Factual Self-Awareness in Language Models: Representation, Robustness, and Scaling

Hovhannes Tamoyan, Subhabrata Dutta, and **Iryna Gurevych**
Ubiquitous Knowledge Processing Lab (UKP Lab)
Department of Computer Science and Hessian Center for AI (hessian.AI)
Technical University of Darmstadt
`www.ukp.tu-darmstadt.de`

Abstract

Factual incorrectness in generated content is one of the primary concerns in ubiquitous deployment of large language models (LLMs). Prior findings suggest LLMs can (sometimes) detect factual incorrectness in their generated content (i.e., fact-checking post-generation). In this work, we provide evidence supporting the presence of LLMs' internal compass that dictate the correctness of factual recall at the time of generation. We demonstrate that for a given subject entity and a relation, LLMs internally encode linear features in the Transformer's residual stream that dictate whether it will be able to recall the correct attribute (that forms a valid entity-relation-attribute triplet). This self-awareness signal is robust to minor formatting variations. We investigate the effects of context perturbation via different example selection strategies. Scaling experiments across model sizes and training dynamics highlight that self-awareness emerges rapidly during training and peaks in intermediate layers. These findings uncover intrinsic self-monitoring capabilities within LLMs, contributing to their interpretability and reliability.[1]

1 Introduction

With the recent advents in the ability of Large Language Models (LLMs), security concerns associated with LLM-based AI agents have increased in direct proportion (Huang et al., 2024). A lion's share of the security concerns about everyday LLM usage comes from their tendency to spit out made up facts— a tendency mainly addressed under the broad description of *hallucination*, bearing an insinuation of *forgetfulness* of the models (Huang et al., 2025). Recent research seeking to address hallucination has posed the question of transparency: is there any generalizable signal that can inform the presence (or absence) of certain knowledge in the internal representation of the model?

Prior research in this direction follows two distinct lines. Kadavath et al. (2022) showed that language models (LMs) can (most of the time) fact-check their output. Multiple subsequent studies (Li et al., 2023a; Azaria and Mitchell, 2023; Burns et al., 2023) have investigated and found that the LM encodes a notion of truth (and false) as linear directions within its representations, and these directions causally elicit the internal fact-checking. This is similar to the broader line of research embodied in the self-reflection paradigm (Madaan et al., 2023; Pan et al., 2023): let the model generate first and ask it to check itself. Recent literature has challenged the paradigm itself in problems such as reasoning and planning (Stechly et al., 2025).

Another line of investigation reveals that language models (LMs) can demarcate between known and unknown entities (Ferrando et al., 2024). This demarcation, defined as the LMs ability (or inability) to recall at least three attributes about the candidate entity correctly, is also linearly represented within

---

[1] https://github.com/UKPLab/arxiv2025-self-awareness

Figure 1: Given an input comprising entity type, entity name, and relation, we obtain the model's token-level prediction probabilities for the attribute. Tokens are labeled `known` if their gold label appears in the top-k predictions, and `forgotten` if in the bottom-l. A sample is labeled `known` if it contains more `known` than `forgotten` tokens, and vice versa. Probabilities are visualized with color-coded bars: green (top-k), red (bottom-l), and gray (others). For example, "Christopher Nolan" falls in the top-k, labeling the sample as `known`, whereas "James Brown" appears in the bottom-l, labeling it as `forgotten`. Final token residuals are linearly probed to detect factual self-awareness.

the residual stream of the Transformer model and instruction-tuned models re-purpose these linear features for refusal of questions related to unknown entities. As opposed to the 'truth direction'-style literature that analyzes the truthfulness of LMs on checking their own outputs, Ferrando et al. (2024) impose the notion of transparency at the time of generation.

However, factual hallucination is not associated only with novel entities. An LM may incorrectly recall a specific attribute of an entity while accurately recalling another (e.g., in the dataset we prepared, 1865 out of 3669 unique entities are neither completely known nor completely forgotten; Gemma-2 2B (Team et al., 2024) can recall at least one attribute about them correctly while erring on at least one attribute). Such a hallucination is intuitively harder to detect, compared to known-unknown entities in line with Ferrando et al. (2024): an unknown entity (potentially unseen in the training data) is associated with a hitherto unseen lexical combination of a few successive tokens, whereas an unknown (or forgotten) factual association needs to be understood in a much deeper connection between entities and relationships. In this work, we show that LMs encode meta-knowledge (i.e., ability or inability to correctly recall) about the fine-grained factual relationships as linear directions (see Figure 1). These directions are activated *before* generating the correct (or incorrect) factual recall, as opposed to the truth directions that are triggered after the generation is complete and the model is asked to check the generation. This internal separability between correct and incorrect generations, which we denote as *known* and *forgotten* factual associations, is surprisingly robust to context perturbation. We further investigate the effects of training and parameter counts on the factual self-awareness of the model; while a minimum model size is required to start encoding the signal, we find that it does so very early in training.

Contribution. Towards investigating factual self-awareness of LMs at generation time, we construct a factual recall dataset. We investigate using Gemma-2 models (2B and 9B) (Team et al., 2024), and the Pythia scaling suite (Biderman et al., 2023), and propose a model-dependent annotation of known-forgotten facts based on logit distribution. We show that LMs construct linear subspaces within internal representations that can demarcate between an upcoming correct/incorrect recall (as opposed to faithfulness in post-hoc checking of correct/incorrect facts). The effects of context and prompt formatting on the formation of these linear subspaces of self-awareness are investigated. Finally, we demonstrate the appearance and improvement of factual self-awareness across two directions of LM scaling: amount of training for next token prediction and model parameters.

2 Background and Related Work

In this section, we review existing literature dissecting language models' (LMs) self-awareness and provide background on linear probes and sparse autoencoders (SAEs), including their prior use in investigating truthfulness and self-awareness in LMs.

Self-awareness of LMs. Prior efforts to make LMs transparent about their mistakes have primarily focused on mitigating hallucinations. A common approach is to ask LMs unanswerable questions and test their ability to refrain (Yin et al., 2023; Bajpai et al., 2024). However, these questions are unanswerable not due to unknown or forgotten factual associations. Betley et al. (2025) similarly studied behavioral self-awareness in LMs. By contrast, our work specifically targets factual self-awareness. Prior work in this area follows the 'self-reflexion' (Madaan et al., 2023; Pan et al., 2023) paradigm: ask the model to reflect on its generation and assess its correctness. Kadavath et al. (2022) supported LM self-awareness by demonstrating the success of such reflection on both multiple-choice and open-ended questions. They also introduced fine-tuning strategies to calibrate model-generated scores of knowledge uncertainty. Kapoor et al. (2024) extended this 'teaching to be self-aware' paradigm via calibration tuning so that model-generated logits better reflect internal uncertainty.

Truthfulness and hallucination in internal representation. Several works (Li et al., 2023a; Azaria and Mitchell, 2023; Burns et al., 2023) have investigated how truthfulness is represented in model internals (i.e., intermediate layer outputs) in post-generation self-checking setups similar to Kadavath et al. (2022). The viability of self-consistency (Zhang et al., 2024; Wang et al., 2023) as a proxy for self-awareness has also been explored. Chen et al. (2024) demonstrate an implicit assumption of self-awareness via self-consistency: among a population of diverse generations, certain spectral patterns of internal states signal hallucination. These methods are primarily designed for reasoning-based tasks, where multiple generations can lead to the same answer, in contrast to immediate factual recall. Ji et al. (2024) analyzed training data to study unseen queries and found that LMs linearly represent seen vs. unseen queries in hidden states. In a related direction, Ferrando et al. (2024) examined how LMs internally demarcate known from unknown entities. They identified linearly encoded features that trigger when the model is queried about an unknown entity—features repurposed in chat-tuned models to elicit refusal. A key limitation, however, lies in defining knowledge at the entity level: when an LM fabricates factual associations, it may invent attributes for a known entity.

Linear and sparse probing. Linear probes are arguably the simplest lens for examining high-dimensional neural representations—given a labeled dataset of an expected behavior, a linear classifier is trained and tested on neural representations to detect whether the behavior is encoded. Sparse autoencoders (SAEs) Bricken et al. (2023); Huben et al. (2023), by contrast, have recently gained popularity for uncovering interpretable decompositions of model latent representations without supervised data. Both approaches align with the linear representation hypothesis Park et al. (2023); Mikolov et al. (2013), which posits that interpretable features—such as sentiment or truthfulness—are embedded as linear directions within the representation space, and that model representations consist of sparse linear combinations of these directions Li et al. (2023b); Zou et al. (2023). While SAEs eliminate the need for supervision, they introduce the challenge of costly training: they must be trained on large volumes of data (and corresponding activations) to avoid data bias (Kissane et al., 2024; Sharkey et al., 2025). SAE dimensionality strongly influences the contextual scale of interpretation (Bussmann et al., 2025). In this work, we first show the empirical similarity between linear probes and SAEs in locating factual self-awareness, then adopt linear probes for scalability.

3 Experimental Setup

Dataset. We construct a factual recall dataset covering four categories: football players, movies, cities, and songs, following the approach of Ferrando et al. (2024). We start with 1,000 entities for each of the following four categories: football player, movie, city, and song, limiting ourselves to a maximum of 10 relationships per entity. For each entity we scrape associated features from Wikidata Vrandečić and Krötzsch (2023). Subsequently, we manually construct templates from the triplets[2] (*entity type*, *entity name*, *relation*) to generate statements and predict the corresponding *attribute*, as illustrated in Figure 1.

Due to the web-scale pretraining data used in training the Transformer-based LMs (as well as the non-availability of training data in case of most open-weight models), it is non-trivial to demarcate which factual associations are known (or unknown) to the LM. We use a proxy definition for the same, using the logit distribution of the LM itself: if a model is able to signal that it can (or cannot)

---

[2]Factual associations are typically represented as triplets of the form (*entity name*, *relation*, *attribute*); we additionally use entity type to avoid ambiguities arising from shared naming across entities, e.g., computer scientist *Michael Jordan* vs sportsman *Michael Jordan*.

recall a certain attribute of an entity correctly (i.e., assign a high logit value to the respective token), we diagnose the behavior as *factually self-aware*.

We feed the samples (entity-relation pairs) into an LM to obtain the probability distribution over the predicted tokens, and classify the factual relations as either `known` or `forgotten`. Since the gold labels are sequences of tokens (e.g. "Christopher", "Nolan") we generate the same number of tokens as in the gold label sequence for each sample and check how many of these appear in the top-k or bottom l-th percentile of the model's output space [3]. A sample is labeled as `known` if more of its gold label tokens appear among the top-k predictions in the logit space. Conversely, if more of the gold label tokens fall below the l-th percentile of the logit distribution, the sample is classified as `forgotten`. This design choice avoids decoding and string matching errors by relying solely on model's output space.

We construct (entity type, entity name, relation) triplets using various templates, but some templates introduced spurious correlations due to their phrasing. Details and examples of all templates are provided in Table 3 and Appendix A. For subsequent experiments, we employ a template devoid of such artifacts, encompassing four relations per entity type and excluding problematic attributes (e.g., city coordinates). The complete list of relations is provided in Appendix A.

We construct the final dataset by setting top-$k = 500$ tokens and selecting the bottom-$l = 0.3$ fraction of the vocabulary space. Detailed experiments on the impact of different (k, l) pairs on factual self-awareness signals are presented in subsection 4.3. This procedure yields a total of 7,380 `known` and 7,268 `forgotten` samples for Gemma 2 2B Team et al. (2024). The distribution of labels across entity types is provided in Table 4 in Appendix B. We partition the dataset $\mathcal{D}$ into training and test subsets with a 0,7/0,3 split for subsequent experiments.

Linear Probe. Let $\mathcal{D} = \{(T_i, y_i)\}_{i=1}^N$ denote the labeled dataset, where each T_i is a token sequence (e.g., a factual recall statement) and $y_i \in \{0, 1\}$ indicates the presence or absence of the feature (e.g., `known`/`forgotten`). We run the model on each $T_i \in \mathcal{D}$ and extract the residual stream of the final token of the prompt template x_{T_i}, following Meng et al. (2022); Geva et al. (2023); Nanda et al. (2023).

Definition 1. *For a residual stream representation $x_{l,T_i} \in \mathbb{R}^d$ of sample T_i at layer l, a probe is a learnable function $f : \mathbb{R}^d \to \mathbb{R}$ trained to predict y_i from x_{l,T_i}.*

The linear probe serves as a simple diagnostic classifier that maps the residual stream output to a scalar output via a learned weight vector $\mathbf{w} \in \mathbb{R}^d$ and a bias term $b \in \mathbb{R}$. At layer l, we introduce a linear probe and its corresponding optimization objective as:

$$f_l : \mathbb{R}^d \to \mathbb{R}, \quad f_l(x_{l,T_i}) = \mathbf{w}^\top x_{l,T_i} + b, \qquad \min_{\mathbf{w},b} \sum_{(T_i, y_i) \in \mathcal{D}} \text{BCE}\big(y_i, \sigma\big(f_l(x_{l,T_i})\big)\big). \quad (1)$$

where σ is the sigmoid function and BCE the binary cross-entropy loss. The parameters $\mathbf{w}$ and b are learned by minimizing the binary cross-entropy loss over the dataset $\mathcal{D}$.

To break symmetry and introduce controlled variation across layers, we initialize scalar biases as $b = 0.1 \times (-1)^l$, where the layer index l deterministically seeds randomness via $\text{seed} + 100 \times l$ with $seed$[4]. This ensures consistent yet diverse initialization across layers. To further encourage diversity in learned solutions, each probe is assigned a slightly different learning rate, scaled for layer l as $\text{lr}_l = \texttt{base\_lr} \times \big(1.1 - 0.2 \cdot \frac{l}{L}\big)$, where `base_lr` is the initial rate and L the total number of layers. This encourages probes at different depths to converge to distinct solutions. All probes are optimized using Adam, with initial learning rate $1\mathrm{e}{-4}$ and weight decay $1\mathrm{e}{-5}$.

Separation Scores. Sparse Autoencoders (SAEs) conform to the definition of a probe as stated in Definition 1. We collectively refer to the SAE-encoded representations and the outputs of the linear probe as *activations*. Following Ferrando et al. (2024), we compute separation scores. For each latent dimension j of the activation vector (with $j = 1$ for a linear probe), we calculate the proportion of

---

[3]Note that in this definition, k is an integer denoting the count of tokens, while l is a fraction denoting a subset of the vocabulary. This demarcation arises from the actual count of tokens in these bands: high probability tokens are exponentially fewer than low probability ones.

[4]All experiments are conducted with three random seeds (73, 5, 120); we observe negligible variance and omit the results for brevity.

instances with positive activations (i.e., greater than zero) separately for the `known` and `forgotten` sets: $g_{l,j}^{\text{known}} = \frac{\sum_i^{N^{\text{known}}} \mathbb{1}\left[a_{l,j}(x_{l,T_i}^{\text{known}})>0\right]}{N^{\text{known}}}$ and $g_{l,j}^{\text{forgotten}} = \frac{\sum_i^{N^{\text{forgotten}}} \mathbb{1}\left[a_{l,j}(x_{l,T_i}^{\text{forgotten}})>0\right]}{N^{\text{forgotten}}}$, where N^{known} and $N^{\text{forgotten}}$ denote the total number of prompts, and x_{l,T_i}^{known} and $x_{l,T_i}^{\text{forgotten}}$ represent the latent activations for the `known` and `forgotten` samples, respectively, in each subset of $\mathcal{D}$. Latent separation scores (or vectors) are computed as the difference between these proportions: $s_{l,j}^{\text{known}} = g_{l,j}^{\text{known}} - g_{l,j}^{\text{forgotten}}$ and $s_{l,j}^{\text{forgotten}} = g_{l,j}^{\text{forgotten}} - g_{l,j}^{\text{known}}$, where s_l^{known} is used to detect `known` entities, and $s_l^{\text{forgotten}}$ is used to detect `forgotten` entities.

Computational Resource Requirements. All experiments are conducted on NVIDIA A100-SXM4-80GB GPUs. Models with less than 7B parameters are run on a single GPU, while larger models are executed on two GPUs to accommodate memory and computational requirements.

4 Generation-time factual self-awareness in LMs

4.1 Linear Probes vs. Sparse Autoencoders (SAEs)

We utilize the Gemma 2 2B and 9B models from the Gemma Scope framework (Lieberum et al., 2024), which provides a suite of SAEs pretrained on the activations of each layer of the Gemma 2 models (Team et al., 2024). We train linear probes on the residual stream hooks of these models and generate the corresponding separation plots on test sets for both the 2B and 9B variants using both SAE and linear probing methods. We select the top-five (one for linear probe) latent dimensions with the highest separation scores from the `known` and `forgotten` vectors for each entity type t and layer l. To assess generality and robustness, we compute $\text{MaxMin}^{\text{known},l} = \max_j \min_t s_{l,j}^{\text{known},t}$ and analogously define $\text{MaxMin}^{\text{forgotten},l}$, where j indexes latent dimensions.
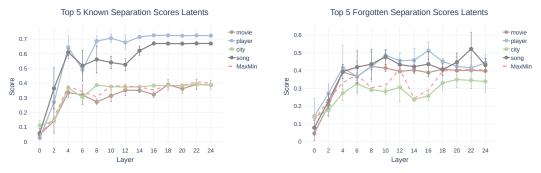

Figure 2: Top-five latent separation scores across transformer layers using SAE activations from Gemma 2 2B. Left: For **known** entities exhibit clear layer-wise separation, peaking around layers 6–14. Right: For **forgotten** entities, separation scores are lower and more variable, indicating reduced disentanglement. Categories include `movie`, `player`, `city`, and `song`; MaxMin denotes the difference between max and min class means.

We illustrate the evolution of separation scores across layers for SAE activations in Figure 2 (Gemma 2 9B model results are provided in Appendix C). As shown by the red curve, MaxMin_l increases in the intermediate layers. This pattern suggests that the most *generalized* latents—those consistently separating `known` from `forgotten` entities across all types are primarily located in the middle and final layers. Linear probe separation scores follow a similar trend, as shown in Figure 3; however, since these activations are scalar, the `known` and `forgotten` scores appear as mirror images, having equal magnitudes but opposite signs.

We train linear probes on the Gemma 2 (2B, 9B) (Team et al., 2024) and Pythia (70M, 1.4B, 6.9B, 12B) (Biderman et al., 2023) models, evaluating performance using standard binary classification metrics and reporting the accuracy improvement over a random baseline, denoted as Δ. Metrics from the final training epoch are reported for both the training and test subsets, as shown in Table 1. Among all models, Gemma 2 2B exhibits the strongest performance, achieving the highest test set separation score with $\Delta = 0.311$. Within the Pythia family, the 12B model performs best ($\Delta = 0.120$); however, a substantial performance gap remains relative to Gemma 2 2B.
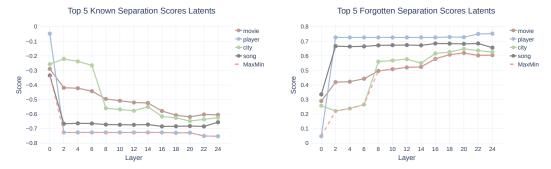
Figure 3: Latent separation scores across layers using Linear Probe activations from Gemma 2 2B. Left: **Known** entities show separation scores that are identical in magnitude but negated in sign compared to **forgotten** entities (right), indicating that the same latents are used for both but with reversed class-directional structure. Categories include `movie`, `player`, `city`, and `song`; MaxMin denotes the difference between maximum and minimum class means.

Table 1: Linear probing results on Gemma 2 and Pythia model families. Metrics are reported on training and test subsets from the final epoch (3). Accuracy gains over random baselines are indicated by Δ, with values in parentheses denoting standard deviations. Gemma 2 2B achieves the highest test set performance, while Pythia 12B is strongest within its family, though with a notable gap.

Model	Subset	Loss	AUC ROC	Accuracies		
				Observed	**Random Baseline**	**Δ (Observed - Baseline)**
Gemma 2 2B	Train	**0.383** (0.063)	**0.901** (0.061)	**0.833** (0.052)	0.501	**0.332**
	Test	0.397 (0.064)	0.896 (0.060)	0.820 (0.056)	0.509	**0.311**
Gemma 2 9B	Train	0.393 (0.052)	0.899 (0.050)	0.826 (0.040)	0.555	0.271
	Test	**0.387** (0.050)	**0.903** (0.047)	**0.829** (0.032)	0.564	0.265
Pythia 70M	Train	0.473 (0.008)	0.546 (0.029)	0.818 (0.001)	0.818	0.000
	Test	0.465 (0.005)	0.551 (0.022)	0.822 (0.001)	0.822	0.000
Pythia 1.4B	Train	**0.358** (0.032)	0.843 (0.057)	**0.837** (0.011)	0.807	0.030
	Test	**0.365** (0.031)	0.842 (0.056)	**0.831** (0.013)	0.803	0.028
Pythia 6.9B	Train	0.393 (0.028)	**0.852** (0.048)	0.829 (0.014)	0.747	0.082
	Test	0.395 (0.026)	**0.857** (0.047)	0.827 (0.011)	0.746	0.081
Pythia 12B	Train	0.442 (0.030)	0.842 (0.046)	0.798 (0.017)	0.687	**0.111**
	Test	0.464 (0.027)	0.846 (0.043)	0.794 (0.022)	0.674	**0.120**

Beyond the overall advantage of the Gemma 2 2B model, several notable trends emerge from the probing results. Within each model family, increasing parameter count does not consistently improve linear probe performance. For instance, while Gemma 2 9B achieves a slightly higher test AUC-ROC than Gemma 2 2B, its accuracy gain over the random baseline (Δ) is lower. This suggests that larger models do not necessarily produce more linearly decodable representations of self-awareness. A similar pattern holds for the Pythia models: although the 12B variant achieves the highest test-time Δ, smaller versions like Pythia 6.9B and 1.4B show comparable accuracies, albeit with smaller gains over their baselines. Pythia 70M represents a degenerate case where accuracy matches the random baseline ($\Delta = 0$), indicating that the smallest model fails to encode self-awareness features.

We further examine the distribution of self-awareness signals across model layers by analyzing layer-wise linear probe accuracy, as shown in Figure 4. For both Gemma 2 2B and Pythia 12B, accuracy rises sharply in the initial layers before stabilizing. In Gemma 2 2B, performance plateaus around the fifth layer, reaching a peak test accuracy of approximately 0.82; well above the random baseline. Accuracy remains consistently high in subsequent layers, with a slight decline in the final three layers, suggesting that self-awareness directions are preserved throughout the network depth.

In contrast, Pythia 12B shows a slower but steady accuracy increase across layers. Its final test accuracy remains below that of Gemma 2 2B, consistent with earlier results. Notably, the random baseline for Pythia 12B is substantially higher than for Gemma 2 2B (approximately 0.69 vs. 0.50). These patterns suggest that Gemma 2 2B achieves linearly accessible self-awareness representations earlier and maintains them more robustly, while Pythia 12B requires deeper processing to approach similar performance.
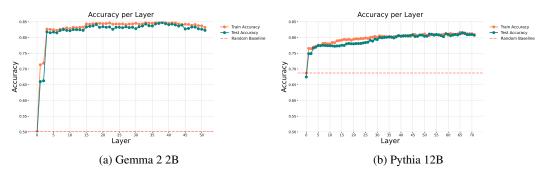
| (a) Gemma 2 2B | (b) Pythia 12B |

Figure 4: Layer-wise linear probe accuracy for Gemma 2 2B and Pythia 12B. Orange/blue: train/test; red dashed: random baseline.

4.2 Robustness Against Context Perturbation

To what extent is a model's factual self-awareness robust to changes in input context? To address this question, we train linear probes on each model layer and evaluate their performance on contextually modified input samples. We design four targeted experiments to systematically assess the test-time robustness of self-awareness directions under such perturbations.

Quotation Marks. Enclose the entity name within single or double quotation marks:

The director of the movie 'Inception' is Christopher Nolan.

Prompt formatting, including punctuation, spacing, and quoting, has been shown to significantly impact model performance (Gonen et al., 2023; Sclar et al., 2024).

Statement Question. Prepend a natural language question crafted from the sample quadruples:

Who is the director of the movie Inception? The director of the movie Inception is Christopher Nolan.

Rephrasing inputs as questions can improve model reasoning and accuracy, with even minor phrasing changes affecting outputs and models performance (Kojima et al., 2022; Mizrahi et al., 2024).

Few-Shot. Prepend few-shot context by adding a small set of sample (e.g., three) from the dataset to the input. These samples are chosen according to one of two entity modes.

Only: all selected samples have the same entity name; however, the relation and attribute may vary:

The release year of the movie Inception is 2010. The director of the movie Inception is Christopher Nolan.

Unique: each entity name appears only once in the context; however, the relations between entities are not necessarily distinct.

The genre of the movie The Matrix is science fiction. The director of the movie Inception is Christopher Nolan.

Few-shot prompting improves generalization but remains brittle to surface-level changes, highlighting the need to assess robustness (Sclar et al., 2024).

Random Statement. Prepend a fixed, unrelated grammatically correct sentence to the input prompt: "The cat darted under the couch as the thunder cracked outside."

The cat darted under the couch as the thunder cracked outside. The director of the movie Inception is Christopher Nolan.

Semantically neutral distractors, such as irrelevant prefixes, significantly affect model predictions, indicating sensitivity to prompt framing beyond content (Sclar et al., 2024).

We assess the robustness of self-awareness directions to contextual perturbations at test time using the Gemma 2 2B model, shown in Table 2. The model is trained on a fixed dataset, and its performance is evaluated across various input modifications.

Adding quotation marks—either single or double—yields only minor reductions in test accuracy (0.802 and 0.797) compared to the unmodified baseline (0.820), indicating robustness to superficial punctuation changes. Rephrasing factual prompts as questions (*Statement Question*) leads to a larger performance drop (0.756), suggesting sensitivity to structural semantics beyond lexical form. In contrast, appending unrelated content (*Random Sentence*) minimally affects performance (0.791), indicating resilience to distractors when the core entity–relation structure is preserved. The *Few-Shot*

7

Table 2: Self-awareness directions robustness against various context perturbations for Gemma 2 2B model. Δ indicates test accuracy gain over the random baseline.). Standard deviation in parentheses.

Modification Type	Train (shared across all)			Test (varies by modification)		
	Loss	AUC ROC	Accuracy	Loss	AUC ROC	Accuracy
None				**0.397** (0.064)	**0.896** (0.060)	**0.820** (0.056)
Quotation Marks (single)				**0.431** (0.066)	**0.882** (0.057)	**0.802** (0.060)
Quotation Marks (double)				0.448 (0.066)	0.877 (0.056)	0.797 (0.072)
Statement Question	0.383 (0.063)	0.901 (0.061)	0.833 (0.052)	0.520 (0.092)	0.856 (0.056)	0.756 (0.061)
Few-Shot (Only)				0.708 (0.183)	0.771 (0.069)	0.650 (0.083)
Few-Shot (Unique)				0.538 (0.157)	0.845 (0.074)	0.753 (0.065)
Random Sentence				0.458 (0.070)	0.871 (0.057)	0.791 (0.061)

(Only) setting causes substantial degradation (0.650), likely due to signal dilution, whereas *Few-Shot (Unique)* better maintains accuracy (0.753), underscoring the role of relational diversity in preserving linear decodability.

Overall, these findings underscore that factual self-awareness in LMs is relatively robust to superficial noise but sensitive to semantically meaningful shifts in input structure.

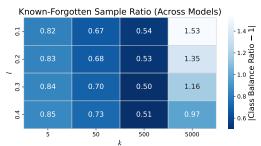### 4.3 Impact of Sampling Parameters k-l on Probe Behavior

Figure 5: Known-Forgotten sample ratio for each (k, l) configuration, aggregated across all models. Lower values (darker) indicate more balanced retention, helping identify the globally optimal (k, l) setting that generalizes across models.
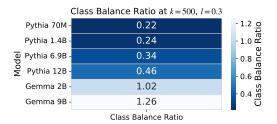

Figure 6: Class balance ratio at $k = 500, l = 0.3$ for each model. Values closer to 1.0 indicate more balanced retention, though Gemma 2B diverges significantly, suggesting this configuration may not generalize well to all architectures.
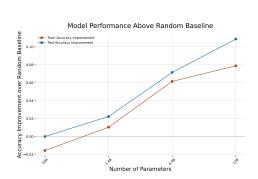
We systematically evaluate the effect of varying k and l values on linear probe performance and the class balance between `known` and `forgotten` samples for Gemma 2 (2B, 9B) (Team et al., 2024) and Pythia (70M, 1.4B, 6.9B, 12B) (Biderman et al., 2023) models. Known-to-forgotten class ratios for different k–l configurations per model m are listed in Appendix D, with visualizations in Figures 13 and 12 for Pythia and Gemma 2, respectively. Among the Pythia models, Pythia-12B exhibits the highest known-to-forgotten ratio under the default k–l setting, with $\text{Ratio}_{\text{Pythia-12B}}(k = 500, l = 0.3) = 0.46$. In the Gemma 2 series, Gemma-2B shows a substantially more balanced attribution of factual knowledge, with $\text{Ratio}_{\text{Gemma-2B}}(k = 500, l = 0.3) = 1.02$.

To complement the analysis of class balance, we examine how the (k, l) configuration affects downstream recovery of factual self-awareness in terms of linear probe performance. In Figure 11 in Appendix D, we report the test and train accuracy improvements over a random baseline for the two representative models—Gemma 2B and Pythia 12B—across all evaluated (k, l) settings. These heatmaps reveal how performance varies with sampling parameters, where each cell shows the absolute accuracy gain (test/train) above random guessing.

Notably, linear probes on the Gemma 2B model exhibit consistent and substantial gains across multiple configurations, particularly around $k = 500$, whereas Pythia 12B shows smaller but more stable improvements. These results highlight the trade-off between class balance and discriminative performance, further motivating the choice of $(k = 500, l = 0.3)$ as a configuration that yields competitive accuracy.

5 Scaling Behavior

To further investigate the emergence of self-awareness directions in language models, we analyze models from the Pythia scaling suite Biderman et al. (2023), focusing on variation across model sizes—specifically 70M, 1.4B, 6.9B, and 12B parameters.

Figure 7: Accuracy gain over random baseline for training (orange) and test (blue) linear probes as a function of model size. Larger models exhibit greater gains in accuracy, with test performance benefiting more substantially from scaling.

Figure 8: Linear probe accuracy across Pythia 1.4B training checkpoints/tokens. (Top) Training accuracy by layer (warmer colors = deeper layers). (Bottom) Test accuracy by layer (cooler colors = deeper layers, brighter = later layers). Red dashed line: random baseline.

We measure the emergence of linearly decodable features associated with self-awareness by computing the accuracy improvement of linear probes over a random baseline across models of increasing scale. As shown in Figure 7, both training and test accuracy improvements grow monotonically with model size, indicating that larger models develop more robust and generalizable representations. Notably, the improvement is more pronounced on the test set, suggesting that increased capacity enhances the transferability of these features beyond the training distribution.

To further examine how these features evolve during training, we evaluate checkpoints of Pythia 1.4B, spaced every 5000 steps from 0 to 143000, as shown in Figure 8. At initialization (0), linear probe accuracy is at the random baseline, indicating no self-awareness directions in the untrained model. During training, middle layers consistently yield the highest probe accuracy, while early and late layers perform worse. Training accuracy rises quickly and plateaus early, while test accuracy improves more gradually. On the test set, the highest accuracy occurs in the upper layers, suggesting self-awareness features are more strongly encoded later in the network. In contrast, middle layers dominate on the training set, indicating a divergence in the distribution of generalizable vs. task-specific features across depth.

6 Conclusion

In this work, we explore the landscape of factual self-awareness of pretrained LMs. We precisely ask the question whether an LM encodes its certainty within its neural representations that it will be able to recall a given factual association. We frame this as awareness-at-generation. Providing an affirmative answer, we show that encoding of such a signal is linear and surprisingly robust against context perturbation. We find that while a threshold model size is essential for the self-awareness signal to appear, the strength of the signal is not directly proportional to scaling. Same trend is observed in terms of training scale: the signal appears quite early in training and saturates quickly. We argue that this specific type of self-awareness, which is evident at the time of generation, can serve as a stronger entry point to curb LM hallucination compared to post-hoc truthfulness, investigated hitherto. Compared to the latter, this awareness-at-generation can be repurposed to restrain the model from a generation attempt before the actual generation.

7 Acknowledgments

Limitations

By definition, the investigated self-awareness signal is limited to factual recall alone. There are multiple other forms of self-awareness that this work does not address. We look into the rudimentary form of factual recall where all the necessary information (e.g., entity name, entity type, relation) is provided within the immediate query. In open-ended generation tasks, the LM might need to gather this information from scattered context, resolve coreferences, perform multi-hop factual recall implicitly, etc. We leave the investigation of self-awareness under such stressors as a future work. Additionally, the dataset used in this study is restricted in its coverage of entity types and relation categories. Expanding the dataset to include a broader and more diverse range of entities and relational structures would provide a more comprehensive understanding of how self-awareness representations generalize across semantic domains. Finally, while we demonstrate the linearly separable signals of factual self-awareness in the intermediate neural representations and its scaling behavior, it is not known how the model learns to encode this from mere next token prediction training, or the internal causal components that construct and use these signals.

References

Xiaowei Huang, Wenjie Ruan, Wei Huang, Gaojie Jin, Yi Dong, Changshun Wu, Saddek Bensalem, Ronghui Mu, Yi Qi, Xingyu Zhao, Kaiwen Cai, Yanghao Zhang, Sihao Wu, Peipei Xu, Dengyu Wu, Andre Freitas, and Mustafa A. Mustafa. A survey of safety and trustworthiness of large language models through the lens of verification and validation. *Artificial Intelligence Review*, 57 (7):175, Jun 2024. ISSN 1573-7462. doi: 10.1007/s10462-024-10824-0. URL https://doi.org/10.1007/s10462-024-10824-0.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Trans. Inf. Syst.*, 43 (2), January 2025. ISSN 1046-8188. doi: 10.1145/3703155. URL https://doi.org/10.1145/3703155.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, Jackson Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olsson, Sam Ringer, Dario Amodei, Tom Brown, Jack Clark, Nicholas Joseph, Ben Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. Language models (mostly) know what they know. *CoRR*, abs/2207.05221, 2022. doi: 10.48550/ARXIV.2207.05221. URL https://doi.org/10.48550/arXiv.2207.05221.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 41451–41530. Curran Associates, Inc., 2023a. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/81b8390039b7302c909cb769f8b6cd93-Paper-Conference.pdf.

Amos Azaria and Tom M. Mitchell. The internal state of an LLM knows when it's lying. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, pages 967–976. Association for

Computational Linguistics, 2023. doi: 10.18653/V1/2023.FINDINGS-EMNLP.68. URL `https://doi.org/10.18653/v1/2023.findings-emnlp.68`.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. Discovering latent knowledge in language models without supervision. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL `https://openreview.net/forum?id=ETKGuby0hcs`.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-refine: Iterative refinement with self-feedback. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL `http://papers.nips.cc/paper_files/paper/2023/hash/91edff07232fb1b55a505a9e9f6c0ff3-Abstract-Conference.html`.

Liangming Pan, Michael Saxon, Wenda Xu, Deepak Nathani, Xinyi Wang, and William Yang Wang. Automatically correcting large language models: Surveying the landscape of diverse self-correction strategies. *CoRR*, abs/2308.03188, 2023. doi: 10.48550/ARXIV.2308.03188. URL `https://doi.org/10.48550/arXiv.2308.03188`.

Kaya Stechly, Karthik Valmeekam, and Subbarao Kambhampati. On the self-verification limitations of large language models on reasoning and planning tasks. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL `https://openreview.net/forum?id=4O0v4s3IzY`.

Javier Ferrando, Oscar Obeso, Senthooran Rajamanoharan, and Neel Nanda. Do i know this entity? knowledge awareness and hallucinations in language models. *arXiv preprint arXiv:2411.14257*, 2024.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. Gemma 2: Improving open language models at a practical size, 2024. *URL https://arxiv. org/abs/2408.00118*, 1(3), 2024.

Stella Biderman, Sid Black, Eric Hallahan, et al. Pythia: A suite for analyzing large language models across training and scaling. *arXiv preprint arXiv:2304.01373*, 2023.

Zhangyue Yin, Qiushi Sun, Qipeng Guo, Jiawen Wu, Xipeng Qiu, and Xuanjing Huang. Do large language models know what they don't know? In Anna Rogers, Jordan L. Boyd-Graber, and Naoaki Okazaki, editors, *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 8653–8665. Association for Computational Linguistics, 2023. doi: 10.18653/V1/2023.FINDINGS-ACL.551. URL `https://doi.org/10.18653/v1/2023.findings-acl.551`.

Prasoon Bajpai, Niladri Chatterjee, Subhabrata Dutta, and Tanmoy Chakraborty. Can llms replace neil degrasse tyson? evaluating the reliability of llms as science communicators. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, pages 15895–15912. Association for Computational Linguistics, 2024. URL `https://aclanthology.org/2024.emnlp-main.889`.

Jan Betley, Xuchan Bao, Martín Soto, Anna Sztyber-Betley, James Chua, and Owain Evans. Tell me about yourself: Llms are aware of their learned behaviors. *arXiv preprint arXiv:2501.11120*, 2025.

Sanyam Kapoor, Nate Gruver, Manley Roberts, Katie Collins, Arka Pal, Umang Bhatt, Adrian Weller, Samuel Dooley, Micah Goldblum, and Andrew Gordon Wilson. Large language models must be taught to know what they don't know. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL `http://papers.nips.cc/paper_files/paper/2024/hash/9c20f16b05f5e5e70fa07e2a4364b80e-Abstract-Conference.html`.

Wenqi Zhang, Yongliang Shen, Linjuan Wu, Qiuying Peng, Jun Wang, Yueting Zhuang, and Weiming Lu. Self-contrast: Better reflection through inconsistent solving perspectives. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 3602–3622. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.ACL-LONG.197. URL https://doi.org/10.18653/v1/2024.acl-long.197.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/forum?id=1PL1NIMMrw.

Chao Chen, Kai Liu, Ze Chen, Yi Gu, Yue Wu, Mingyuan Tao, Zhihang Fu, and Jieping Ye. INSIDE: llms' internal states retain the power of hallucination detection. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=Zj12nzlQbz.

Ziwei Ji, Delong Chen, Etsuko Ishii, Samuel Cahyawijaya, Yejin Bang, Bryan Wilie, and Pascale Fung. LLM internal states reveal hallucination risk faced with a query. *CoRR*, abs/2407.03282, 2024. doi: 10.48550/ARXIV.2407.03282. URL https://doi.org/10.48550/arXiv.2407.03282.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, et al. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2, 2023.

Robert Huben, Hoagy Cunningham, Logan Riggs Smith, Aidan Ewart, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. In *The Twelfth International Conference on Learning Representations*, 2023.

Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models. *arXiv preprint arXiv:2311.03658*, 2023.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. *Advances in Neural Information Processing Systems*, 26, 2013.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. *Advances in Neural Information Processing Systems*, 36:41451–41530, 2023b.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, et al. Representation engineering: A top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*, 2023.

Connor Kissane, Robert Krzyzanowski, Neel Nanda, and Arthur Conmy. Saes are highly dataset dependent: A case study on the refusal direction. In *Alignment Forum*, 2024.

Lee Sharkey, Bilal Chughtai, Joshua Batson, Jack Lindsey, Jeff Wu, Lucius Bushnaq, Nicholas Goldowsky-Dill, Stefan Heimersheim, Alejandro Ortega, Joseph Isaac Bloom, Stella Biderman, Adrià Garriga-Alonso, Arthur Conmy, Neel Nanda, Jessica Rumbelow, Martin Wattenberg, Nandi Schoots, Joseph Miller, Eric J. Michaud, Stephen Casper, Max Tegmark, William Saunders, David Bau, Eric Todd, Atticus Geiger, Mor Geva, Jesse Hoogland, Daniel Murfet, and Tom McGrath. Open problems in mechanistic interpretability. *CoRR*, abs/2501.16496, 2025. doi: 10.48550/ARXIV.2501.16496. URL https://doi.org/10.48550/arXiv.2501.16496.

Bart Bussmann, Noa Nabeshima, Adam Karvonen, and Neel Nanda. Learning multi-level features with matryoshka sparse autoencoders. *CoRR*, abs/2503.17547, 2025. doi: 10.48550/ARXIV.2503.17547. URL https://doi.org/10.48550/arXiv.2503.17547.

Denny Vrandečić and Markus Krötzsch. Wikidata. https://www.wikidata.org, 2023.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in gpt. *Advances in neural information processing systems*, 35:17359–17372, 2022.

Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. Dissecting recall of factual associations in auto-regressive language models. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12216–12235, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.751. URL https://aclanthology.org/2023.emnlp-main.751/.

Neel Nanda, Senthooran Rajamanoharan, János Kramár, and Rohin Shah. Fact finding: Attempting to reverse-engineer factual recall on the neuron level. https://www.alignmentforum.org/posts/iGuwZTHWb6DFY3sKB/fact-finding-attempting-to-reverse-engineer-factual-recall, 2023. AI Alignment Forum.

Tom Lieberum, Senthooran Rajamanoharan, Arthur Conmy, Lewis Smith, Nicolas Sonnerat, Vikrant Varma, János Kramár, Anca Dragan, Rohin Shah, and Neel Nanda. Gemma scope: Open sparse autoencoders everywhere all at once on gemma 2. *arXiv preprint arXiv:2408.05147*, 2024.

Hila Gonen, Yonatan Belinkov, Ido Dagan, and Yoav Goldberg. Demystifying prompts in language models via perplexity estimation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2023.

Noam Sclar, Ehud Guriel, and Omer Levy. Quantifying lms' sensitivity to spurious prompt formatting. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2024.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35: 22199–22213, 2022.

Itay Mizrahi, Nimrod Sznajder, Libby Barak, and Yoav Goldberg. State of what art? a call for multi-prompt llm evaluation. *Transactions of the Association for Computational Linguistics (TACL)*, 2024.

A   Input Templates

The selected relations include: `player` — 'birth place', 'birth date', 'position', 'nationality'; `movie` — 'director', 'release date', 'genre', 'country'; `city` — 'country', 'first mayor', 'founded date', 'climate type'; and `song` — 'artist', 'album', 'release date', 'language'.

Initially, we constructed the input templates using the relations described in Ferrando et al. (2024) for four categories: `football player`, `movie`, `city`, and `song`, we call this set of relations as `relations1`. The specific relationships extracted for each category are as follows:

- `player`: birthplace, birth date, teams played.
- `movie`: director, screenwriter, release date, genre, duration, cast.
- `city`: country, population, elevation, coordinates.
- `song`: artist, album involvement, publication year, genre.

Since the number of relations is not balanced for the categories and some relation answers have non-trivial modality, e.g. coordinates of a city, we propose a new unified dataset, that is balanced in the sense of number of features and standard output modalities, we call this set of relations `relations2`. The following relations shape the `unified` dataset:

- `player`: birthplace, birthdate, position, nationality.
- `movie`: director, release date, genre, production country.
- `city`: country, population, founded date, timezone.
- `song`: artist, album label, release date, language.

Afterwards, we hand-craft templates from the quadruples to form statements. For example, "The movie Inception was directed by *director* Christopher Nolan". Since the expected answer for some relations can be ambiguous as in "The player Michael Jordan was born in *city of...*" where it is unclear whether the response should be a location or a year—we incorporate "hints" at the end of each relation.

We experimented with four input templates (see Table 3) using the `relations2` set, aiming to eliminate spurious correlations and isolate the self-awareness signal captured by linear probes. Notably, only `template2` consistently captures the self-awareness signal without interference from confounding factors.

Template Name	Sample Sentence
template1	"The player Youri Djorkaeff was born in the city of"
template1_const_end	"The player Youri Djorkaeff's birth city is"
template2	"The city of birth for the player Youri Djorkaeff is"
template2_balanced	"The city of birth for the player Youri Djorkaeff is"

Table 3: Input templates using the entity type "player" and entit name "Youri Djorkaeff"

`template1`: Places the entity type and name at the beginning of the sentence and ends with variable tokens. For example: *"The <entity_type> <entity_name> was born in the city of..."*

`template1_const_end`: Similar to `template1`, it places the entity type and name at the beginning of the sentence but always ends with the fixed token *"is"*. For example: *"The <entity_type> <entity_name>'s birth city is..."*

`template2`: In contrast to `template1`, it places the entity type and name at the end of the prompt. Like `template1_const_end`, it also ends with the token *"is"*. For example: *"The city of birth for the <entity_type> <entity_name> is..."*

`template2_balanced`: Further improvement of `template2` to have balanced number of known and forgotten samples across relations per category.

See the full set of templates from `template2_balanced` that were used in the experiments Table 3.

B Sample Distribution Across Entity Types

We present the full distribution of known and forgotten samples using `template2_balanced` for Gemma 2 2B with $k = 500$ and $l = 0.3$ parameters in Table 4 and for Pythia 12B in Table 5.

Table 4: Distribution of known and forgotten samples across entity categories for the Gemma 2 2B model, using top-$k = 500$ and bottom-$l = 0.3$ thresholds.

Category	Known	Forgotten	Subset Total
Player	1286	2922	4208
Movie	3602	1810	5412
City	1017	541	1558
Song	1475	1995	3470
Total	7380	7268	14648

Table 5: Distribution of known and forgotten samples across entity categories for the Pythia 12B model, using top-$k = 500$ and bottom-$l = 0.3$ thresholds.

Category	Known	Forgotten	Subset Total
Player	657	3551	4208
Movie	3602	1810	5412
City	574	984	1558
Song	538	2932	3470
Total	5371	9277	14648

C Separation Scores

The latent separation scores for Gemma 2 9B using SAE activations (Figure 9) and Linear Probe activations (Figure 10).
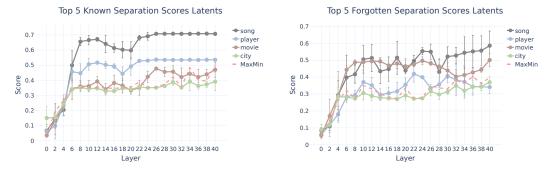
Figure 9: Latent separation scores using SAE activations on Gemma 2 9B. Left: **known** entities. Right: **forgotten** entities.
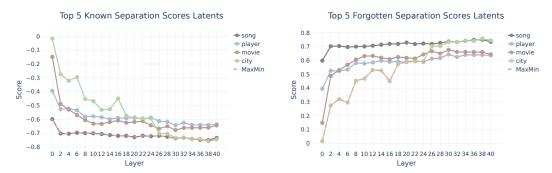

Figure 10: Latent separation scores using Linear Probe activations on Gemma 2 9B. Left: **known** entities. Right: **forgotten** entities.

D (k,l) Pair Impact on Probe Behavior

The figures in this section illustrate how varying the (k, l) parameters—representing the number of known and forgotten samples, respectively—affects linear probe performance and class balance outcomes across different model scales. Figure 11 presents the test and train accuracy gains over a random baseline for the Gemma 2 2B and Pythia 12B models. Notably, we observe that increases in k (the number of known samples) generally correspond to higher accuracy gains, particularly when l (the number of forgotten samples) remains low. This trend is more pronounced in larger models, consistent with their greater capacity to capture and retain class-discriminative features.

Figures 13 and 12 explore the implications of (k, l) settings on class balance, defined as the ratio of known to forgotten samples, for Pythia and Gemma 2 model families, respectively. The heatmaps indicate that this ratio grows with increasing k and decreasing l, with larger models showing a more marked divergence between known and forgotten categories. These findings highlight the sensitivity of probe performance and interpretability metrics to sampling configurations, underscoring the importance of systematic calibration of (k, l) pairs when designing probing protocols.
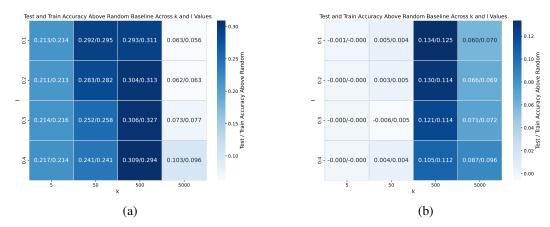
(a)

(b)

Figure 11: Accuracy gain over a random baseline from linear probes on: (a) Gemma 2 2B, and (b) Pythia 12B. Each cell displays the test/train accuracy above random for a given combination of (k, l). Darker blue indicates greater accuracy gains.
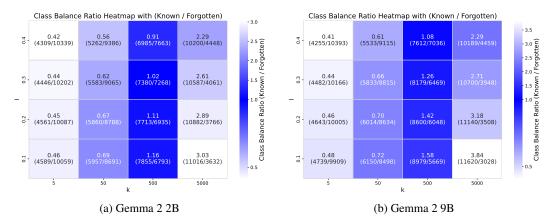

(a) Gemma 2 2B

(b) Gemma 2 9B

Figure 12: k-l parameters dependence on number of class balance (known and forgotten samples ratio) heatmaps for Gemma 2 models.
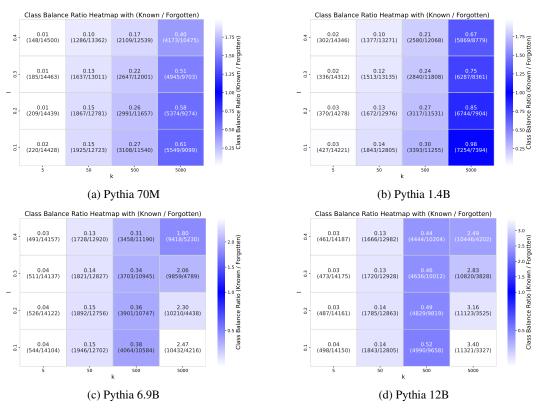
Figure 13: k-l parameters dependence on number of class balance (known and forgotten samples ratio) heatmaps for Pythia models.