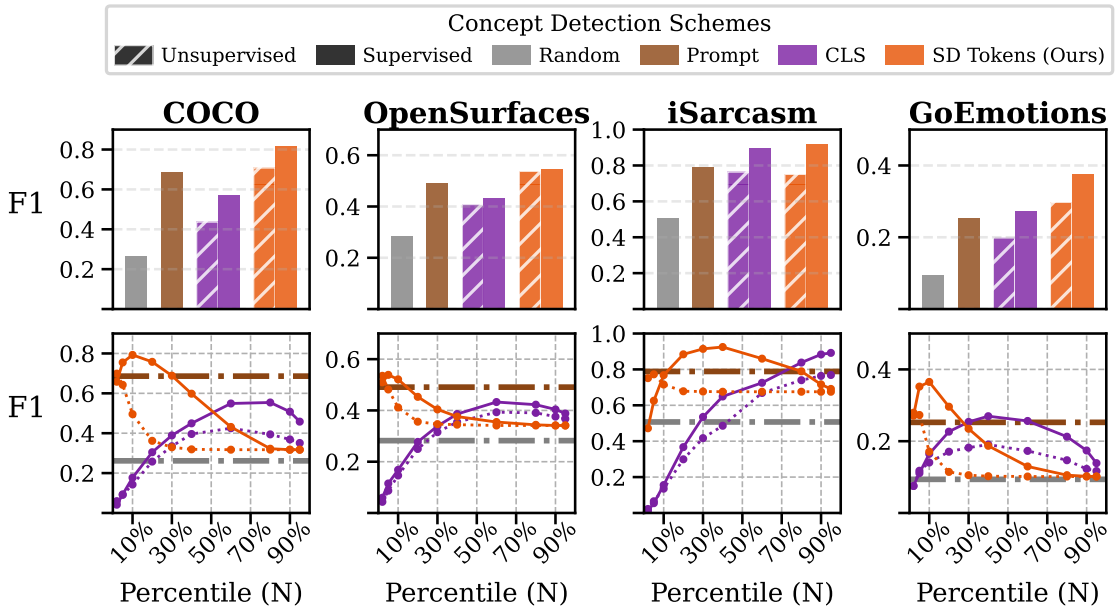

Figure 2: **Superdetector Tokens consistently detect concepts more accurately than CLS- and prompt-based methods, performing best when using only a small fraction of tokens.** Results are shown for the best-performing configuration—LLaMA with separator-based concept vectors. Top: weighted average F_1 with optimal per-concept thresholds tuned on a validation set. Bottom: weighted average F_1 as a function of the Superdetector percentile N.

The most reliable signals emerge from very small N, where only a handful of tokens per sample are selected. Despite their sparsity, these tokens appear consistently across datasets, modalities, and model families, suggesting a general property of how transformers encode concepts. In the following sections, we show that Superdetector Tokens provide a more reliable basis for detection and localization.

3. Experiments

3.1. Experimental Setup

We evaluate across four vision datasets: CLEVR (Johnson et al., 2017), COCO (Lin et al., 2014), Broden-Pascal (Bau et al., 2020), Broden-OpenSurfaces (Bau et al., 2020)), and three generated/augmented text datasets: Sarcasm, iSarcasm (Oprea and Magdy, 2020), GoEmotions (Demszky et al., 2020). Moreover, we extract embeddings from the final layers of CLIP (Radford et al., 2021), LLaMA (Meta, 2024), Gemma (Team et al., 2024), and Qwen3 (Zhang et al., 2025) transformer-based models. We utilize various concept extraction methods and a prompt baseline. See Appendix A for details.

Table 1: **Superdetector Tokens yield more accurate and faithful attributions than global vectors** on COCO and iSarcasm. Columns report inversion F_1, insertion scores (↑ more faithful), and deletion scores (↓ more faithful).

Method	Dataset	Inversion F_1		Insertion Score		Deletion Score	
		Concept	Superdetectors	Concept	Superdetectors	Concept	Superdetectors
LIME	COCO	0.49	**0.50**	0.333	**0.367**	0.010	**0.007**
	iSarcasm	0.79	**0.87**	0.383	**0.412**	0.009	**0.005**
IntGrad	COCO	0.43	**0.48**	0.326	**0.359**	0.013	**0.010**
	iSarcasm	0.75	**0.82**	0.375	**0.405**	0.011	**0.008**
MFABA	COCO	0.35	**0.39**	0.339	**0.374**	0.006	**0.004**
	iSarcasm	0.80	**0.88**	0.391	**0.420**	0.006	**0.003**

3.2. Superdetector Tokens Achieve Superior Detection Performance

For our Superdetector Token–based detection method, we say a concept is detected in a sample X if it contains at least one Superdetector Token, i.e., if $\max_{x \in X} s_c(x) \geq \tau_c^{\text{super}}$, where τ_c^{super} is the Superdetector threshold derived from a held-out validation set.

As shown in Figure 2, our Superdetector method consistently outperforms CLS and prompting baselines, with F_1 scores peaking when only a few tokens are used for detection. Unsupervised clusters that best match ground-truth concepts follow the same trend, often maximizing at just 2–5% of tokens. These results indicate that transformers concentrate concept signals in a sparse set of highly activated tokens that reliably appear across samples. This Superdetector mechanism emerges under all experimental configurations (see Appendix D), suggesting that it is a general property of transformer representations.

3.3. Superdetector Tokens Enable More Accurate and Faithful Attributions

Concept attributions (or inversions) identify the features responsible for a model's alignment with a concept, aiding in debugging and interpretation. A key requirement is *faithfulness* (Zhang et al., 2023): explanations should highlight features the model causally relies on. It is typically evaluated with insertion and deletion metrics (Gomez et al., 2022).

Standard approaches invert with respect to a single global concept vector, which blurs context and highlights irrelevant regions. We hypothesized that Superdetector Tokens, by isolating the strongest and most context-specific signals, provide a superior basis for inversion. For concept c in sample X, we define the local objective as the mean of its Superdetector Token embeddings:

$$\bar{z}_c^{\text{super}}(X) = \frac{1}{|\mathcal{T}_c^{\text{super}}(X)|} \sum_{x \in \mathcal{T}_c^{\text{super}}(X)} z(x),$$

where $\mathcal{T}_c^{\text{super}}(X)$ is the set of tokens chosen using the N that maximizes validation detection.

Table 1 shows that across three attribution methods on COCO and iSarcasm, Superdetectors consistently achieve higher F_1 with the ground truth and are more faithful. These gains also generalize across additional methods (Appendix **??**).

References

Abubakar Abid, Mert Yuksekgonul, and James Zou. Meaningfully debugging model mistakes using conceptual counterfactual explanations, 2022. URL https://arxiv.org/abs/2106.12723.

David Bau, Jun-Yan Zhu, Hendrik Strobelt, Agata Lapedriza, Bolei Zhou, and Antonio Torralba. Understanding the role of individual units in a deep neural network. *Proceedings of the National Academy of Sciences*, 2020. ISSN 0027-8424. doi: 10.1073/pnas.1907375117. URL https://www.pnas.org/content/early/2020/08/31/1907375117.

Sean Bell, Paul Upchurch, Noah Snavely, and Kavita Bala. Opensurfaces: A richly annotated catalog of surface appearance. *ACM Transactions on Graphics (SIGGRAPH)*, 32 (4), 2013.

Trenton Bricken, Adly Templeton, Jonathan Batson, Brian Chen, Adam Jermyn, Tom Conerly, and *et al.* Towards monosemanticity: Decomposing language models with dictionary learning. Anthropic Research Preprint, 2023. Available at Anthropic's website.

Hyunjin Choi, Judong Kim, Seongho Joe, and Youngjune Gwon. Evaluation of bert and albert sentence embedding performance on downstream nlp tasks, 2021. URL https://arxiv.org/abs/2101.10642.

Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. *ArXiv*, abs/2309.08600, 2023. URL https://api.semanticscholar.org/CorpusID:261934663.

Fahim Dalvi, Abdul Rafae Khan, Firoj Alam, Nadir Durrani, Jia Xu, and Hassan Sajjad. Discovering latent concepts learned in bert. *ArXiv*, abs/2205.07237, 2022. URL https://api.semanticscholar.org/CorpusID:248810913.

Dorottya Demszky, Dana Movshovitz-Attias, Jeongwoo Ko, Alan Cowen, Gaurav Nemade, and Sujith Ravi. Goemotions: A dataset of fine-grained emotions. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 4040–4054, 2020.

Teresa Dorszewski, Lenka Tvetkov'a, Robert Jenssen, Lars Kai Hansen, and Kristoffer Wickstrøm. From colors to classes: Emergence of concepts in vision transformers. *ArXiv*, abs/2503.24071, 2025. URL https://api.semanticscholar.org/CorpusID:277467666.

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, Roger Grosse, Sam McCandlish, Jared Kaplan, Dario Amodei, Martin Wattenberg, and Christopher Olah. Toy models of superposition, 2022. URL https://arxiv.org/abs/2209.10652.

Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge, 2010.

Thomas Fel, Agustin Picard, Louis Béthune, Thibaut Boissin, David Vigouroux, Julien Colin, Rémi Cadène, and Thomas Serre. Craft: Concept recursive activation factorization for explainability. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2711–2721, 2022. URL https://api.semanticscholar.org/CorpusID:253708233.

Thomas Fel, Alexandre Jullien, David Vigouroux, Remi Cadene, Thomas Nicodeme, Matthieu Laly, Asma Fermanian, Benjamin Audit, and Thomas Scantamburlo. Explaining groups of instances with shap-iq. In *International Conference on Artificial Intelligence and Statistics*, pages 6467–6491. PMLR, 2023.

Yossi Gandelsman, Alexei A. Efros, and Jacob Steinhardt. Interpreting clip's image representation via text-based decomposition. *ArXiv*, abs/2310.05916, 2023. URL https://api.semanticscholar.org/CorpusID:263829688.

Leo Gao, Tom Dupr'e la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. Scaling and evaluating sparse autoencoders. *ArXiv*, abs/2406.04093, 2024. URL https://api.semanticscholar.org/CorpusID:270286001.

Amin Ghiasi, Hamid Kazemi, Eitan Borgnia, Steven Reich, Manli Shu, Micah Goldblum, Andrew Gordon Wilson, and Tom Goldstein. What do vision transformers learn? a visual exploration. *ArXiv*, abs/2212.06727, 2022. URL https://api.semanticscholar.org/CorpusID:254591270.

Amirata Ghorbani, James Wexler, and Been Kim. Automating interpretability: Discovering and testing visual concepts learned by neural networks. *ArXiv*, abs/1902.03129, 2019. URL https://api.semanticscholar.org/CorpusID:59842921.

Gabriel Goh, Nick Cammarata †, Chelsea Voss †, Shan Carter, Michael Petrov, Ludwig Schubert, Alec Radford, and Chris Olah. Multimodal neurons in artificial neural networks. *Distill*, 2021. doi: 10.23915/distill.00030. https://distill.pub/2021/multimodal-neurons.

Tristan Gomez, Thomas Fréour, and Harold Mouchère. Metrics for saliency map evaluation of deep learning explanation methods, 2022. URL https://arxiv.org/abs/2201.13291.

Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C. Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International conference on machine learning*, pages 2668–2677. PMLR, 2018.

Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft

coco: Common objects in context. In *European Conference on Computer Vision (ECCV)*, 2014.

Sheng Liu, Haotian Ye, Lei Xing, and James Y. Zou. In-context vectors: Making in context learning more effective and controllable through latent space steering. *ArXiv*, abs/2311.06668, 2023. URL https://api.semanticscholar.org/CorpusID:265149781.

Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Advances in neural information processing systems 30*, 2017.

Divyanshu Mahajan, Chenhao Tan, and Matthew Turek. Calm: A causality-guided framework for generating local and global model explanations. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1215–1224, 2021.

Anita Mahinpei, Justin Clark, Isaac Lage, Finale Doshi-Velez, and Weiwei Pan. Promises and pitfalls of black-box concept learning models. *ArXiv*, abs/2106.13314, 2021. URL https://api.semanticscholar.org/CorpusID:235652059.

Tomas Mikolov, Wen tau Yih, and Geoffrey Zweig. Linguistic regularities in continuous space word representations. In *North American Chapter of the Association for Computational Linguistics*, 2013. URL https://api.semanticscholar.org/CorpusID:7478738.

Georgii Mikriukov, Gesina Schwalbe, Christian Hellert, and Korinna Bade. *Evaluating the Stability of Semantic Concept Representations in CNNs for Robust Explainability*, page 499–524. Springer Nature Switzerland, 2023. ISBN 9783031440670. doi: 10.1007/978-3-031-44067-0_26. URL http://dx.doi.org/10.1007/978-3-031-44067-0_26.

Angus Nicolson, Lisa Schut, J. Alison Noble, and Yarin Gal. Explaining explainability: Recommendations for effective use of concept activation vectors, 2025. URL https://arxiv.org/abs/2404.03713.

Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. Zoom in: An introduction to circuits. *Distill*, 2020. doi: 10.23915/distill.00024.001. https://distill.pub/2020/circuits/zoom-in.

OpenAI. Gpt-4o system card, 2024. URL https://openai.com/index/gpt-4o-system-card/. Model documentation and safety evaluation.

Silviu Oprea and Walid Magdy. isarcasm: A dataset of intended sarcasm. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2020.

Laura O'Mahony, Vincent Andrearczyk, Henning Müller, and Mara Graziani. Disentangling neuron representations with concept vectors. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 3770–3775, 2023. doi: 10.1109/CVPRW59228.2023.00390.

Vitali Petsiuk, Abir Das, and Kate Saenko. Rise: Randomized input sampling for explanation of black-box models. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2018.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021. URL https://arxiv.org/abs/2103.00020.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Matt Turner. Steering llama 2 via contrastive activation addition. *ArXiv*, abs/2312.06681, 2023. URL https://api.semanticscholar.org/CorpusID:266174252.

Antonio De Santis, Riccardo Campi, Matteo Bianchi, and Marco Brambilla. Visual-tcav: Concept-based attribution and saliency maps for post-hoc explainability in image classification. *ArXiv*, abs/2411.05698, 2024. URL https://api.semanticscholar.org/CorpusID:273950563.

Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

Suraj Srinivas and François Fleuret. Full-gradient representation for neural network visualization. In *Advances in Neural Information Processing Systems 32*, 2019.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International conference on machine learning*, pages 3319–3328. PMLR, 2017.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, Johan Ferret, Peter Liu, Pouya Tafti, Abe Friesen, Michelle Casbon, Sabela Ramos, Ravin Kumar, Charline Le Lan, Sammy Jerome, Anton Tsitsulin, Nino Vieillard, Piotr Stanczyk, Sertan Girgin, Nikola Momchev, Matt Hoffman, Shantanu Thakoor, Jean-Bastien Grill, Behnam Neyshabur, Olivier Bachem, Alanna Walton, Aliaksei Severyn, Alicia Parrish, Aliya Ahmad, Allen Hutchison, Alvin Abdagic, Amanda Carl, Amy Shen, Andy Brock, Andy Coenen, Anthony Laforge, Antonia Paterson, Ben Bastian, Bilal Piot, Bo Wu, Brandon Royal, Charlie Chen, Chintu Kumar, Chris Perry, Chris Welty, Christopher A. Choquette-Choo, Danila Sinopalnikov, David Weinberger, Dimple Vijaykumar, Dominika Rogozińska, Dustin Herbison, Elisa Bandy, Emma Wang, Eric Noland, Erica Moreira, Evan Senter, Evgenii Eltyshev, Francesco Visin, Gabriel Rasskin, Gary Wei, Glenn Cameron, Gus Martins, Hadi Hashemi, Hanna Klimczak-Plucińska, Harleen Batra, Harsh Dhand, Ivan Nardini, Jacinda Mein, Jack Zhou, James Svensson, Jeff Stanway, Jetha Chan, Jin Peng Zhou, Joana Carrasqueira, Joana Iljazi, Jocelyn Becker, Joe Fernandez, Joost van Amersfoort, Josh Gordon, Josh Lipschultz, Josh

Newlan, Ju yeong Ji, Kareem Mohamed, Kartikeya Badola, Kat Black, Katie Millican, Keelin McDonell, Kelvin Nguyen, Kiranbir Sodhia, Kish Greene, Lars Lowe Sjoesund, Lauren Usui, Laurent Sifre, Lena Heuermann, Leticia Lago, Lilly McNealus, Livio Baldini Soares, Logan Kilpatrick, Lucas Dixon, Luciano Martins, Machel Reid, Manvinder Singh, Mark Iverson, Martin Görner, Mat Velloso, Mateo Wirth, Matt Davidow, Matt Miller, Matthew Rahtz, Matthew Watson, Meg Risdal, Mehran Kazemi, Michael Moynihan, Ming Zhang, Minsuk Kahng, Minwoo Park, Mofi Rahman, Mohit Khatwani, Natalie Dao, Nenshad Bardoliwalla, Nesh Devanathan, Neta Dumai, Nilay Chauhan, Oscar Wahltinez, Pankil Botarda, Parker Barnes, Paul Barham, Paul Michel, Pengchong Jin, Petko Georgiev, Phil Culliton, Pradeep Kuppala, Ramona Comanescu, Ramona Merhej, Reena Jana, Reza Ardeshir Rokni, Rishabh Agarwal, Ryan Mullins, Samaneh Saadat, Sara Mc Carthy, Sarah Cogan, Sarah Perrin, Sébastien M. R. Arnold, Sebastian Krause, Shengyang Dai, Shruti Garg, Shruti Sheth, Sue Ronstrom, Susan Chan, Timothy Jordan, Ting Yu, Tom Eccles, Tom Hennigan, Tomas Kocisky, Tulsee Doshi, Vihan Jain, Vikas Yadav, Vilobh Meshram, Vishal Dharmadhikari, Warren Barkley, Wei Wei, Wenming Ye, Woohyun Han, Woosuk Kwon, Xiang Xu, Zhe Shen, Zhitao Gong, Zichuan Wei, Victor Cotruta, Phoebe Kirk, Anand Rao, Minh Giang, Ludovic Peran, Tris Warkentin, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, D. Sculley, Jeanine Banks, Anca Dragan, Slav Petrov, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena Buchatskaya, Sebastian Borgeaud, Noah Fiedel, Armand Joulin, Kathleen Kenealy, Robert Dadashi, and Alek Andreev. Gemma 2: Improving open language models at a practical size, 2024. URL https://arxiv.org/abs/2408.00118.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. Steering language models with activation engineering, 2024. URL https://arxiv.org/abs/2308.10248.

Zhengxuan Wu, Aryaman Arora, Atticus Geiger, Zheng Wang, Jing Huang, Dan Jurafsky, Christopher D. Manning, and Christopher Potts. Axbench: Steering llms? even simple baselines outperform sparse autoencoders, 2025. URL https://arxiv.org/abs/2501.17148.

Xuemin Yu, Fahim Dalvi, Nadir Durrani, and Hassan Sajjad. Latent concept-based explanation of nlp models. *ArXiv*, abs/2404.12545, 2024. URL https://api.semanticscholar.org/CorpusID:269282778.

Lihua Zhang and Shihua Zhang. A unified joint matrix factorization framework for data integration. *ArXiv*, abs/1707.08183, 2017. URL https://api.semanticscholar.org/CorpusID:21228616.

Yang Zhang, Yawei Li, Hannah Brown, Mina Rezaei, Bernd Bischl, Philip Torr, Ashkan Khakzar, and Kenji Kawaguchi. Attributionlab: Faithfulness of feature attribution under controllable environments. *arXiv preprint arXiv:2310.06514*, 2023. URL https://arxiv.org/abs/2310.06514.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. Qwen3 embed-

ding: Advancing text embedding and reranking through foundation models, 2025. URL https://arxiv.org/abs/2506.05176.

Hong Zhou, Rui Zhang, Peifeng Lai, Chaoran Guo, Yong Wang, Zhida Sun, and Junjie Li. El-vit: Probing vision transformer with interactive visualization, 2024. URL https://arxiv.org/abs/2401.12666.

Zhiyu Zhu, Huaming Chen, Jiayu Zhang, Xinyi Wang, Zhibo Jin, Minhui Xue, Dongxiao Zhu, and Kim-Kwang Raymond Choo. Mfaba: A more faithful and accelerated boundary-based attribution method for deep neural networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(15):17228–17236, 2024. doi: 10.1609/aaai.v38i15.29669.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, Shashwat Goel, Nathaniel Li, Michael J. Byun, Zifan Wang, Alex Troy Mallen, Steven Basart, Sanmi Koyejo, Dawn Song, Matt Fredrikson, Zico Kolter, and Dan Hendrycks. Representation engineering: A top-down approach to ai transparency. *ArXiv*, abs/2310.01405, 2023.

Appendix A. Datasets

A.1. Dataset Specifics

CLEVR (Single-Object)(Johnson et al., 2017): A synthetic dataset of 1,000 images, each containing a red, green, or blue object with shape sphere, cylinder, or cube. Images and segmentation masks are generated programmatically, allowing fine-grained control over object properties and patch-level annotations.

COCO(Lin et al., 2014): We use the 2017 validation set, containing 5,500 images with everyday scenes involving people, objects, and natural contexts. Each image comes with human-annotated segmentations, providing dense labels for both object categories and broader supercategories.

Broden–Pascal (Everingham et al., 2010) and Broden–OpenSurfaces (Bell et al., 2013): We use 4,503 samples from Pascal and 3,578 samples from OpenSurfaces. These are subsets of the Broden dataset (Bau et al., 2020), which unifies multiple segmentation datasets into a single benchmark for concept-based interpretability research. Pascal primarily contains natural images with segmented objects from diverse categories such as animals, vehicles, and household items, while OpenSurfaces emphasizes fine-grained material and surface property annotations (e.g., wood, fabric, metal). These subsets focus on patch-level segmentation where concepts do not necessarily span the entire image.

Sarcasm (Fully Synthetic): We generate a dataset of 1,446 paragraphs, where roughly half contain exactly one sarcastic sentence surrounded by neutral sentences.

iSarcasm (Augmented): We adapt 1,734 samples from the original iSarcasm dataset (Oprea and Magdy, 2020), which provides sarcastic tweets alongside non-sarcastic rewrites conveying the same meaning (both provided by the original authors). We augment these by embedding sarcastic and non-sarcastic sentences into short paragraphs of neutral context, with sarcastic spans explicitly marked.

GoEmotions (Augmented): We use 5,427 samples from the GoEmotions dataset (Demszky et al., 2020), a human-annotated collection of Reddit comments labeled with 27 emotion categories. We augment selected samples by embedding emotional sentences within surrounding neutral context, tagging the emotional span while preserving natural paragraph flow.

A.2. Text Augmentation Pipelines and Prompts

This section describes the augmentation pipelines used for generating and adapting text datasets, along with the exact prompts. Our goal was to create datasets with localized token-level concept spans, since most publicly available text datasets only provide unit-level (sentence, tweet, comment, etc) labels. Generation and augmentation are performed via controlled prompting of GPT-4o (OpenAI, 2024).

A.2.1. Sarcasm (Fully Synthetic)

Pipeline: We generate entirely new paragraphs containing exactly one sarcastic sentence. The sarcastic sentence is wrapped in `<SARCASM>` tags, while all other sentences are neutral. This ensures that each paragraph contains exactly one labeled sarcastic span, with natural context surrounding it. By constraining sarcastic content to a single line, we obtain a controlled setup where token-level supervision is precise and unambiguous.

Prompt:

```
Write 10 short paragraphs (4{8 sentences each). Each paragraph must include
**exactly one sarcastic sentence**, wrapped in <SARCASM> ... </SARCASM> tags.

Guidelines:
- The sarcastic sentence should be subtle, deadpan, or context-dependent.
- All other sentences must be sincere and literal.
- Vary topic, tone, and structure across paragraphs.

Only the sarcastic line may be wrapped in tags.

Return only the 10 numbered paragraphs.
```

Example: Jane always prided herself on her cooking abilities. `<SARCASM>`Indeed, the local fire department must have also appreciated her culinary exploits, given the number of times they've had to rush to her house.`</SARCASM>` Still, she was not deterred and continued to experiment in the kitchen, determined to perfect her skills. She understood that learning anything new involved a process of trial and error.

A.2.2. iSarcasm Augmentation

Dataset Overview: The original iSarcasm dataset contains sarcastic tweets paired with author-provided sincere rewrites conveying the same meaning. We extend this dataset synthetically by surrounding the sarcastic tweets with literal, neutral context, ensuring precise span-level supervision. Only sarcastic samples are selected for augmentation, and

for each sarcastic input we generate both a sarcastic augmented post and a non-sarcastic rewrite.

Augmentation Pipeline: Each sarcastic input is expanded into casual, paragraph-like text using controlled prompting of GPT-4.0. To introduce variation, random structural features are applied:

- 20% chance of forcing a `[Sarcasm][Trigger]` structure.
- 15% chance of adding emojis or hashtags.
- Otherwise, a random choice among `[Sarcasm][Trigger]`, `[Trigger][Sarcasm]`, or `[Trigger][Sarcasm][Trigger]`.

Sarcastic Augmentation Prompt:

```
You are a data annotation machine. Your only goal is to produce perfectly literal
text that follows the rules. You must not be creative or clever. You must not
generate any figurative language outside of the provided tags.

Your Task:
You will be given a sarcastic tweet and its true meaning. Rewrite the tweet by
embedding it within a strictly literal train of thought that matches the original's
casual tone.

Structure: [Randomly choose or force specific structure]
[Optional emoji/hashtag instruction if selected]

Constraints Checklist:
- The tone is casual and informal.
- The added text is not redundant.
- Outside <SARCASM> tags is strictly literal and descriptive.
- The original sarcastic tweet is fully preserved within <SARCASM> tags.
- Output contains ONLY the final post.

Input Sarcastic Tweet: "{sarcastic_tweet}"
Sincere Meaning (for your context): "{rephrased_text}"

Your Output:
```

Non-Sarcastic Augmentation Prompt.

```
You are a data annotation machine. Your only goal is to produce perfectly literal
text that follows the rules. You must not be creative or clever. You must not
invent new details.

Your Task:
Take a sincere idea and expand it slightly into a personal, casual post,
```

```
remaining 100% faithful to the original meaning.

[Optional emoji/hashtag instruction if selected]

Constraints Checklist:
- The tone is casual and informal.
- The entire post is strictly literal and descriptive.
- No sarcasm, irony, overstatement, or rhetorical questions.
- The post must be 100% faithful to the meaning of the original idea.
- Output contains ONLY the final post.

Input Sincere Idea: "{rephrased_text}"

Your Output:
```

Verification Process: Outputs are verified via flexible matching with progressively lenient checks: exact matching (case-insensitive), whitespace normalization, URL/punctuation removal, and word-overlap thresholds. If all attempts fail, the original tweet is wrapped in `<SARCASM>` tags as a fallback.

Example:

> **Input sarcastic tweet:** "The only thing I got from college is a caffeine addiction."
> **Input sincere rephrase:** "College is really difficult, expensive, tiring, and I often question if a degree is worth the stress."
>
> **Sarcastic augmentation:** "I just checked my calendar and saw how many assignments are due this week. ¡SARCASM¿the only thing i got from college is a caffeine addiction¡/SARCASM¿"
> **Non-sarcastic rewrite:** "college is really difficult. it's also expensive and tiring. sometimes i find myself questioning if getting a degree is worth all the stress."

A.2.3. GoEmotions Augmentation

Dataset Overview: GoEmotions is a large-scale dataset of Reddit comments labeled with up to 27 fine-grained emotions. We extend it synthetically by surrounding the original emotional comment with strictly neutral filler context, ensuring the emotional span remains localized and clearly marked with `<EMOTION>` tags.

Augmentation Pipeline: Every comment in GoEmotions is augmented without filtering, following a two-step process:

1. **Step 1: Generation.** A "Neutral Filler Machine" prompt is used to generate five diverse neutral-context options embedding the original emotional comment.
2. **Step 2: Selection.** A "Grader" prompt evaluates the five drafts and selects the best single option according to neutrality and naturalness.

To increase variation, a random structure is sampled per comment:

- 50% chance: `[Emotion][Context]`
- 25% chance: `[Context][Emotion]`
- 25% chance: `[Context][Emotion][Context]`

Step 1 — Neutral Filler Prompt:

```
You are a Neutral Filler Machine. Your task is to generate neutral,
non-emotional text to surround a given Reddit comment.

Task:
- Preserve the original emotional comment exactly inside <EMOTION> tags.
- Generate five unique and diverse neutral contexts that flow naturally.
- All options must follow the required structure.

Constraints:
- Text outside <EMOTION> must be strictly neutral (no emotion leakage).
- Sound natural and casual like a Reddit post.
- No redundancy with the emotional comment.

Input Emotional Comment: "{emotional_comment}"
Primary Emotion(s): "{emotion_labels_str}"
Required Structure: "{structure_choice}"

Your Output: Five options, each in the correct structure.
```

Step 2 — Selection Prompt.

```
You are a data annotation quality assurance specialist.
Your task is to select the best draft among five options.

Checklist:
- Context must be strictly neutral (no emotions).
- Flow naturally as a Reddit comment.
- No contradiction or redundancy.
- Only output the single best final option.

Draft Options:
{draft_options}

Your Final, Best Output:
```

Verification Process: The augmented comments are verified using flexible string matching to ensure that the original text is preserved inside `<EMOTION>` tags. We allow up to five retry attempts with progressively lenient checks. If all attempts fail, the fallback is to wrap the original comment directly in `<EMOTION>` tags.

Example:

> **Original emotional comment (gratitude):** "I didn't know that, thank you for teaching me something today!"
>
> **Augmented output:** "A comment explained the process behind recycling plastics and how it affects the environment. ¡EMOTION¿I didn't know that, thank you for teaching me something today!¡/EMOTION¿"

A.3. Concept Extraction Details

We use the following concept extraction methods in our work:

Supervised Methods: We compute concept from labeled data by (1) averaging embeddings of concept-positive examples to form a prototype vector v_c (Zou et al., 2023), and by (2) training a linear classifier and using the normal vector of the separating hyperplane as a Concept Activation Vector (Kim et al., 2018).

Unsupervised Methods: We obtain concepts in an unsupervised manner by (1) clustering embeddings with K-means and using cluster centroids as prototypes (with k=1000 tokens, k=50 CLS, determined experimentally) (Ghorbani et al., 2019; Dalvi et al., 2022), and by (2) training linear classifiers between clusters to yield separator-style directions. We incorporate the discovered clusters into our evaluation by matching each ground-truth concept with the unsupervised unit that is most reliable at detecting it.

Prompt Baseline: We prompt LLaMA-3.2-11B-Vision-Instruct 'Is c concept present in sample X?' for each concept individually.

Models: We extract embeddings from the final layers of the following transformer-based models: CLIP ViT-L/14 (Radford et al., 2021) and LLaMA-3.2-11B-Vision-Instruct for images, as well as LLaMA-3.2-11B-Vision-Instruct, Gemma-2-9B (Team et al., 2024), and Qwen3-Embedding-4B for text.

A.4. Concepts Used in Experiments

- **COCO:** accessory, animal, appliance, bench, book, bottle, bowl, bus, car, chair, couch, cup, dining table, electronic, food, furniture, indoor, kitchen, motorcycle, outdoor, person, pizza, potted plant, sports, train, truck, tv, umbrella, vehicle.
- **Broden–Pascal:** object::airplane, object::bicycle, object::bird, object::boat, object::body, object::book, object::building, object::bus, object::cap, object::car, object::cat, object::cup, object::dog, object::door, object::ear, object::engine, object::grass, object::hair, object::horse, object::leg, object::mirror, object::motorbike, object::mountain, object::painting, object::person, object::pottedplant, object::saddle, object::screen, object::sky, object::sofa, object::table, object::track, object::train, object::tvmonitor, object::wheel, object::wood, part::arm, part::bag, part::beak, part::bottle, part::box, part::cabinet, part::ceiling, part::chain wheel, part::chair, part::coach, part::curtain, part::eye, part::eyebrow, part::fabric, part::fence, part::floor, part::foot, part::ground, part::hand, part::handle bar, part::head, part::headlight, part::light, part::mouth, part::muzzle, part::neck, part::nose, part::paw, part::plant, part::plate, part::plaything, part::pole, part::pot, part::road, part::rock, part::rope, part::shelves, part::sidewalk, part::signboard, part::stern, part::tail, part::torso, part::tree, part::wall, part::water, part::windowpane, part::wing.

- **Broden–OpenSurfaces:** material::brick, material::cardboard, material::carpet, material::ceramic, material::concrete, material::fabric, material::food, material::fur, material::glass, material::granite, material::hair, material::laminate, material::leather, material::metal, material::mirror, material::painted, material::paper, material::plastic-clear, material::plastic-opaque, material::rock, material::rubber, material::skin, material::tile, material::wallpaper, material::wicker, material::wood.
- **CLEVR:** color::blue, color::green, color::red, shape::cube, shape::cylinder, shape::sphere.
- **Sarcasm:** sarcasm.
- **iSarcasm:** sarcastic.
- **GoEmotions:** confusion, joy, sadness, anger, love, caring, optimism, amusement, curiosity, disapproval, approval, annoyance, gratitude, admiration.

Appendix B. Global Thresholds are Incompatible with Transformer Concept Signals

We show on examples of OpenSurfaces and iSarcasm that transformers do not distribute concept alignment evenly across ground truth regions.

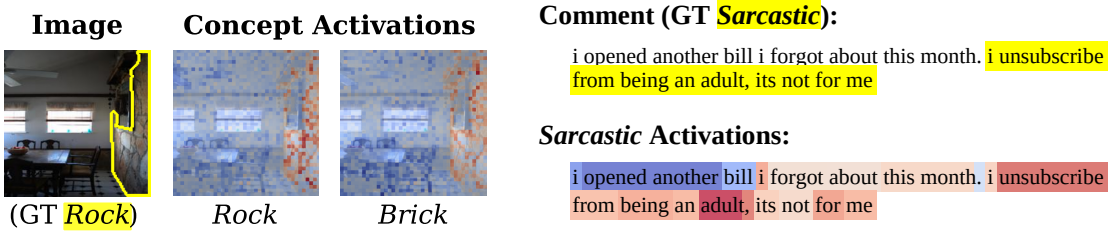

Figure 3: Concept activation heatmaps for OpenSurfaces (left) and iSarcasm (right), with high-activation tokens shown in red. Two issues are evident: **(1)** token activations within true concept regions are highly variable, with many in-concept tokens exhibiting weaker activations than out-of-concept tokens; and **(2)** semantically related concepts (e.g., *Rock* vs. *Brick*) co-activate on the same region, even when only one is present (here, *Rock*).

In Figure 3, it is evident that activations within true concept vary significantly. Moreover, the OpenSurfaces example illustrates the ambiguity that arises when entangled concepts co-activate.

We next empirically show that global thresholds cannot effectively separate in-concept and out-of-concept activations. Figure 4 presents token-level results for the top-performing Pascal concepts (measured by F_1 alignment with labels), while Figure 5 presents token-level results for the median-performing Pascal concepts.

The distributions show that a considerable portion of the in-concept and out-of-concept activation scores occupy the same region, with the F_1-optimal threshold often positioned

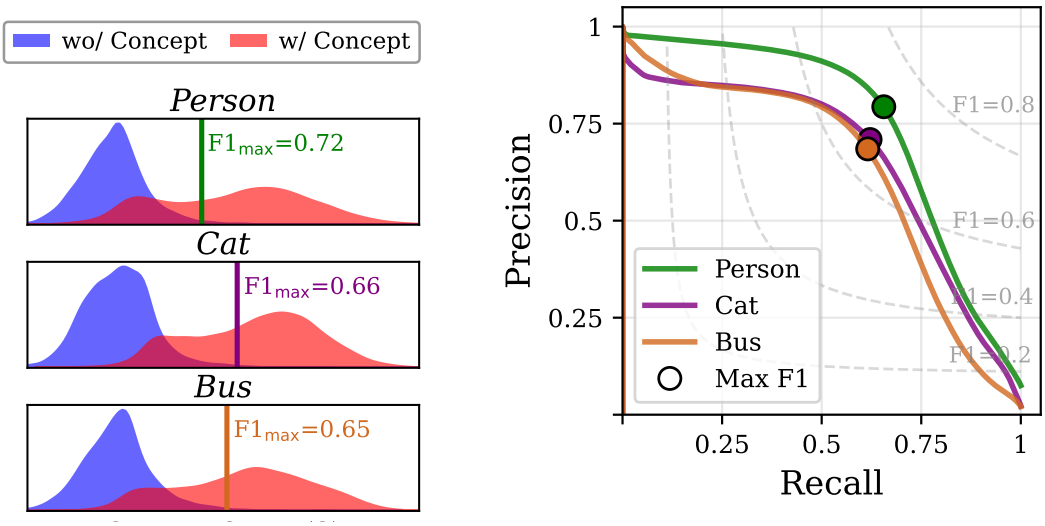

Figure 4: **Token-level concept scores reveal substantial overlap between in-concept and out-of-concept distributions, limiting the effectiveness of per-concept global thresholds.** The top three Pascal concepts are shown with score distributions (left) for in-concept (red) and out-of-concept (blue) samples, each normalized to unit area, as well as Precision–Recall curves (right) for the same concepts. The max F_1 thresholds demarcated on each plot represent the best possible global separator for that concept.

near the middle of the in-concept distribution. Precision–Recall curves confirm that precision deteriorates rapidly once recall exceeds roughly 60%, indicating that recovering most concept instances comes at the cost of many false positives. Concepts computed via CLS tokens also struggle to balance precision and recall, as shown in Figure 6.

In line with the qualitative observations introduced previously, these results reinforce our assertion that the extremes of in-concept and out-of-concept activations are reliable, but mid-range activations remain ambiguous. Consequently, global thresholding is fundamentally misaligned with the noisy manner in which transformers encode concept signals, making it an unreliable basis for concept detection.

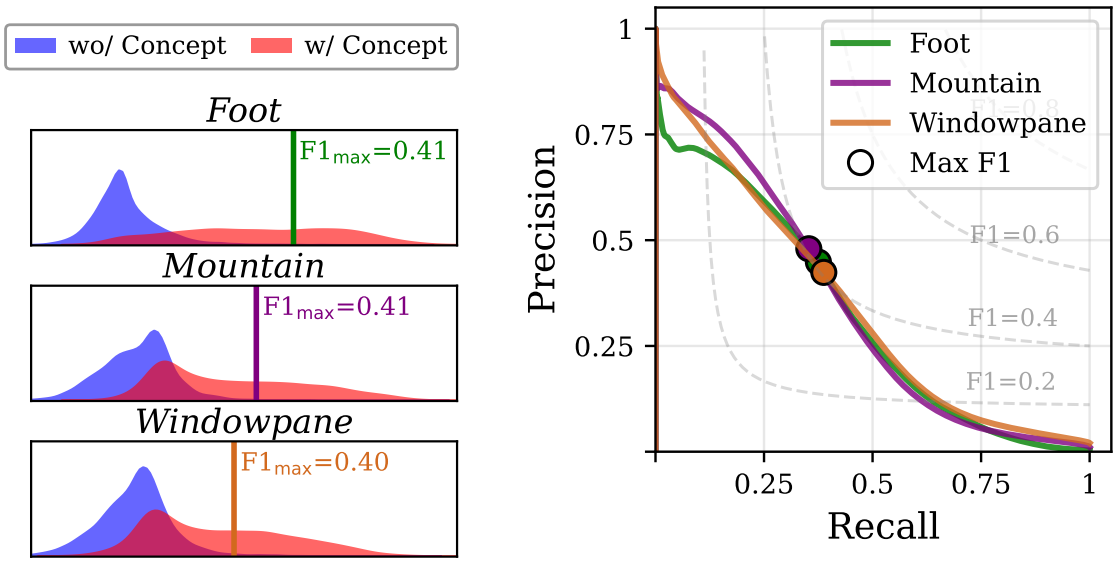

Figure 5: Global threshold limitations for medium-performing concepts in PASCAL. Left: activation histogram. Right: precision–recall curves under different thresholds.

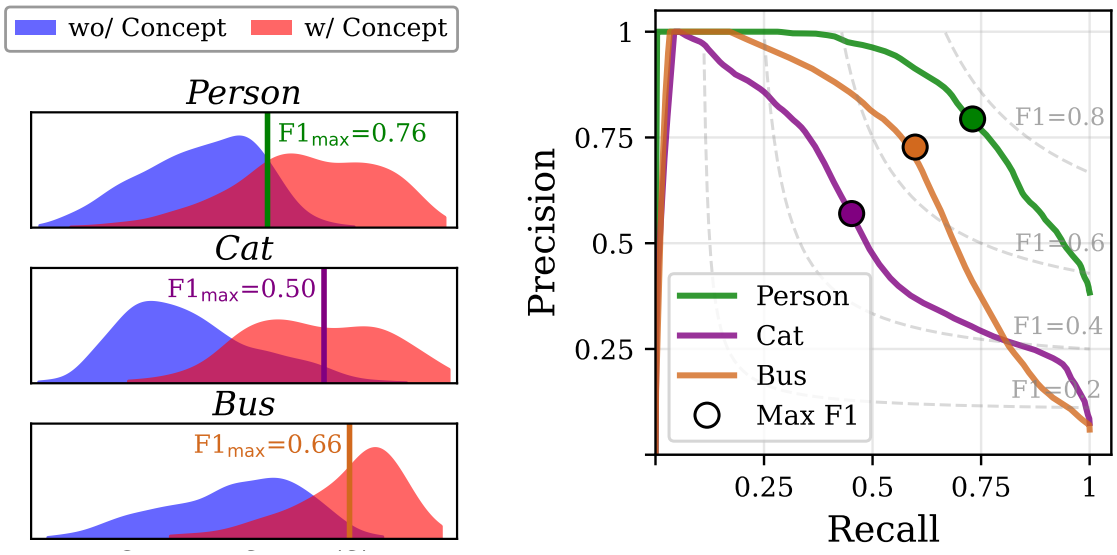

Figure 6: Global threshold limitations for best-performing concepts extracted from CLS embeddings in PASCAL. Left: activation histogram. Right: precision–recall curves under different thresholds.

Appendix C. Text Superdetector Example

Figure C shows an example of Superdetector Tokens on the GoEmotions dataset. Concepts *Anger* and *Love* both appear to be activated, while the *Superdetector Token* correctly distinguish the correct concept, *Anger*.

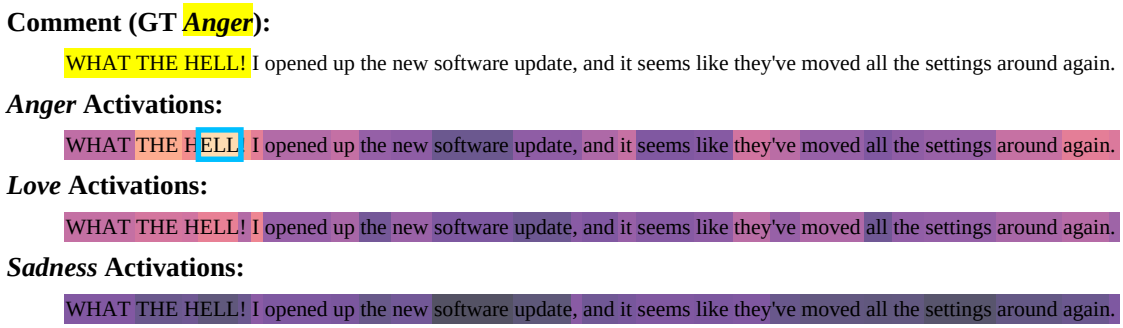

Figure 7: Example of Superdetector Tokens on GoEmotions example.

Appendix D. Comprehensive Detection Results

The following tables show the F_1 detection scores for all models, sample type (Superdetector Token vs CLS), and concept extracton method (mean prototype, linear separator, K-Means, linear separator on K-Means clusters) across datasets, sorted from best to worst. Our Superdetector Token method performs the best on all configurations.

CLEVR: Detection Performance

Model	Sample Type	Method	F1
Llama	**Token**	**KMeans**	**0.994**
Llama	N/A	Prompt	0.987
Llama	Token	KMeans LinSep	0.986
CLIP	CLS	Average	0.972
CLIP	CLS	LinSep	0.972
Llama	Token	LinSep	0.928
Llama	Token	Average	0.913
CLIP	Token	KMeans	0.897
CLIP	Token	LinSep	0.879
Llama	CLS	Average	0.873
Llama	CLS	LinSep	0.872
CLIP	Token	Average	0.840
Llama	CLS	KMeans LinSep	0.836
CLIP	Token	KMeans LinSep	0.836
CLIP	CLS	KMeans	0.816
CLIP	CLS	KMeans LinSep	0.809
Llama	CLS	KMeans	0.802
N/A	N/A	Random	0.395

COCO: Detection Performance

Model	Sample Type	Method	Best F1
Llama	**Token**	**LinSep**	**0.815**
CLIP	Token	LinSep	0.775
Llama	Token	KMeans	0.718
Llama	Token	KMeans LinSep	0.708
Llama	Token	Average	0.699
CLIP	CLS	LinSep	0.698
CLIP	Token	KMeans	0.686
Llama	N/A	Prompt	0.686
CLIP	Token	KMeans LinSep	0.685
CLIP	Token	Average	0.679
CLIP	CLS	Average	0.679
CLIP	CLS	KMeans	0.580
CLIP	CLS	KMeans LinSep	0.573
Llama	CLS	LinSep	0.571
Llama	CLS	Average	0.536
Llama	CLS	KMeans LinSep	0.437
Llama	CLS	KMeans	0.427
N/A	N/A	Random	0.262

OpenSurfaces: Detection Performance

Model	Sample Type	Method	F1
Llama	**Token**	**LinSep**	**0.546**
Llama	Token	KMeans	0.545
CLIP	Token	LinSep	0.541
Llama	Token	KMeans LinSep	0.538
CLIP	Token	KMeans	0.534
CLIP	Token	KMeans LinSep	0.527
Llama	Token	Average	0.516
CLIP	Token	Average	0.505
CLIP	CLS	LinSep	0.495
Llama	N/A	Prompt	0.491
CLIP	CLS	Average	0.484
CLIP	CLS	KMeans	0.449
CLIP	CLS	KMeans LinSep	0.448
Llama	CLS	LinSep	0.430
Llama	CLS	Average	0.425
Llama	CLS	KMeans	0.409
Llama	CLS	KMeans LinSep	0.408
N/A	N/A	Random	0.285

Pascal: Detection Performance

Model	Sample Type	Method	F1
CLIP	**Token**	**LinSep**	**0.812**
Llama	Token	LinSep	0.799
Llama	Token	KMeans LinSep	0.760
Llama	Token	Average	0.756
Llama	Token	KMeans	0.741
CLIP	Token	Average	0.735
CLIP	CLS	LinSep	0.729
CLIP	Token	KMeans	0.726
CLIP	Token	KMeans LinSep	0.706
CLIP	CLS	Average	0.703
Llama	N/A	Prompt	0.680
CLIP	CLS	KMeans	0.632
CLIP	CLS	KMeans LinSep	0.625
Llama	CLS	LinSep	0.619
Llama	CLS	Average	0.584
Llama	CLS	KMeans	0.538
Llama	CLS	KMeans LinSep	0.529
N/A	N/A	Random	0.308

Sarcasm: Detection Performance

Model	Sample Type	Method	F1
Llama	**Token**	**LinSep**	**0.805**
Qwen	Token	LinSep	0.800
Gemma	Token	LinSep	0.775
Llama	Token	KMeans LinSep	0.754
Llama	Token	KMeans	0.750
Llama	CLS	LinSep	0.716
Qwen	Token	Average	0.689
Qwen	Token	KMeans LinSep	0.689
Llama	CLS	KMeans LinSep	0.688
Gemma	Token	Average	0.681
Llama	CLS	Average	0.680
Llama	N/A	Prompt	0.679
Gemma	Token	KMeans	0.678
Qwen	CLS	LinSep	0.677
Llama	Token	Average	0.673
Gemma	Token	KMeans LinSep	0.672
Qwen	CLS	KMeans LinSep	0.671
Qwen	CLS	KMeans	0.670
Gemma	CLS	Average	0.670
Gemma	CLS	LinSep	0.670
Gemma	CLS	KMeans LinSep	0.665
Qwen	CLS	Average	0.663
Llama	CLS	KMeans	0.660
Gemma	CLS	KMeans	0.655
Qwen	Token	KMeans	0.639
N/A	N/A	Random	0.521

Augmented iSarcasm: Detection Performance

Model	Sample Type	Method	F1
Llama	**Token**	**LinSep**	**0.914**
Gemma	CLS	LinSep	0.912
Gemma	Token	LinSep	0.900
Llama	CLS	LinSep	0.892
Gemma	CLS	Average	0.886
Llama	CLS	Average	0.867
Qwen	Token	LinSep	0.839
Llama	Token	Average	0.818
Gemma	CLS	KMeans	0.814
Gemma	CLS	KMeans LinSep	0.810
Qwen	CLS	LinSep	0.797
Gemma	Token	KMeans	0.790
Llama	N/A	Prompt	0.789
Gemma	Token	KMeans LinSep	0.784
Qwen	CLS	Average	0.774
Llama	CLS	KMeans	0.772
Llama	CLS	KMeans LinSep	0.764
Qwen	Token	KMeans LinSep	0.761
Gemma	Token	Average	0.754
Qwen	Token	KMeans	0.751
Llama	Token	KMeans LinSep	0.751
Qwen	CLS	KMeans LinSep	0.741
Qwen	CLS	KMeans	0.730
Qwen	Token	Average	0.711
Llama	Token	KMeans	0.699
N/A	N/A	Random	0.507

Augmented GoEmotions: Detection Performance

Model	Sample Type	Method	F1
Qwen	**Token**	**LinSep**	**0.431**
Qwen	Token	KMeans	0.401
Qwen	Token	KMeans LinSep	0.397
Llama	Token	LinSep	0.375
Qwen	Token	Average	0.350
Gemma	Token	LinSep	0.315
Llama	Token	KMeans	0.299
Llama	Token	KMeans LinSep	0.298
Gemma	CLS	LinSep	0.279
Qwen	CLS	LinSep	0.276
Llama	CLS	Average	0.273
Llama	CLS	LinSep	0.273
Gemma	CLS	Average	0.268
Qwen	CLS	Average	0.260
Llama	N/A	Prompt	0.252
Llama	Token	Average	0.252
Gemma	Token	KMeans LinSep	0.244
Gemma	Token	KMeans	0.237
Gemma	Token	Average	0.235
Llama	CLS	KMeans LinSep	0.199
Llama	CLS	KMeans	0.195
Qwen	CLS	KMeans LinSep	0.168
Qwen	CLS	KMeans	0.168
Gemma	CLS	KMeans	0.146
Gemma	CLS	KMeans LinSep	0.144
N/A	N/A	Random	0.094

Appendix E. Concept Inversion

E.1. Inversion Methods

This section provides a brief overview of several inversion methods where the inversion objective is either a global concept vector v_c or the average embedding of local Superdetector Tokens.

- **LIME (Local Interpretable Model-agnostic Explanations)** (Ribeiro et al., 2016) explains an individual prediction by approximating the complex model with a simpler, interpretable model (e.g., a linear model) in the local vicinity of the prediction. It achieves this by generating a new dataset of perturbed samples around the instance being explained and learning the simpler model on this new dataset, weighted by proximity to the original instance.
- **SHAP (SHapley Additive exPlanations)** (Lundberg and Lee, 2017) assigns an importance value to each feature for a particular prediction. Based on cooperative game theory, this value represents the feature's marginal contribution to the model's output, ensuring the sum of all values explains the difference between the model's prediction and a baseline.
- **RISE (Randomized Input Sampling for Explanation)** (Petsiuk et al., 2018) generates a visual explanation by probing the model with numerous randomly masked versions of an input image. The final importance map is a weighted average of these random masks, where weights are determined by the model's output confidence for each corresponding masked image.
- **SHAP IQ (SHAP Interaction-aware exPlanations for Quantifying feature importance)** (Fel et al., 2023) extends the SHAP framework to quantify the effects of feature interactions. Beyond calculating the main effect of each feature, it also computes interaction indices to provide a more complete picture of how combinations of features jointly influence a prediction.
- **IntGrad (Integrated Gradients)** (Sundararajan et al., 2017) calculates the importance of each input feature by integrating the gradients of the model's output with respect to the feature's inputs. This integration is performed along a straight-line path from a baseline input (e.g., a black image) to the actual input, satisfying key axioms like sensitivity.
- **Grad-CAM (Gradient-weighted Class Activation Mapping)** (Selvaraju et al., 2017) produces a coarse localization map for CNNs by using the gradients of the target class score with respect to the feature maps of the final convolutional layer. These gradients are used to compute a weighted combination of the activation maps, highlighting important image regions.
- **FullGrad** (Srinivas and Fleuret, 2019) enhances gradient-based explanations by aggregating gradient information from all layers of a neural network. It combines the input gradients with bias gradients from all intermediate feature maps to capture more comprehensive feature representations, resulting in more detailed saliency maps.

- **CALM (Class Activation Latent Mapping)** (Mahajan et al., 2021) improves on Class Activation Mapping (CAM) by introducing a probabilistic latent variable that directly represents the location of the most important visual cue for a model's prediction. Trained with the Expectation-Maximization (EM) algorithm, the method outputs a probability map showing the likelihood that each pixel is the critical cue for the decision.

- **MFABA (More Faithful and Accelerated Boundary-based Attribution)** (Zhu et al., 2024) is an attribution method that uses gradient ascent to generate a path from an original input to an adversarial sample across the model's decision boundary. Its key feature is the use of a second-order Taylor expansion to more accurately model the change in the model's loss function.

E.2. Additional Results for Concept Inversion

This section presents the full results for concept inversion across all experimental configurations, which were summarized in Table **??** in the main text. These detailed tables are provided to demonstrate that our main findings are consistent across all individual concepts and experimental settings. As these results confirm, using the average embedding of Superdetector Tokens as the inversion objective consistently leads to better performance than using the concept vector directly. Moreover, linear separators generally outperform simple clustering for concept representation.

We present results across four tables, evaluating two concept representations (clustering-based vs. linear separators) and two inversion objectives (global concept vector vs. average local superdetector patch embedding). Each table reports the average F1 score across all concepts, weighted by the frequency of the concepts in the test set (Appendix A.4). The tables are organized as follows:

- Tables 2 & 3: Results for the supervised setting on vision and text tasks, respectively. The concept types used for supervised settings are detailed in Appendix A.3.

- Tables 4 & 5: Results for the unsupervised setting on vision and text tasks, respectively. Here, concepts are derived from k-means clusters, and for each concept, we evaluate the best-performing cluster out of 1000 candidates. The concept types are detailed in Appendix A.3.

Table 2: Average F1 for Supervised (Image Datasets)

Inversion Method	Concept Type	Inversion Objective	CLEVR		COCO		OpenSurfaces		Pascal	
			CLIP	LLaMA	CLIP	LLaMA	CLIP	LLaMA	CLIP	LLaMA
Random			0.04	0.03	0.10	0.09	0.06	0.05	0.09	0.08
CosSim	Cluster	Concept	0.60	0.78	0.43	0.36	0.22	0.19	0.42	0.40
		Supertok	0.60	0.55	0.40	0.37	0.18	0.15	0.35	0.29
	LinSep	Concept	0.65	0.85	0.52	0.46	0.28	0.23	0.54	0.46
		Supertok	0.61	0.54	0.45	0.44	0.22	0.17	0.42	0.33
LIME	Cluster	Concept	0.49	0.76	0.32	0.47	0.42	0.55	0.50	0.69
		Supertok	0.55	0.81	0.38	0.51	0.50	0.62	0.52	0.71
	LinSep	Concept	0.49	0.70	0.29	0.49	0.46	0.60	0.51	0.71
		Supertok	0.68	0.85	0.40	0.50	0.50	0.68	0.55	0.72
SHAP	Cluster	Concept	0.51	0.75	0.34	0.48	0.40	0.53	0.48	0.65
		Supertok	0.53	0.80	0.38	0.51	0.42	0.57	0.52	0.70
	LinSep	Concept	0.52	0.75	0.35	0.49	0.42	0.55	0.50	0.69
		Supertok	0.58	0.80	0.37	0.55	0.44	0.56	0.52	0.72
RISE	Cluster	Concept	0.53	0.55	0.34	0.36	0.40	0.51	0.50	0.52
		Supertok	0.53	0.56	0.34	0.38	0.42	0.52	0.51	0.55
	LinSep	Concept	0.58	0.60	0.35	0.35	0.43	0.53	0.54	0.55
		Supertok	0.59	0.63	0.38	0.40	0.45	0.55	0.54	0.58
SHAP IQ	Cluster	Concept	0.52	0.55	0.33	0.35	0.40	0.51	0.50	0.52
		Supertok	0.53	0.58	0.35	0.36	0.43	0.53	0.51	0.55
	LinSep	Concept	0.58	0.60	0.34	0.36	0.42	0.52	0.52	0.53
		Supertok	0.58	0.61	0.37	0.38	0.45	0.52	0.53	0.54
IntGrad	Cluster	Concept	0.46	0.77	0.30	0.42	0.43	0.46	0.48	0.69
		Supertok	0.53	0.80	0.33	0.45	0.51	0.47	0.51	0.71
	LinSep	Concept	0.49	0.72	0.28	0.43	0.44	0.56	0.49	0.67
		Supertok	0.55	0.78	0.35	0.48	0.49	0.62	0.52	0.71
GradCAM	Cluster	Concept	0.45	0.50	0.31	0.32	0.41	0.45	0.43	0.45
		Supertok	0.48	0.52	0.31	0.35	0.43	0.46	0.45	0.45
	LinSep	Concept	0.48	0.50	0.37	0.37	0.44	0.45	0.44	0.47
		Supertok	0.48	0.52	0.38	0.37	0.46	0.51	0.47	0.50
FullGrad	Cluster	Concept	0.46	0.47	0.33	0.35	0.38	0.40	0.41	0.40
		Supertok	0.46	0.49	0.32	0.38	0.41	0.41	0.44	0.42
	LinSep	Concept	0.50	0.51	0.39	0.40	0.42	0.43	0.44	0.44
		Supertok	0.52	0.55	0.43	0.39	0.45	0.47	0.45	0.44
CALM	Cluster	Concept	0.48	0.49	0.30	0.30	0.33	0.35	0.42	0.44
		Supertok	0.52	0.53	0.32	0.29	0.35	0.37	0.42	0.45
	LinSep	Concept	0.55	0.57	0.37	0.38	0.35	0.36	0.46	0.48
		Supertok	0.56	0.57	0.42	0.41	0.38	0.41	0.48	0.52
MFABA	Cluster	Concept	0.50	0.51	0.31	0.33	0.42	0.44	0.50	0.50
		Supertok	0.51	0.53	0.37	0.34	0.44	0.44	0.52	0.51
	LinSep	Concept	0.55	0.56	0.33	0.35	0.45	0.44	0.53	0.51
		Supertok	0.55	0.58	0.39	0.39	0.48	0.47	0.55	0.52

Table 3: Average F1 for Supervised (Text Datasets)

Inversion Method	Concept Type	Inversion Objective	Sarcasm			iSarcasm			GoEmotions		
			LLaMA	Qwen	Gemma	LLaMA	Qwen	Gemma	LLaMA	Qwen	Gemma
Random			0.20	0.21	0.20	0.40	0.40	0.40	0.05	0.05	0.05
CosSim	Cluster	Concept	0.39	0.38	0.42	0.70	0.57	0.65	0.18	0.25	0.19
		Supertok	0.25	0.26	0.25	0.65	0.55	0.60	0.16	0.23	0.16
	LinSep	Concept	0.63	0.58	0.57	0.81	0.74	0.83	0.29	0.31	0.25
		Supertok	0.37	0.37	0.40	0.74	0.65	0.71	0.25	0.28	0.23
LIME	Cluster	Concept	0.34	0.33	0.36	0.71	0.63	0.67	0.20	0.27	0.21
		Supertok	0.46	0.45	0.50	0.78	0.67	0.73	0.25	0.31	0.24
	LinSep	Concept	0.52	0.51	0.54	0.79	0.71	0.76	0.29	0.33	0.28
		Supertok	0.70	0.65	0.63	0.87	0.80	0.89	0.34	0.37	0.30
SHAP	Cluster	Concept	0.35	0.34	0.37	0.72	0.64	0.68	0.21	0.28	0.22
		Supertok	0.47	0.46	0.51	0.79	0.68	0.74	0.26	0.32	0.25
	LinSep	Concept	0.53	0.52	0.55	0.80	0.72	0.77	0.30	0.34	0.29
		Supertok	0.71	0.66	0.64	0.88	0.81	0.90	0.35	0.38	0.31
RISE	Cluster	Concept	0.39	0.38	0.42	0.76	0.67	0.72	0.24	0.30	0.25
		Supertok	0.52	0.50	0.55	0.83	0.73	0.79	0.30	0.35	0.28
	LinSep	Concept	0.57	0.56	0.59	0.84	0.76	0.81	0.33	0.37	0.32
		Supertok	0.76	0.71	0.69	0.92	0.85	0.94	0.39	0.42	0.35
SHAP IQ	Cluster	Concept	0.36	0.36	0.39	0.74	0.65	0.70	0.22	0.29	0.23
		Supertok	0.49	0.48	0.53	0.81	0.70	0.76	0.28	0.33	0.26
	LinSep	Concept	0.55	0.54	0.57	0.82	0.74	0.79	0.31	0.35	0.30
		Supertok	0.73	0.68	0.66	0.90	0.83	0.92	0.37	0.40	0.33
IntGrad	Cluster	Concept	0.27	0.27	0.29	0.66	0.56	0.61	0.17	0.24	0.17
		Supertok	0.40	0.39	0.43	0.71	0.58	0.66	0.19	0.26	0.20
	LinSep	Concept	0.39	0.38	0.41	0.75	0.66	0.72	0.26	0.29	0.24
		Supertok	0.64	0.59	0.58	0.82	0.75	0.84	0.30	0.32	0.26
GradCAM	Cluster	Concept	0.31	0.30	0.33	0.69	0.59	0.64	0.19	0.26	0.19
		Supertok	0.44	0.43	0.47	0.75	0.62	0.70	0.23	0.29	0.22
	LinSep	Concept	0.43	0.42	0.45	0.78	0.69	0.74	0.28	0.31	0.27
		Supertok	0.68	0.63	0.62	0.86	0.78	0.87	0.34	0.36	0.29
FullGrad	Cluster	Concept	0.28	0.28	0.30	0.67	0.57	0.62	0.18	0.25	0.18
		Supertok	0.41	0.40	0.44	0.72	0.60	0.67	0.21	0.27	0.21
	LinSep	Concept	0.40	0.39	0.42	0.76	0.67	0.73	0.27	0.30	0.25
		Supertok	0.65	0.60	0.59	0.83	0.76	0.85	0.31	0.33	0.27
CALM	Cluster	Concept	0.34	0.33	0.36	0.71	0.61	0.66	0.21	0.27	0.22
		Supertok	0.47	0.46	0.50	0.78	0.66	0.73	0.26	0.32	0.25
	LinSep	Concept	0.52	0.51	0.54	0.81	0.73	0.78	0.30	0.34	0.29
		Supertok	0.71	0.66	0.65	0.89	0.81	0.91	0.36	0.39	0.32
MFABA	Cluster	Concept	0.33	0.32	0.35	0.70	0.60	0.65	0.20	0.26	0.21
		Supertok	0.46	0.45	0.49	0.77	0.65	0.72	0.25	0.31	0.24
	LinSep	Concept	0.51	0.50	0.53	0.80	0.72	0.77	0.29	0.33	0.28
		Supertok	0.70	0.65	0.64	0.88	0.80	0.90	0.35	0.38	0.31

Table 4: Average F1 for Unsupervised (Image Datasets)

Inversion Method	Concept Type	Inversion Objective	CLEVR		COCO		OpenSurfaces		Pascal	
			CLIP	LLaMA	CLIP	LLaMA	CLIP	LLaMA	CLIP	LLaMA
Random			0.04	0.03	0.10	0.09	0.06	0.05	0.09	0.08
CosSim	Cluster	Concept	0.63	0.46	0.34	0.22	0.19	0.14	0.27	0.22
		Supertok	0.64	0.43	0.37	0.28	0.19	0.15	0.33	0.24
	LinSep	Concept	0.60	0.38	0.33	0.23	0.19	0.15	0.24	0.22
		Supertok	0.59	0.33	0.36	0.26	0.18	0.14	0.30	0.24
LIME	Cluster	Concept	0.52	0.76	0.36	0.45	0.37	0.37	0.33	0.33
		Supertok	0.61	0.81	0.38	0.52	0.41	0.37	0.34	0.32
	LinSep	Concept	0.52	0.68	0.38	0.49	0.39	0.38	0.36	0.33
		Supertok	0.77	0.83	0.41	0.55	0.41	0.39	0.35	0.33
SHAP	Cluster	Concept	0.51	0.75	0.34	0.48	0.40	0.53	0.48	0.65
		Supertok	0.53	0.80	0.38	0.51	0.42	0.57	0.52	0.70
	LinSep	Concept	0.52	0.75	0.35	0.49	0.42	0.55	0.50	0.69
		Supertok	0.58	0.80	0.37	0.53	0.44	0.56	0.52	0.72
RISE	Cluster	Concept	0.53	0.55	0.34	0.36	0.40	0.51	0.50	0.52
		Supertok	0.53	0.56	0.34	0.38	0.42	0.52	0.51	0.55
	LinSep	Concept	0.58	0.60	0.35	0.35	0.43	0.53	0.54	0.55
		Supertok	0.59	0.63	0.38	0.40	0.45	0.55	0.54	0.58
SHAP IQ	Cluster	Concept	0.52	0.55	0.33	0.35	0.40	0.51	0.50	0.52
		Supertok	0.53	0.58	0.35	0.36	0.43	0.53	0.51	0.55
	LinSep	Concept	0.58	0.60	0.34	0.36	0.42	0.52	0.52	0.53
		Supertok	0.58	0.61	0.37	0.38	0.45	0.52	0.53	0.54
IntGrad	Cluster	Concept	0.47	0.56	0.28	0.48	0.33	0.32	0.33	0.34
		Supertok	0.47	0.58	0.31	0.47	0.34	0.35	0.33	0.35
	LinSep	Concept	0.58	0.62	0.31	0.38	0.35	0.34	0.34	0.34
		Supertok	0.59	0.64	0.35	0.39	0.35	0.35	0.34	0.34
GradCAM	Cluster	Concept	0.41	0.50	0.28	0.31	0.36	0.43	0.42	0.43
		Supertok	0.45	0.47	0.31	0.33	0.40	0.42	0.42	0.40
	LinSep	Concept	0.48	0.48	0.35	0.36	0.42	0.44	0.43	0.44
		Supertok	0.46	0.49	0.36	0.34	0.43	0.46	0.45	0.47
FullGrad	Cluster	Concept	0.45	0.42	0.29	0.30	0.36	0.36	0.37	0.38
		Supertok	0.42	0.45	0.31	0.33	0.37	0.38	0.42	0.38
	LinSep	Concept	0.49	0.50	0.35	0.37	0.38	0.41	0.43	0.42
		Supertok	0.49	0.53	0.39	0.34	0.40	0.44	0.42	0.43
CALM	Cluster	Concept	0.44	0.46	0.29	0.26	0.29	0.33	0.37	0.41
		Supertok	0.50	0.48	0.29	0.25	0.32	0.36	0.37	0.43
	LinSep	Concept	0.50	0.53	0.35	0.35	0.32	0.34	0.43	0.45
		Supertok	0.54	0.54	0.39	0.36	0.34	0.39	0.46	0.49
MFABA	Cluster	Concept	0.45	0.47	0.29	0.28	0.40	0.42	0.46	0.48
		Supertok	0.48	0.52	0.33	0.32	0.40	0.41	0.50	0.47
	LinSep	Concept	0.51	0.54	0.30	0.33	0.43	0.42	0.51	0.49
		Supertok	0.50	0.55	0.35	0.36	0.45	0.44	0.49	0.47

Table 5: Average F1 for Unsupervised (Text Datasets)

Inversion Method	Concept Type	Inversion Objective	Sarcasm			iSarcasm			GoEmotions		
			LLaMA	Qwen	Gemma	LLaMA	Qwen	Gemma	LLaMA	Qwen	Gemma
Random			0.20	0.21	0.20	0.40	0.40	0.40	0.05	0.05	0.05
CosSim	Cluster	Concept	0.28	0.26	0.24	0.56	0.59	0.60	0.18	0.23	0.15
		Supertok	0.28	0.25	0.23	0.57	0.59	0.60	0.18	0.26	0.15
	LinSep	Concept	0.28	0.24	0.24	0.60	0.57	0.60	0.18	0.23	0.14
		Supertok	0.28	0.24	0.23	0.60	0.58	0.60	0.19	0.25	0.16
LIME	Cluster	Concept	0.29	0.31	0.33	0.68	0.61	0.72	0.18	0.28	0.23
		Supertok	0.50	0.45	0.51	0.75	0.62	0.69	0.26	0.26	0.25
	LinSep	Concept	0.50	0.53	0.55	0.76	0.76	0.76	0.25	0.34	0.24
		Supertok	0.74	0.60	0.66	0.80	0.83	0.94	0.35	0.38	0.31
SHAP	Cluster	Concept	0.30	0.30	0.35	0.69	0.65	0.65	0.22	0.32	0.19
		Supertok	0.46	0.45	0.46	0.83	0.71	0.78	0.27	0.37	0.27
	LinSep	Concept	0.54	0.54	0.51	0.81	0.69	0.74	0.27	0.33	0.29
		Supertok	0.74	0.68	0.67	0.88	0.79	0.92	0.31	0.40	0.28
RISE	Cluster	Concept	0.40	0.39	0.46	0.80	0.64	0.74	0.21	0.32	0.24
		Supertok	0.49	0.52	0.55	0.80	0.75	0.81	0.27	0.38	0.27
	LinSep	Concept	0.59	0.53	0.60	0.84	0.75	0.84	0.36	0.37	0.32
		Supertok	0.72	0.74	0.70	0.84	0.89	0.85	0.36	0.42	0.34
SHAP IQ	Cluster	Concept	0.38	0.37	0.40	0.74	0.61	0.67	0.20	0.28	0.24
		Supertok	0.46	0.45	0.51	0.85	0.71	0.80	0.27	0.31	0.22
	LinSep	Concept	0.52	0.52	0.59	0.85	0.74	0.80	0.34	0.35	0.29
		Supertok	0.74	0.70	0.66	0.83	0.82	0.82	0.35	0.38	0.35
IntGrad	Cluster	Concept	0.39	0.38	0.41	0.74	0.56	0.65	0.23	0.27	0.18
		Supertok	0.27	0.29	0.27	0.68	0.53	0.63	0.19	0.25	0.19
	LinSep	Concept	0.38	0.41	0.39	0.75	0.66	0.74	0.28	0.27	0.24
		Supertok	0.67	0.58	0.58	0.74	0.77	0.88	0.29	0.32	0.23
GradCAM	Cluster	Concept	0.31	0.33	0.34	0.67	0.56	0.63	0.20	0.25	0.20
		Supertok	0.45	0.44	0.48	0.72	0.61	0.68	0.21	0.31	0.21
	LinSep	Concept	0.44	0.42	0.46	0.70	0.70	0.76	0.27	0.33	0.25
		Supertok	0.70	0.65	0.62	0.74	0.71	0.78	0.34	0.35	0.26
FullGrad	Cluster	Concept	0.28	0.26	0.29	0.66	0.56	0.61	0.18	0.23	0.16
		Supertok	0.39	0.43	0.41	0.73	0.63	0.65	0.19	0.26	0.22
	LinSep	Concept	0.38	0.41	0.42	0.73	0.64	0.70	0.26	0.29	0.27
		Supertok	0.65	0.58	0.60	0.82	0.75	0.87	0.30	0.32	0.25
CALM	Cluster	Concept	0.34	0.34	0.36	0.74	0.61	0.66	0.23	0.28	0.22
		Supertok	0.49	0.46	0.49	0.72	0.64	0.65	0.24	0.30	0.25
	LinSep	Concept	0.51	0.50	0.56	0.80	0.72	0.75	0.29	0.33	0.27
		Supertok	0.72	0.67	0.66	0.82	0.73	0.79	0.35	0.37	0.30
MFABA	Cluster	Concept	0.34	0.35	0.32	0.73	0.62	0.66	0.19	0.27	0.23
		Supertok	0.48	0.43	0.50	0.75	0.66	0.71	0.26	0.34	0.26
	LinSep	Concept	0.54	0.52	0.51	0.81	0.74	0.80	0.28	0.32	0.29
		Supertok	0.71	0.66	0.65	0.85	0.79	0.88	0.36	0.36	0.34

E.3. Qualitative Example Showing Superdetector Tokens for Improved Concept Attribution

Figure 11 further illustrates the advantage: attribution using Superdetector Tokens for the concept *person* provides better coverage for the full target object while avoiding irrelevant regions such as tables, which the global vector incorrectly highlights.

Figure 8: *
(a) Original Image

Figure 9: *
(b) Global Concept Objective

Figure 10: *
(c) Superdetector Tokens Objective

Figure 11: LIME-based concept inversion comparing the Global Concept Objective (b) with the Superdetector Tokens Objective (c) for the concept *person*. Red indicates positive contributions and blue negative contributions. **Superdetector Tokens produce a more complete and precise attribution mask**, while the global objective yields a partial and noisy attribution that misses parts of the target concept and highlights irrelevant background features.

Appendix F. Related Work

Concept-Based Interpretability: Concept-based interpretability links model internals with human-understandable features. Approaches include defining concept vectors as linear separators (e.g., TCAV; (Kim et al., 2018)), or as centroid embeddings from labeled examples (Zou et al., 2023). Unsupervised discovery methods include ACE (Ghorbani et al., 2019), hierarchical clustering (Dalvi et al., 2022), matrix factorization approaches (Zhang and Zhang, 2017; Fel et al., 2022), and sparse autoencoders (Cunningham et al., 2023; Gao et al., 2024). Across these works, concepts are assumed to be recoverable as structured vectors, clusters, or basis elements within representation space.

Ambiguity in Concept Representations: Many open questions remain concerning the structure of concept representations. The linearity hypothesis posits that concepts correspond to directions in activation space, linearly separable and recoverable with simple probes (Mikolov et al., 2013; Elhage et al., 2022). In practice, however, representations are often entangled: polysemanticity allows a single unit or direction to encode multiple features (Goh et al., 2021; Olah et al., 2020; Bricken et al., 2023; O'Mahony et al., 2023), a byproduct of superposition where limited capacity forces overlaps. Concept boundaries are also brittle across layers, exemplar sets, and seeds (Wu et al., 2025; Mahinpei et al., 2021; Nicolson et al., 2025; Mikriukov et al., 2023). Transformer probing studies provide partial

insight: Gandelsman et al. (2023) decompose CLIP embeddings into patch- and head-level contributions, while others trace hierarchical abstraction across layers in language and vision models (Dalvi et al., 2022; Dorszewski et al., 2025; Ghiasi et al., 2022). Our work contributes to this inquiry by showing that extreme activations provide a more structured and faithful view of how transformers encode concepts.

Concept Detection: Concept detection has been approached in several ways. One common strategy is to use global CLS or pooled embeddings, which can be effective (Choi et al., 2021) but tend to dilute local signals and struggle to capture fine-grained concepts. Token-level methods instead evaluate local activations, including approaches that identify concepts via maximally activated tokens (Wu et al., 2025). Our Superdetector Token-based method falls within this class but extends prior work by analyzing the mechanisms through which extreme activations provide reliable concept signals. More recently, zero-shot prompting with vision–language models such as CLIP (Radford et al., 2021) has emerged as another effective concept detection strategy.

Concept Attribution: Traditional attribution methods such as Integrated Gradients (Sundararajan et al., 2017) and Grad-CAM (Selvaraju et al., 2017), along with concept-based adaptations (Kim et al., 2018; Santis et al., 2024; Yu et al., 2024; Fel et al., 2022), have been used to connect predictions to concepts. Beyond these, Gandelsman et al. (2023) decompose CLIP embeddings into patch- and head-level components aligned with text concepts, while other works visualize concept presence more directly via per-patch or token activations, such as cosine similarity heatmaps (Zhou et al., 2024).

Appendix G. Discussion and Future Work

In this work, we introduced and characterized *Superdetector Tokens*, showing that transformers concentrate concept signals in a small subset of highly activated tokens. By focusing on these extreme activations, we demonstrated that it is possible to overcome the limitations of global threshold-based concept recovery approaches, achieving both stronger detection and more accurate, faithful localization. Our results contribute to the broader effort to understand how transformers represent concepts, suggesting that Superdetector Tokens may reflect a general organizing principle of these architectures.

However, we still do not know why these tokens arise, or the processes by which they emerge. Investigating how Superdetector Tokens manifest across layers, evolve during training, or vary with model scale could provide deeper insight into their role. Pursuing these directions may not only advance theoretical understanding but also guide the development of models that represent and use concepts in more interpretable and reliable ways.