RELATIONAL LINEARITY: PROBING LANGUAGE MODELS THROUGH THEIR OWN PREDICTIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

We study relational linearity in language models: whether a single linear map on the context representations alone ("*Yesterday, we watched a movie*") reproduces the model own next-token distribution for a fixed query ("*What is the tense of the previous sentence?*") over candidate answers ("*past*", "*present*", "*future*"). We introduce `LiRe`, a probe fitted by matching the logits of the prompted model, thus requiring no ground-truth annotation, and apply it at every layer of four language models on six queries of increasing abstraction. We find that relational linearity is more pronounced in larger models and tracks the abstraction a query demands: surface-level linguistic relations (language, verb tense) are already recoverable in early layers, whereas more abstract ones (subjectivity, faithfulness) peak in middle layers. It is also tied to how the query is formulated: *e.g.*, translating it, affects smaller models markedly and larger ones only mildly. Finally, relational linearity can be high even when the model most likely candidate answer disagrees with the ground-truth label, highlighting a structure that would be missed if probed only with ground-truth annotations.

1 INTRODUCTION

How do large language models encode knowledge about the world, specifically about its entities and their relations, in their latent representations? A growing line of work approaches this question by studying whether syntactic and semantic properties of textual inputs are reflected in linear structure in the representations of language models (Mikolov et al., 2013; Abdou et al., 2021; Elhage et al., 2022; Marks & Tegmark, 2024; Park et al., 2024; Engels et al., 2025; Wu et al., 2025). Such linear structure has been argued to support both the interpretation of a model internal computations (Geiger et al., 2021) and the control of model behavior through activation steering (Marks & Tegmark, 2024; Turner et al., 2023; Wu et al., 2024).

Relational linearity in language models (Paccanaro & Hinton, 2002; Hernandez et al., 2024b; Marconato et al., 2025) investigates whether relations between text sequences are encoded linearly in model representations. In the framing of Marconato et al. (2025), it is a property of the model itself: for a fixed *relation/query*, the model predictions given the *subject/context* and the *relation/query* can be recovered through a linear map from the *subject/context* representation alone. For example, for the subject *"Jimi Hendrix"* and the relation *"played at"*, this means recovering the model prediction of an object such as *"Woodstock"* from the representation of *"Jimi Hendrix"*, without appending the relation to the subject. Whether that prediction is factually correct is separate from whether it can be recovered linearly. Previous work has evaluated this linear structure through the recovery of annotated concepts or relation types (Chanin et al., 2024; Popovič & Färber, 2026), or through agreement with model predictions on examples where those predictions match the annotated objects (Hernandez et al., 2024b). Our work pursues a complementary goal: to disentangle relational linearity, as an intrinsic property of the model and its own next-token distributions, from whether its predictions agree with ground-truth answers.

To this end, we fit and evaluate linear probes against the model own answer distributions, without restricting the analysis to correctly answered contexts. First, we introduce an experimental procedure that permits evaluating relational linear properties as formulated in (Marconato et al., 2025), which relies on how the model shapes the next-token distribution of the tested queries. Second, we introduce a novel probing method, the Linear Relational Regressor (`LiRe`) requiring only the model

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

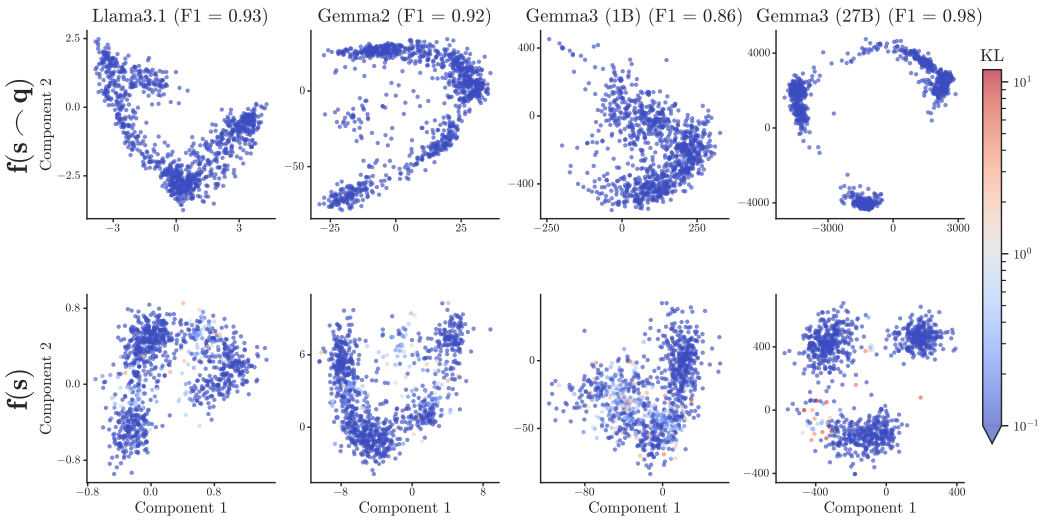


Figure 1: **All models exhibit relational linearity in the representations of the TENSE dataset**. For the models we test (`Llama3.1`, `Gemma2`, `Gemma3`(1B), and `Gemma3`(27B)), we compare the penultimate representations computed from contexts concatenated with the query $\mathbf{q}$ = *"What is the tense of the previous sentence?"*, denoted by $\mathbf{f}(\mathbf{s} \frown \mathbf{q})$, with those detected by the Linear Relational Regressor (`LiRe`) on the context representations $\mathbf{f}(\mathbf{s})$, for contexts $\mathbf{s}$ from the TENSE dataset (Ayman et al., 2024). The top row shows the reference representations $\mathbf{f}(\mathbf{s} \frown \mathbf{q})$, while the bottom row shows the corresponding `LiRe` representations $\mathbf{f}(\mathbf{s})$. Plots show the linear $2d$ projection (details in Section C.1), where points are coloured by their KL divergence to the reference next-token distribution induced by $\mathbf{f}(\mathbf{s} \frown \mathbf{q})$, with reference KL set to zero (all points scoring KL ≤ 0.01 are colored in blue). The smaller the KL, the closer the top representations are to being a linear transformation of the bottom ones.

latent representations both when it is prompted with (a) the context and the query of interest, and with (b) the context only, without relying on annotated context-query-answer.

We investigate relational linearity using our method across the internal activations of four language models (from small scale `Gemma3` (1B, Gemma Team et al., 2025) and `Gemma2` (2B, Gemma Team et al., 2024) to medium scale `Llama3.1` (8B, Grattafiori et al., 2024) and `Gemma3` (27B, Gemma Team et al., 2025)) and six queries of increasing complexity. In Fig. 1, we visualize the results obtained with `LiRe` on TENSE dataset (Ayman et al., 2024): for all models, we obtain a relatively small error in approximating the model target representations (top) by a linear transformation of the unprompted model representations (bottom). Overall, our experiments reveal the following:

a) Language models encode some relational properties largely linearly, while also exhibiting well-established layer-wise dynamics, with surface relations emerging early and abstract ones peaking in middle layers (Q1; Section 4.3);

b) Relational linear properties emerge also when model answers are at odds with ground-truth ones, showing that relational linearity is a property of the model own distribution rather than of some external linguistic variables (Q2; Section 4.4)

c) Paraphrasing relational queries yields varied effects on linear probing (Q3; Section 4.5).

2 PRELIMINARIES

We denote by $\mathcal{Y}$ a finite vocabulary of tokens. Textual inputs are elements $\mathbf{x} \in \mathcal{X}$, where $\mathcal{X}$ is a set of finite ordered sequences of tokens from $\mathcal{Y}$. We denote the operation of concatenation of token sequences by "$\frown$"; for example, if tokens y_1, y_2, y_3 are rendered as *"A"*, *"perfect"*, and *"circle"*, then $y_1 \frown y_2 \frown y_3$ is rendered as *"A perfect circle"*.

Given a textual input $\mathbf{x} \in \mathcal{X}$, autoregressive language models compute a sequence of hidden states through $L > 1$ transformer layers. We denote by $\mathbf{f}^{(\ell)}(\mathbf{x}) \in \mathbb{R}^d$, with $\ell \in \{1, \ldots, L\}$, the hidden state at layer ℓ used to predict the next token,[1] and write $\mathbf{f} := \mathbf{f}^{(L)}$ for the final-layer *embedding*. We use the term *representation* broadly to refer to such hidden states, covering both intermediate-layer hidden states and final-layer embeddings. The model also has an output *unembedding* map $\mathbf{g} : \mathcal{Y} \to \mathbb{R}^d$, which projects hidden states into the output vocabulary by associating each token $y \in \mathcal{Y}$ with an unembedding vector. Together, $\mathbf{f}$ and $\mathbf{g}$ define the next-token distribution

$$p_{\mathbf{f},\mathbf{g}}(y \mid \mathbf{x}) = \frac{\exp(\mathbf{f}(\mathbf{x})^\top \mathbf{g}(y))}{\sum_{y' \in \mathcal{Y}} \exp(\mathbf{f}(\mathbf{x})^\top \mathbf{g}(y'))}. \tag{1}$$

2.1 Relational Linearity

Following the setting by Paccanaro & Hinton (2002), we consider textual strings that can be decomposed as triplets $\mathbf{s} \frown \mathbf{q} \frown y$, where $\mathbf{s}, \mathbf{q} \in \mathcal{X}$ and $y \in \mathcal{Y}$. These can be either *subject-relation-object* triplets, *e.g.*, $\mathbf{s}$ ="*David Gilmour*", $\mathbf{q}$ ="*plays the*", and y ="*guitar*", or, more generally, as *context-query-answer* triplets, *e.g.*, $\mathbf{s}$ ="*Today I'm happy.*", $\mathbf{q}$ ="*Is it a positive statement?*", and y ="*Yes*".[2] We will refer to $\mathbf{s}$ as to the *context* (or the *subject*), to $\mathbf{q}$ as the *query* (or the *relation*), and to y as the *answer* (or the *object*).

We mainly follow the setup described by Marconato et al. (2025) and analyze *linear relational probing*,[3] that is, when the probability distribution on a set of tokens $\mathcal{Y}_P \subseteq \mathcal{Y}$ of the query-prompted model can be well approximated by a linear map with a softmax activation from the unprompted model embeddings $\mathbf{f}(\mathbf{s})$. As in Hernandez et al. (2024b), we extend it to all layers of language models. Formally, let the conditional distribution restricted to tokens in $\mathcal{Y}_P$ be defined as $p(y \mid \cdot\,; \mathcal{Y}_P) := p(y \mid \cdot) \Big/ \sum_{y' \in \mathcal{Y}_P} p(y' \mid \cdot)$. We define linear relational probing as follows:

Definition 1 (Linear Relational Probing)**.** *For a query $\mathbf{q} \in \mathcal{X}$ and a subset of tokens $\mathcal{Y}_P \subseteq \mathcal{Y}$ of dimension $k = |\mathcal{Y}_P|$, a model $(\mathbf{f}, \mathbf{g})$ can be* linearly probed for $\mathbf{q}$ on the subset $\mathcal{Y}_P$ at layer ℓ *if there exist k weights $\mathbf{w}_\mathbf{q}^y \in \mathbb{R}^d$ and biases $b_\mathbf{q}^y \in \mathbb{R}$ such that, for all $y \in \mathcal{Y}_P$ and $\forall \mathbf{s} \in \mathcal{X}$,*

$$p_{\mathbf{f},\mathbf{g}}(y \mid \mathbf{s} \frown \mathbf{q}; \mathcal{Y}_P) = \frac{\exp(\mathbf{w}_\mathbf{q}^{y,\top} \mathbf{f}^{(\ell)}(\mathbf{s}) + b_\mathbf{q}^y)}{\sum_{y' \in \mathcal{Y}_P} \exp(\mathbf{w}_\mathbf{q}^{y',\top} \mathbf{f}^{(\ell)}(\mathbf{s}) + b_\mathbf{q}^{y'})}. \tag{2}$$

The following example illustrates the setting:

> **Example 1.** *Consider a query $\mathbf{q}$ ="What is the tense of the previous sentence?" and consider three possible answers $\mathcal{Y}_P$ = {"past", "present", "future"}. Take as context $\mathbf{s}$ ="Yesterday, we watched a movie", to which the language model assigns high-probability to "past" token (with probability $1 - 2\epsilon$) and vanishing probability to "present" and "future" (both with probability ϵ) when prompted with input $\mathbf{s} \frown \mathbf{q}$.*
>
> *The language model can be linearly probed for $\mathbf{q}$ on $\mathcal{Y}_P$ at layer ℓ if this probability distribution can be well reconstructed from the embeddings $\mathbf{f}(\mathbf{s})$ with a linear classifier and softmax activation (according to Equation 2 and for multiple different contexts $\mathbf{s} \in \mathcal{X}$).*
>
> *In Fig. 1, we observe this in `Llama3.1`, `Gemma2`, `Gemma3`(1B), and `Gemma3`(27B).*

Intuitively, Definition 1 ensures that, for all prompted contexts, the probability distribution on the relevant next-tokens for a query can be obtained through a linear probe of the representations of the context alone.

[1] Each transformer layer produces hidden states at all input positions; $\mathbf{f}^{(\ell)}(\mathbf{x})$ denotes the final-position hidden state used for next-token prediction. We consider only these layer outputs, not internal components such as attention heads or MLP blocks.

[2] The definition is not restricted to well-formed sentences: both the query $\mathbf{q}$ and sequences $\mathbf{s}$ could be arbitrary strings, such as *"@!as1smf kqw#ù"*.

[3] Marconato et al. (2025) introduce a more general framework for relational linear properties of which our Definition 1 is a special case. We provide the general definition in Definition 2.

2.2 Relaxation of Linear Relational Probing

We also consider a weaker version of linear relational probing that only involves the likeliest tokens. In this setting, we do not need to consider the full equivalence between distributions as in Equation 2, but only the match among the likeliest tokens. This specializes Equation 2 to the following (see Definition 3 for a complete description):

$$\operatorname*{argmax}_{y \in \mathcal{Y}_P} p_{\mathbf{f},\mathbf{g}}(y \mid \mathbf{s} \frown \mathbf{q}; \mathcal{Y}_P) = \operatorname*{argmax}_{y \in \mathcal{Y}_P} \left(\mathbf{w}_{\mathbf{q}}^{y,\top} \mathbf{f}^{(\ell)}(\mathbf{s}) + b_{\mathbf{q}}^{y} \right). \tag{3}$$

Notice that Equation 2 imposes a stronger condition than Equation 3, as the former equation implies the latter, but the opposite direction does not hold. A model can satisfy weak linear relational probing without matching the full reference distribution. E.g., both the probe and the prompted model may assign the highest probability to *"past"* in Example 1, but assign it different probabilities.

3 Testing Relational Linearity

Here, we present our main experimental setting and discuss how to estimate relational linearity. We first note a key aspect of relational linearity that allows for straightforward evaluation (Section 3.1), then we introduce the main setup for testing (Section 3.2), and finally `LiRe` (Section 3.3).

3.1 Practical Implication of Definition 1

Linear relational probing characterizes a property intrinsic to the language model and its own next-token distributions: whether a linear probe can approximate well the model answer distributions from context representations alone. This differs from probing against ground truth, which tests whether representations encode externally annotated properties of language (Alain & Bengio, 2016; Kim et al., 2018; Marks & Tegmark, 2024; Park et al., 2024). From a practical perspective, this has two immediate consequences.

First, estimating relational linearity does only require, for a given query $\mathbf{q}$, prompting the model with only the set of contexts and the query $(\mathbf{s}, \mathbf{q})$, whereas previous works (*e.g.*, Hernandez et al., 2024b; Chanin et al., 2024) need also annotations on the answer to get well-constructed triples $(\mathbf{s}, \mathbf{q}, y)$. Second, relational linearity can hold even when the model predictions disagree with ground truth. However, it can also hold trivially when the model answer distribution is independent of the context. For example, if $p_{\mathbf{f},\mathbf{g}}(y \mid \mathbf{s} \frown \mathbf{q}; \mathcal{Y}_P) = c_y$ for every context $\mathbf{s} \in \mathcal{X}$, with $c_y > 0$ and $\sum_{y \in \mathcal{Y}_P} c_y = 1$, Definition 1 is satisfied by setting $\mathbf{w}_{\mathbf{q}}^{y} = \mathbf{0}$ and tuning the biases to $b_{\mathbf{q}}^{y} = \log c_y$. More generally, distributions concentrated in a small region of the probability simplex may yield high probing performance without substantial context-dependent variation.

We propose to detect these trivial scenarios introducing a **collapse on the simplex score** (CSS) that leverages the normalized Shannon entropy $\mathrm{H}_k(\cdot)$[4] on a dataset of model probabilities $\{p_{\mathbf{f},\mathbf{g}}(\cdot \mid \mathbf{s}_i \frown \mathbf{q})\}_{i=1}^{N}$, that is

$$\mathrm{CSS} := 1 - \mathrm{H}_k\left(\overline{p_{\mathbf{f},\mathbf{g}}(\cdot \mid \mathbf{s}_i \frown \mathbf{q})}\right) + \overline{\mathrm{H}_k\left(p_{\mathbf{f},\mathbf{g}}(\cdot \mid \mathbf{s}_i \frown \mathbf{q})\right)}, \tag{4}$$

where $\bar{\cdot}$ denotes the average over the dataset. CSS ranges in $[0, 1]$, where high values indicate a collapse towards $p_{\mathbf{f},\mathbf{g}}(y \mid \mathbf{s} \frown \mathbf{q}) = \mathrm{const}$, whereas low values indicate that model probabilities span multiple regions of the simplex. We empirically validate it in Table 1.

Next, we discuss the experimental setup to probe relational linearity.

3.2 Setup

We assume to have access to a set of contexts $\mathcal{S} = \{\mathbf{s}_i \in \mathcal{X}\}_{i=1}^{N}$. For each given query $\mathbf{q} \in \mathcal{X}$, we design a prompt $\tilde{\mathbf{q}}$ to elicit the model for being more predictive of the relevant tokens $\mathcal{Y}_P$ chosen for the query. In this setting, we test a slight variation of Definition 1, where the original query is rewritten in a system prompt $\tilde{\mathbf{q}}$ and the order of $\mathbf{s}$ and $\tilde{\mathbf{q}}$ might be swapped:[5]

[4] It uses a base-k logarithm, where k is the token count in $\mathcal{Y}_P$.

[5] *e.g.*, a system prompt prepended to an LLM user's input. This is the case for `Llama3.1`, while all `Gemma` models maintain the structure we exposed so far.

Example 2. *For the query* $\mathbf{q}$ *="Does the previous sentence reflect a truthful statement?" and tokens* $\mathcal{Y}_P = \{$*"Yes", "No"*$\}$*, the system prompt is:*

$\tilde{\mathbf{q}}$ *="You are a linguistic classifier. Respond with only one word.* ***Task: Identify the truthfulness of a sentence. Options: Yes, No.*** *Truthfulness:".*

From this, we evaluate the probability distribution of the model $p_{\mathbf{f},\mathbf{g}}(y_j \mid \mathbf{s} \frown \tilde{\mathbf{q}}; \mathcal{Y}_P)$ for all relevant tokens $y_j \in \mathcal{Y}_P$. This gives a dataset of input context embeddings and probability pairs $\mathcal{D}_{\mathbf{q}} = \{\big(\mathbf{f}(\mathbf{s}_i), p_{\mathbf{f},\mathbf{g}}(\cdot \mid \mathbf{s}_i \frown \tilde{\mathbf{q}}; \mathcal{Y}_P)\big)\}_{i=1}^{N}$.

Paraphrases. The prompt $\mathbf{q}$ expresses a natural-language question. Paraphrasing this question yields an alternative query string $\tilde{\mathbf{q}}$, which might in turn have a different natural set of relevant tokens $\mathcal{Y}_P$. The following example illustrates this:

Example 3. *As in example 2, we consider the query* $\mathbf{q}$ *="Does the previous sentence reflect a truthful statement?" but now with tokens* $\mathcal{Y}'_P = \{$*"True", "False"*$\}$*, The system prompt we use in this setting is a paraphrase of* $\tilde{\mathbf{q}}$*:*

$\tilde{\mathbf{q}}'$ *="You are a linguistic classifier. Respond with only one word.* ***Task: Classify a sentence as true or false. Options: True, False****. Truthfulness:".*

A natural question is whether different textual alternatives of a query exhibit comparable relational linearity, which we empirically evaluate in Section 4 for two distinct relations.

3.3 LiRe

To address the question of whether a language model encodes any relational linearity, we introduce the Linear Relational Regressor (`LiRe`), specifically tailored to test linear relational probing using the logits of a language model. `LiRe` uses a linear model, with k weights $\mathbf{w}_y \in \mathbb{R}^d$ and biases $b_y \in \mathbb{R}$, to define a conditional distribution over the set of tokens in $\mathcal{Y}_P$, given by

$$p^{(\ell)}_{\mathbf{W},\mathbf{b}}(y \mid \mathbf{s}) = \exp(\mathbf{w}_y^\top \mathbf{f}^{(\ell)}(\mathbf{s}) + b_y) \Big/ \textstyle\sum_{y' \in \mathcal{Y}_P} \exp(\mathbf{w}_{y'}^\top \mathbf{f}^{(\ell)}(\mathbf{s}) + b_{y'}) \ . \tag{5}$$

The weights and biases are trained (for more details, ref. to Section C.2) by minimizing the mean-squared error between the logits of the Linear Relational Regressor and of the logits of the reference distribution $p_{\mathbf{f},\mathbf{g}}(y \mid \mathbf{s} \frown \mathbf{q}; \mathcal{Y})$, also known as the logit distance (Nielsen et al., 2026):

$$(\hat{\mathbf{W}}, \hat{\mathbf{b}}) = \underset{\mathbf{W},\mathbf{b}}{\operatorname{argmin}} \frac{1}{|\mathcal{D}_{\mathbf{q}}|} \sum_{\mathbf{s} \in \mathcal{D}_{\mathbf{q}}} \sum_{y \in \mathcal{Y}_P} \Big[\log p_{\mathbf{f},\mathbf{g}}(y \mid \mathbf{s} \frown \mathbf{q}) - \log p^{(l)}_{\mathbf{W},\mathbf{b}}(y \mid \mathbf{s})\Big]^2 \ . \tag{6}$$

4 Empirical Verification

Section 4.1 describes our experimental setup, while Section 4.2 evaluates the model next-token performance in our setup. Afterwards, Sections 4.3 to 4.5 address the following research questions:

- **Q1** Do language models encode any linear relational property, and to what degree? How does relational linearity vary across layers of these models?
- **Q2** Do these hold even when model predictions do not match ground-truth annotations?
- **Q3** Are paraphrases of the queries similarly encoded in model representations?

4.1 Setting

Language Models. We adopt two different language models: `Llama3.1`(8B) Grattafiori et al., 2024, which has 32 hidden layers and an embedding size of 4096; `Gemma2`(2B) Gemma Team et al., 2024), which has 26 hidden layers and embedding size of 2304; `Gemma3`(1B) Gemma Team et al., 2025), which has 26 hidden layers and embedding size of 1152; `Gemma3`(27B) Gemma Team

Table 1: Performance of language models in predicting the ground-truth label when prompted with the query appended. It also reports the CSS, and a `Majority` class baseline. Best across columns are marked in **bold**. Models scoring `CSS` above 0.75 are marked in **red**.

	LANG		TENSE		SUBJ		TRUTH		PROP		AWARE	
	F1	CSS	F1	CSS	F1	CSS	F1	CSS	F1	CSS	F1	CSS
`Majority`	0.05	–	0.37	–	0.64	–	0.50	–	0.50	–	0.57	–
`Llama3.1`	0.97	0.02	0.88	0.16	0.60	0.19	0.87	0.09	0.44	0.80	**0.58**	0.80
`Gemma2`	0.79	0.10	0.80	0.26	0.54	0.33	0.86	0.10	**0.65**	0.64	0.38	0.87
`Gemma3`(1B)	0.37	0.37	0.59	0.37	0.57	0.22	0.69	0.10	0.50	0.54	0.33	0.78
`Gemma3`(27B)	**0.98**	**0.01**	**0.97**	**0.01**	**0.63**	**0.01**	**0.88**	**0.02**	0.60	**0.28**	0.51	**0.37**

et al., 2025), which has 62 hidden layers and embedding size of 5376. All models are instruction-tuned, and we apply 4-bit quantization to `Gemma3`(27B) (Dettmers et al., 2022).

Methods. To detect linear relational properties, encoded in the neural embeddings of language models, we contrast `LiRe` to a `Random` baseline that defines the null hypothesis for relational linearity. It uses the `LiRe` method, but is trained with randomly permuted embeddings (see Equation 9). As an additional baseline, we implement the Linear Relational Embedding (`LRE`) method proposed by Hernandez et al. (2024b) but, due to suboptimal performance, we confine all experimental results in Section B. Moreover, we also train three (ground-truth) supervised baselines. As these methods are not designed to detect relational linearity, we compare them to `LiRe` in Section C.8.

Metrics. We measure the F1 score to the model most likely token (Equation 3), denoted as F1(LLM). We use ground-truth query answers to measure F1 score of model and probe predictions, denoted as F1(GT). We also measure the KL divergence between model and `LiRe` distribution, refer to Section C. As a sanity check, we measure the `CSS` score (Equation 4) when concatenating contexts and queries. We evaluate the mean and standard deviation with a 10-fold cross-validation.

Datasets. We consider six sentence-level datasets, each one annotated with a distinct linguistic attribute. Specifically, we study relational linearity on different syntactical and semantic queries:

- LANG (Papariello, 2021). Each sentence is annotated with the origin language, for a total of 20 distinct idioms (each one ranging from 3.9% to 5.2% of the total). For our experiments, we consider only a subset of the total data, including 9542 samples.
- TENSE (Ayman et al., 2024). Here, sentences are written in English and are labelled according to their main verbal tense (either *past*, *present*, or *future*). Labels are in proportion 36.9% *present*, 33.8% *future*, 29.3% *past*. The dataset includes 10k samples.
- SUBJ (Alam et al., 2025). Sentences are labeled based on whether they express an opinion or a fact. This multilingual dataset covers Arabic (37%), German (23%), English (14%), Italian (14%), and Bulgarian (12%). The dataset counts 8090 samples.
- TRUTH (Marks & Tegmark, 2024). Sentences are binary labelled on factual accuracy, indicating whether they reflect true or false statements. The two classes are perfectly balanced. All sentences are written in English. The dataset contains 9123 samples.
- PROP (Hasanain et al., 2025). Contexts are labelled according to a binary label on whether they employ a propagandistic framing (e.g. loaded language, appeal to fear). The dataset covers English (50%) and Arabic (50%), and the two classes are balanced, for a total of 6280 samples.
- AWARE (Needham et al., 2025). Each input is a transcript of an interaction with a language model, labelled according to whether it originates from an evaluation or from real-world deployment. Labels are 57.4% for *Test* and 42.6% for *Real*. Transcripts are in English, counting 500 samples.

Queries. For each dataset, we separately construct a textual query designed to elicit model outputs that match the dataset's labels. For example, for the LANG relation, we use a system prompt with the query "What is the written language?", allowing 20 possible token-level answers corresponding to the languages in the dataset (see also Table 5). Throughout the paper, we refer to each query by its dataset name; for example, the above is denoted as the LANG query.

4.2 PERFORMANCE OF LANGUAGE MODELS ON NEXT-TOKEN PREDICTION

We explore the capabilities of language models to correctly predict the ground truth label on the four datasets. Results are presented in Table 1. The majority class baseline provides a reference for comparing LLM capabilities. The collapse score (Equation 4) is also reported as a preliminary measure of how indicative subsequent probing results are of relational linearity; values near 1 suggest less meaningful relational probing outcomes. We highlight that `Llama3.1` on the PROP query and all models exception made for `Gemma3`(27B) on AWARE query show high collapse (with a peak ≈ 0.87 for `Gemma2`), implying these relations are almost trivially encoded, as the model output probabilities collapse into a narrow region. Therefore, **we interpret probing results for these high-collapse model–query pairs with caution and report them separately** in Section C.

`Gemma3`(27B) lead in F1(GT), exception made for the more challenging queries PROP and AWARE, where more collapsed `Gemma2` and `Llama3.1` fare better, respectively. `Llama3.1` in non-collapsed regimes is second best and almost always above the `Majority` class voting. `Gemma2` follows, winning in F1(GT) in PROP but with high `CSS`. `Gemma3`(1B) always worsen to other methods, exception for SUBJ where fares slightly above `Gemma2`, with higher drops on LANG and TENSE queries. Since `LiRe` is trained directly on the model predictions instead of ground-truth, lower results in F1(GT) for the query-prompted model do not invalidate possible relational linearity.

4.3 Q1: DO LANGUAGE MODELS ENCODE ANY LINEAR RELATIONAL PROPERTY?

We first compare `LiRe` with the Linear Relational Embedding (`LRE`) method of Hernandez et al. (2024b) at matched intermediate layers of `Llama3.1`, `Gemma2`, and `Gemma3`(1B). Due to the high computational cost of `LRE`, we could not evaluate this method on the larger `Gemma3`(27B).

Across all evaluated model-query pairs, `LiRe` achieves higher F1(LLM) and lower KL divergence, indicating closer recovery of the prompted model predictions and answer distributions (Table 3, Section B.2). The gap is particularly pronounced for LANG and TENSE in `Llama3.1`, where `LiRe` closely reproduces the model answer distributions. `LRE` comes closest to `LiRe` on TRUTH, particularly for `Llama3.1`. We remark that `LiRe` directly targets the property in Definition 1 by fitting a linear probe to recover the prompted model answer distribution from context-only representations. `LRE` instead constructs the probe through a local linear approximation of the model computation. Thus, the improved distributional recovery achieved by `LiRe` provides stronger empirical evidence of approximate relational linearity under our definition, including in cases where `LRE` yields poor recovery. In addition to its closer recovery, our method reduces the reported computational cost by roughly three orders of magnitude (Table 2). Motivated by these results, we focus on `LiRe` in the subsequent layer-wise and paraphrase analyses. In Section B, we provide the detailed comparison.

Layer-wise analysis. Fig. 2 reports F1(LLM), which measures agreement between `LiRe` and the prompted model most likely candidate answer. For `Llama3.1` and `Gemma3`(27B), LANG and TENSE already achieve high agreement at early layers. At the first evaluated layer, LANG scores are above 0.95, while TENSE scores are above 0.85 and improve further with depth. By contrast, SUBJ and TRUTH generally improve toward the middle of the network. SUBJ typically peaks around intermediate layers and then declines, whereas TRUTH often maintains a broad plateau, with some models attaining their highest scores later. The smaller models show broadly similar depth profiles for TENSE and TRUTH, but substantially lower agreement on LANG, especially `Gemma3`(1B).

Next, we focus on more abstract queries for PROP and AWARE. For PROP, the models below the `CSS` threshold of 0.75 show moderate improvements toward intermediate layers, with peak F1(LLM) values of 0.82 for `Gemma2`, 0.79 for `Gemma3`(1B), and 0.62 for `Gemma3`(27B). On AWARE, `Gemma3`(27B), the only model below this threshold, reaches approximately 0.75, with substantial variability across folds. High `CSS` limits the interpretation of the remaining model-query pairs as evidence of non-trivial linear relational probing. Overall, the results are consistent with early recoverability of surface linguistic properties and improvements at intermediate layers for some more abstract queries, mirroring observations from prior work (Section A; *e.g.*, Tenney et al. (2019); Cheng et al. (2025)), while showing important differences across models and queries.

To assess distribution-level linear relational probing (Definition 1), we also measure KL divergence between `LiRe` and the prompted model over the candidate answers. The layer-wise profiles broadly support the answer-agreement results: distribution matching is already strong at early layers

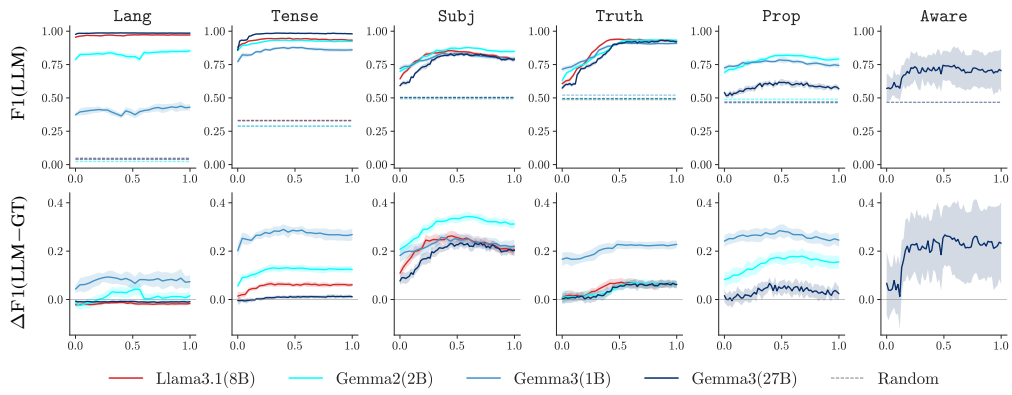


Figure 2: Linear relational probing across layers (normalized into the range 0–1), comparing `LiRe` on `Llama3.1`, `Gemma2`, `Gemma3`(1B), and `Gemma3`(27B). The complete plots are in Fig. 5.

for LANG in the larger models and improves substantially toward intermediate layers for TRUTH. However, improvement over `Random` need not always imply a close distributional match. E.g., `Gemma3`(1B) on LANG has a minimum KL of 1.55, while `Gemma3`(27B) on PROP remains above 4.1 despite improving over its `Random` reference of 7.65. This evidence supports approximate distribution recovery in several settings, rather than strong matching across all models and queries. Full results are reported in Section C.7.

4.4 Q2: How relational linearity is intrinsic to the model predictions vs to the ground-truth annotations?

We compare `LiRe` predictions against two targets: the prompted model most likely candidate answers, measured by F1(LLM), and the annotated labels, measured by F1(GT). Their difference is $\Delta\text{F1} := \text{F1(LLM)} - \text{F1(GT)}$. A positive gap means that the probe agrees more closely with the model than with ground truth. High F1(LLM) together with a positive gap therefore supports separating recovery of model behavior from recovery of the annotated property. This gap is distinct from the prompted model own ground-truth score in Table 1.

For LANG and TENSE, `Llama3.1` and `Gemma3`(27B) show relatively small gaps. The smaller models exhibit a clearer separation on TENSE: at the middle layer, `Gemma2` has a gap of 0.13, while `Gemma3`(1B) achieves F1(LLM) of 0.88 and F1(GT) of 0.59, yielding a gap of 0.29. Thus, strong recovery of the model answers can coexist with substantially lower agreement with the annotated labels. This distinction also appears on more abstract queries. At the middle layer, SUBJ gaps range from 0.23 to 0.34 across models. For TRUTH, `Gemma3`(1B) has the largest gap, approximately 0.23, compared with 0.06–0.07 for the other models. For PROP, `Gemma2` and `Gemma3`(1B) have middle-layer gaps of 0.17 and 0.28, respectively, whereas `Gemma3`(27B) has a smaller gap of 0.05. On AWARE,`Gemma3`(27B)'s mean gap remains approximately 0.21–0.27 over the second half of the network, with substantial variability across folds. These observations support relational linearity as a property of the model predictions that need not coincide with accurate ground-truth predictions.

The comparison with standard supervised probes in Section C.8 (Fig. 6) illustrates the practical consequence of this gap: probes trained on ground-truth labels can recover those labels well while agreeing less closely with the model answers, often failing to detect relational linearity.

4.5 Q3: Are paraphrases of the queries similarly encoded in model representations?

We test alternative formulations of LANG and TRUTH: English, Italian, Spanish, and Chinese instructions and answer labels for LANG, and *True/False*, *Valid/Invalid*, and *Yes/No* formulations for TRUTH (Table 6). These changes affect both query wording and candidate answers.

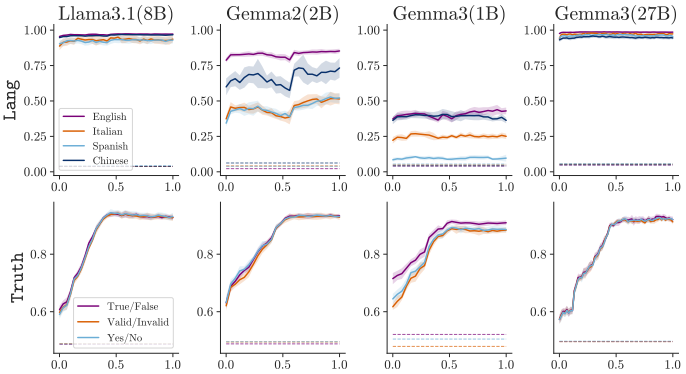


Figure 3: We test linear relational probing through LiRe. The plots show the F1(LLM) for all models on the LANG (cross-lingual queries) and TRUTH (diverse binary labels) datasets. Paraphrases can both change the query formulation and on its relevant tokens. Dotted lines denote the random baselines.

Fig. 3 shows greater sensitivity on LANG for the smaller models. `Llama3.1` and `Gemma3`(27B) maintain high F1(LLM) across formulations, whereas `Gemma2` and `Gemma3`(1B) show larger differences. For `Gemma2`, Chinese performs below English but above Italian and Spanish. For `Gemma3`(1B), English and Chinese yield broadly similar scores, while Italian and especially Spanish reduce agreement. TRUTH is comparatively stable across formulations, except for `Gemma3`(1B), whose *True/False* formulation yields higher agreement, particularly at early layers.

Overall, the tested formulations affect answer-level relational probing more strongly in the smaller models, especially on LANG. Distribution matching on TRUTH is comparatively stable across formulations; see Appendix D, Figs. 7 and 8, for the ΔF1 and the layer-wise KL analysis across paraphrases. Table 7 additionally reports the ground-truth and collapse analyses.

5 CONCLUSION

We investigate relational linearity through `LiRe`, a linear regressor trained to recover a language model query-conditioned answer distribution from context representations alone, without ground-truth supervision. Compared with `LRE` (Hernandez et al., 2024b) at matched intermediate layers, `LiRe` achieves closer recovery of model answer distributions across all evaluated model-query pairs, while reducing the reported computational cost by roughly three orders of magnitude. Across four models and six queries, our experiments (discussed in Section 4 and Section C) support the model-intrinsic formulation of relational linearity developed by Marconato et al. (2025). (**Q1**) Recovery varies across models, queries, and layers: LANG and TENSE predictions are often recoverable early, whereas SUBJ and TRUTH generally improve toward middle layers. Propaganda detection (PROP) and evaluation awareness (AWARE) yield weaker or inconclusive evidence for non-trivial relational linearity, with limited recovery or substantial collapse. (**Q2**) High agreement with model predictions can coexist with substantially lower agreement with ground-truth annotations, and (**Q3**) sensitivity to the tested query formulations is greater in smaller models. Together, these findings support relational linearity at both the answer and distribution levels, while distinguishing it from factual correctness and accounting for trivial agreement due to output collapse.

In future work, we plan to extend this evaluation to nonlinear relational mappings, similarly to (Engels et al., 2025), and to tensor-factorized models of context–query–answer interactions inspired by knowledge graph embeddings (Sun et al., 2019; Loconte et al., 2023).

Limitations. We deliberately use annotated datasets to compare `LiRe` with supervised probing baselines on common inputs and to track F1(LLM) alongside F1(GT), separating agreement with the model from agreement with ground truth. Ground-truth labels are not required to fit `LiRe` or evaluate its agreement with the model. Nevertheless, our findings concern the selected context distributions, six queries, and finite candidate-answer sets; they do not establish that the same behavior extends to arbitrary contexts, ambiguous or inapplicable queries, or unrestricted generation.

Our study focuses on recovering the model query-conditioned predictions, but does not establish whether the recovered maps also enable changes to these predictions through activation steering. Establishing when `LiRe`-derived maps enable selective changes to model outputs, while preserving responses to other queries, requires intervention experiments beyond the scope of the present study.

AI USE STATEMENT

In this work, we used generative AI tools for code review, to design the plot interface, to review the current manuscript, and for a general literature scan. We have not used generative AI tools for the conceptualization of this work, for the design of `LiRe`, or to write the main experimental setup. We have not used generative AI tools for other tasks with required disclosure. We have reviewed all AI-assisted work, by running extensive checks and by manually editing generated code. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

All experimental details can be found in Section 4 and in Section C. We also attached an anonymized version of the code in the supplementary material, which **will be published upon acceptance**.

REFERENCES

Mostafa Abdou, Artur Kulmizev, Daniel Hershcovich, Stella Frank, Ellie Pavlick, and Anders Søgaard. Can language models encode perceptual structure without grounding? A case study in color. In *Proceedings of the 25th conference on computational natural language learning*, pp. 109–132, 2021.

Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*, 2016.

Firoj Alam, Julia Maria Struß, Tanmoy Chakraborty, Stefan Dietze, Salim Hafid, Katerina Korre, Arianna Muti, Preslav Nakov, Federico Ruggeri, Sebastian Schellhammer, Vinay Setty, Megha Sundriyal, Konstantin Todorov, and Venktesh V. The clef-2025 checkthat! lab: Subjectivity, fact-checking, claim normalization, and retrieval. In Claudia Hauff, Craig Macdonald, Dietmar Jannach, Gabriella Kazai, Franco Maria Nardini, Fabio Pinelli, Fabrizio Silvestri, and Nicola Tonellotto (eds.), *Advances in Information Retrieval*, pp. 467–478, Cham, 2025. Springer Nature Switzerland. ISBN 978-3-031-88720-8.

Umme Ayman, Md Hafizur Rahman, and Md Shafiqul Islam. Englishtense: A large scale english texts dataset categorized into three categories: Past. *Present, Future tenses*, 2024.

David Chanin, Anthony Hunter, and Oana-Maria Camburu. Identifying linear relational concepts in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 1524–1535, 2024.

Emily Cheng, Diego Doimo, Corentin Kervadec, Iuri Macocco, Lei Yu, Alessandro Laio, and Marco Baroni. Emergence of a high-dimensional abstraction phase in language transformers. In *The Thirteenth International Conference on Learning Representations*, 2025.

Miranda Anna Christ, Adrián Csiszárik, Gergely Becsó, and Dániel Varga. The structure of relation decoding linear operators in large language models. *Advances in Neural Information Processing Systems*, 38:43187–43215, 2026.

Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. Gpt3. int8 (): 8-bit matrix multiplication for transformers at scale. *Advances in neural information processing systems*, 35: 30318–30332, 2022.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021. https://transformer-circuits.pub/2021/framework/index.html.

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, et al. Toy models of superposition. *arXiv preprint arXiv:2209.10652*, 2022.

Joshua Engels, Isaac Liao, Eric J Michaud, Wes Gurnee, and Max Tegmark. Not all language model features are one-dimensionally linear. In *International Conference on Learning Representations*, pp. 84591–84622, 2025.

Javier Ferrando, Gabriele Sarti, Arianna Bisazza, and Marta R Costa-Jussà. A primer on the inner workings of transformer-based language models. *arXiv preprint arXiv:2405.00208*, 2024.

Atticus Geiger, Hanson Lu, Thomas Icard, and Christopher Potts. Causal abstractions of neural networks. *Advances in neural information processing systems*, 34:9574–9586, 2021.

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. *arXiv preprint arXiv:2408.00118*, 2024.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, et al. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025.

Navita Goyal, Hal Daumé III, Alexandre Drouin, and Dhanya Sridhar. Causal differentiating concepts: Interpreting lm behavior via causal representation learning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Maram Hasanain, Fatema Ahmad, and Firoj Alam. Can GPT-4 identify propaganda? annotation and detection of propaganda spans in news articles. In Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue (eds.), *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pp. 2724–2744, Torino, Italia, May 2024. ELRA and ICCL. URL https://aclanthology.org/2024.lrec-main.244.

Maram Hasanain, Md Arid Hasan, Mohamed Bayan Kmainasi, Elisa Sartori, Ali Ezzat Shahroor, Giovanni Da San Martino, and Firoj Alam. PropXplain: Can LLMs enable explainable propaganda detection? In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 23855–23863, 2025.

Evan Hernandez, Belinda Z. Li, and Jacob Andreas. Inspecting and editing knowledge representations in language models. In *First Conference on Language Modeling*, 2024a.

Evan Hernandez, Arnab Sen Sharma, Tal Haklay, Kevin Meng, Martin Wattenberg, Jacob Andreas, Yonatan Belinkov, and David Bau. Linearity of relation decoding in transformer language models. In *International Conference on Learning Representations*, pp. 10504–10526, 2024b.

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*, pp. 2668–2677. PMLR, 2018.

Lorenzo Loconte, Nicola Di Mauro, Robert Peharz, and Antonio Vergari. How to turn your knowledge graph embeddings into generative models. *Advances in Neural Information Processing Systems*, 36:77713–77744, 2023.

Ang Lv, Yuhan Chen, Kaiyi Zhang, Yulong Wang, Lifeng Liu, Ji-Rong Wen, Jian Xie, and Rui Yan. Interpreting key mechanisms of factual recall in transformer-based language models. *arXiv preprint arXiv:2403.19521*, 2024.

Emanuele Marconato, Sébastien Lachapelle, Sebastian Weichwald, and Luigi Gresele. All or none: Identifiable linear properties of next-token predictors in language modeling. *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2025.

Samuel Marks and Max Tegmark. The geometry of truth: Emergent linear structure in large language model representations of true/false datasets. In *First Conference on Language Modeling*, 2024.

Jack Merullo, Carsten Eickhoff, and Ellie Pavlick. Language models implement simple word2vec-style vector arithmetic. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 5030–5047, 2024.

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.

Joe Needham, Giles Edkins, Govind Pimpale, Henning Bartsch, and Marius Hobbhahn. Large language models often know when they are being evaluated, 2025. *URL https://arxiv.org/abs/2505.23836*, 2025.

Beatrix Miranda Ginn Nielsen, Emanuele Marconato, Luigi Gresele, Andrea Dittadi, and Simon Buchholz. Logit distance bounds representational similarity. In *Forty-third International Conference on Machine Learning*, 2026.

Alberto Paccanaro and Geoffrey E. Hinton. Learning distributed representations of concepts using linear relational embedding. *IEEE Transactions on Knowledge and Data Engineering*, 13(2): 232–244, 2002.

Luca Papariello. Language-identification. https://huggingface.co/datasets/papluca/language-identification, November 2021. Accessed: 2025-27-06.

Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models. In *International Conference on Machine Learning (ICML)*, pp. 39643–39666. PMLR, 2024.

Jakub Piskorski, Nicolas Stefanovitch, Giovanni Da San Martino, and Preslav Nakov. SemEval-2023 task 3: Detecting the category, the framing, and the persuasion techniques in online news in a multi-lingual setup. In Atul Kr. Ojha, A. Seza Doğruöz, Giovanni Da San Martino, Harish Tayyar Madabushi, Ritesh Kumar, and Elisa Sartori (eds.), *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pp. 2343–2361, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.semeval-1.317. URL https://aclanthology.org/2023.semeval-1.317/.

Nicholas Popovič and Michael Färber. Tracing relational knowledge recall in large language models. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 43490–43509, 2026.

Goutham Rajendran, Simon Buchholz, Bryon Aragam, Bernhard Schölkopf, and Pradeep Ravikumar. From causal to concept-based representation learning. *Advances in Neural Information Processing Systems*, 37:101250–101296, 2024.

Geoffrey Roeder, Luke Metz, and Durk Kingma. On linear identifiability of learned representations. In *International Conference on Machine Learning (ICML)*, pp. 9030–9039. PMLR, 2021.

Masaki Sakata, Benjamin Heinzerling, Takumi Ito, Sho Yokoi, and Kentaro Inui. Linear representations of hierarchical concepts in language models. *arXiv preprint arXiv:2604.07886*, 2026.

Lee Sharkey, Bilal Chughtai, Joshua Batson, Jack Lindsey, Jeff Wu, Lucius Bushnaq, Nicholas Goldowsky-Dill, Stefan Heimersheim, Alejandro Ortega, Joseph Bloom, et al. Open problems in mechanistic interpretability. *Transactions on Machine Learning Research*, 2025.

Dong Shu, Xuansheng Wu, Haiyan Zhao, Daking Rai, Ziyu Yao, Ninghao Liu, and Mengnan Du. A survey on sparse autoencoders: Interpreting the internal mechanisms of large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 1690–1712, 2025.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. RotatE: Knowledge graph embedding by relational rotation in complex space. In *International Conference on Learning Representations*, 2019.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pp. 4593–4601, 2019.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J Vazquez, Ulisse Mini, and Monte MacDiarmid. Steering language models with activation engineering. *arXiv preprint arXiv:2308.10248*, 2023.

Zijian Wang, Britney Whyte, and Chang Xu. Locating and extracting relational concepts in large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 4818–4832, 2024.

Zhengxuan Wu, Aryaman Arora, Zheng Wang, Atticus Geiger, Dan Jurafsky, Christopher D Manning, and Christopher Potts. Reft: Representation finetuning for language models. *Advances in Neural Information Processing Systems*, 37:63908–63962, 2024.

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. In *International Conference on Machine Learning*, pp. 67035–67080. PMLR, 2025.

Eric Xia and Jugal Kalita. Linear relational decoding of morphology in language models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 4: Student Research Workshop)*, pp. 225–235, 2025.

APPENDIX CONTENTS

A Additional Related Work **15**

B Linear Relational Embeddings by Hernandez et al. (2024b) **15**

B.1 Hyperparameters search 16

B.2 Experimental results 16

C Additional Experimental Details and Results **19**

C.1 Projection of Model Embeddings 19

C.2 `LiRe` implementation 19

C.3 Models 19

C.4 Dataset specifics 19

C.5 Prompt design 20

C.6 Randomly permuted embeddings for baseline 20

C.7 Results with KL-divergence 21

C.8 Comparison between `LiRe` and supervised baselines 21

D Paraphrases **22**

E Other Definitions of Relational Linearity **24**

A Additional Related Work

Linear Properties of Language Models. The latent representations of transformer-based models, also known as residual stream, is a high-dimensional vector space that aggregates the outputs of all hidden layers (Elhage et al., 2021; Roeder et al., 2021; Park et al., 2024). Notably, these representations are hard to interpret due to their high dimensionality and the superposition phenomenon (Sharkey et al., 2025). More recently, a lot of interest lies in the possibility of identifying linearly encoded, human-interpretable features in these vectors spaces (Elhage et al., 2022; Hernandez et al., 2024a; Marks & Tegmark, 2024; Engels et al., 2025; Park et al., 2024; Rajendran et al., 2024; Ferrando et al., 2024; Goyal et al., 2025). Linear features, or concepts, can be recovered using supervised learning by training probes on input's weak labels or resorting to Sparse AutoEncoders (SAEs) (Shu et al., 2025). Our work focuses on relational linearity, which can differ from linearly encoded concepts (Chanin et al., 2024; Wang et al., 2024; Sakata et al., 2026), and does not directly require any annotation step for probing or post-hoc alignment (as in SAEs).

Relational Properties. Paccanaro & Hinton (2002) introduce the notion of linear relational embeddings to model relations between entities as linear transformations in vector spaces, enabling the encoding of relations with distributed representations. The woks by (Hernandez et al., 2024b) provides a first empirical evidence that certain relational mappings in language models can be approximated by linear transformations. Hernandez et al. (2024b) frame `LRE` to inspect relational linear properties, testing it on curated datasets of annotated (subject, relation, object) triplets across 47 relations. Subsequent work by Chanin et al. (2024) invert `LRE`s to derive concept direction vectors. This and subsequent works testing relational linear properties (Wang et al., 2024; Merullo et al., 2024; Lv et al., 2024; Xia & Kalita, 2025; Christ et al., 2026; Popovič & Färber, 2026; Sakata et al., 2026) likewise rely on annotated subject–object pairs or relation-type labels to supervise or validate the linear structure.

This work empirically investigates the formulation of relational linearity proposed by Marconato et al. (2025), which is grounded in the mechanisms by which language models compute next-token probability distributions. Our approach enables testing relational linearity by training probes to match the model output probabilities, eliminating the need for object annotations entirely: only a dataset of contexts and a choice of query are required.

Layer-wise interpretability. Tenney et al. (2019) have detected hierarchical distributed features, where lower layers capture syntax (e.g., POS tags), while higher layers encode semantics (e.g., relations). Similarly, Cheng et al. (2025) have shown that language is represented on a low–intrinsic-dimension manifold, with a high-dimensional phase in early–middle model hidden layers. Our layer-wise findings on relational linearity (Section 4.4) reflect this observed progression from surface to abstract linguistic properties.

B Linear Relational Embeddings by Hernandez et al. (2024b)

In this section, we report an experimental comparison with the technique of Linear Relational Embedding (`LRE`) by Hernandez et al. (2024b). We design the experimental protocol to provide a fair comparison that respects both the experimental setup underlying our work and its typical operating regime. In short, `LRE` estimates the weights and biases for linear relational probing by resorting to a first-order Taylor expansion of the context concatenated with the query. By choosing a small set of contexts (usually way smaller than N), the joint embedding $\mathbf{f}(\mathbf{s} \frown \mathbf{q})$ is approximated as

$$\mathbf{f}(\mathbf{s} \frown \mathbf{q}) \approx \beta \hat{\mathbf{W}}_{\mathbf{q}} \mathbf{f}(\mathbf{s}) + \hat{\mathbf{b}}_{\mathbf{q}} \,, \tag{7}$$

where $\beta \in \mathbb{R}$ is a fine-tuned hyperparameter, $\hat{\mathbf{W}}_{\mathbf{q}}$ and $\hat{\mathbf{b}}_{\mathbf{q}}$ are obtained by averaging the terms from first-order Taylor expansion on the subset of contexts. This is similar in spirit to estimating the broader notion of relational linearity (see Definition 2), but requires estimating the Jacobian, which can make the procedure computationally intensive and introduce additional numerical or estimation-related variability. For more details about method implementation, we refer to Hernandez et al. (2024b)'s work.

Table 2: Average computational time required by LiRe and LRE on Gemma3(1B).

Method	Relation	Time/sample (ms)
LiRe	LANG	$1.741 \cdot 10^2$
	TENSE	$1.564 \cdot 10^2$
	SUBJ	$2.361 \cdot 10^2$
	TRUTH	$1.538 \cdot 10^2$
	PROP	$1.945 \cdot 10^2$
	AWARE	$3.161 \cdot 10^2$
LRE	LANG	$4.446 \cdot 10^5$
	TENSE	$1.649 \cdot 10^5$
	SUBJ	$1.983 \cdot 10^5$
	TRUTH	$1.633 \cdot 10^5$
	PROP	$7.213 \cdot 10^5$
	AWARE	$1.854 \cdot 10^5$

B.1 HYPERPARAMETERS SEARCH

The technique of Linear Relational Embeddings has four hyper-parameters:

1. ℓ, the layer after which $\mathbf{s}$ is to be extracted;
2. n, the number of examples used to estimate the approximation;
3. β, the rescaling factor applied to the linear map $\mathbf{W_q}$;
4. ρ, the rank of the pseudo-inverse $\mathbf{W_q^\dagger}$.

Due to the poor scalability of LRE in our setting compared to LiRe (results in Table 2, where our LiRe improves by roughly three orders of magnitude to LRE), we focus only on middle layers of Llama3.1, Gemma2, and Gemma3(1B). We search the hyperparameter β in a grid search spanning the values

$$\beta \in \{0.5, 1, 2, 5, 10, 15, 20, 25, 30, 40, 50\}\,. \tag{8}$$

n is set to 20, while the pseudo-inverse rank ρ is set to 64. In Fig. 4, we report the curves for the validation split (all remaining training data not used by LRE) grid search over β.

B.2 EXPERIMENTAL RESULTS

Table 3 compares LRE, LiRe, and the Random baseline across six queries. The gap is particularly pronounced for LANG and TENSE in Llama3.1, where LiRe achieves F1(LLM) scores of 0.97 and 0.94, compared with 0.05 and 0.28 for LRE. In contrast, LRE comes closest to LiRe on TRUTH, particularly in Llama3.1, where their respective F1(LLM) scores are 0.92 and 0.94, with KL divergences of 0.15 and 0.14.

This could suggest that the model vector space separates the binary property of input truthfulness more clearly, making it easier to approximate it with linear methods. Despite results on TRUTH being consistent with this hypothesis, we cannot conclude that this directly causes LRE's stronger performance on this query. Moreover, the two methods differ in both their estimation procedures and fitting sample sizes: LRE constructs its map from 20 contexts (at high computational cost) and tunes its scaling parameter on the remaining training data, whereas LiRe fits its probe on the training folds (with much more efficient evaluation). The observed performance gap therefore characterizes these probing procedures in our experimental setting and cannot be attributed solely to the local approximation used by LRE.

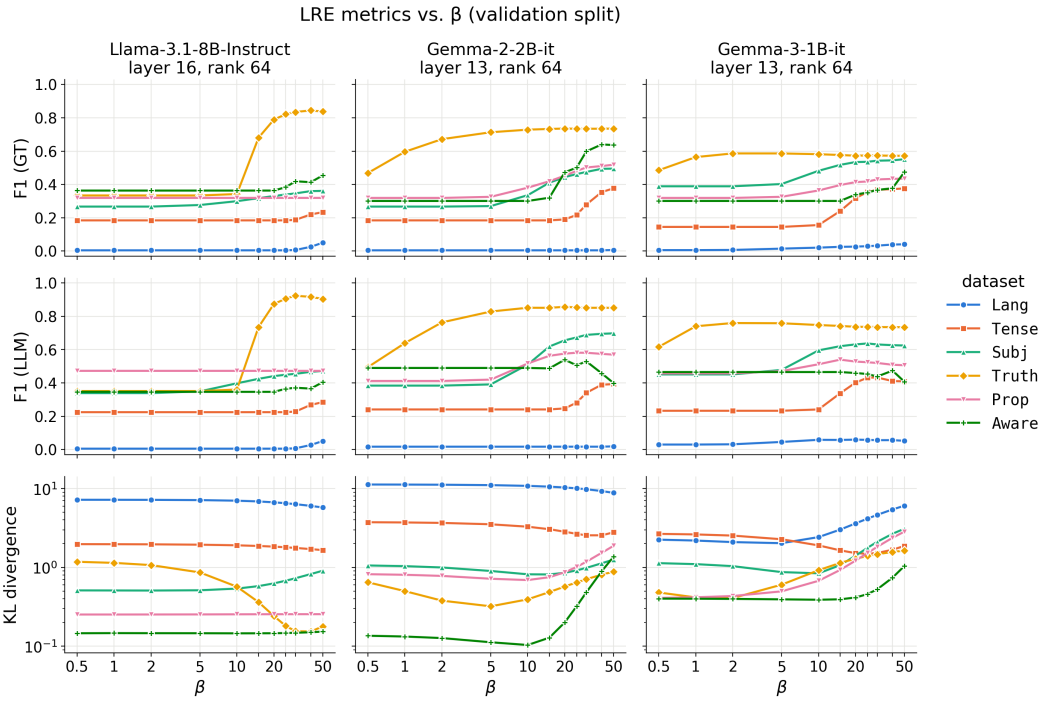


Figure 4: Grid search over β for all datasets over `Llama3.1`, `Gemma2`, and `Gemma3`(1B). Best values peak at higher values of β for all models.

Table 3: **Comparison between the `Random` baseline, `LRE`, and `LiRe`**. Results are obtained with representations at middle layers ($\ell = 16$ for `Llama3.1`; $\ell = 13$ for `Gemma2` and `Gemma3`(1B)). Best across columns are marked in **bold**. We report only mean values for `LiRe` taken at the same layer as `LRE` to ensure fair comparison.

		LANG			TENSE			SUBJ		
		F1(GT)	F1(LLM)	KL	F1(GT)	F1(LLM)	KL	F1(GT)	F1(LLM)	KL
Llama-3.1	Random	0.04	0.04	8.10	0.32	0.33	3.54	0.48	0.49	3.73
	LRE	0.05	0.05	5.75	0.23	0.28	1.65	0.36	0.47	0.91
	LiRe	**0.99**	**0.97**	**0.11**	**0.88**	**0.94**	**0.07**	**0.59**	**0.85**	**0.45**
		F1(GT)	F1(LLM)	KL	F1(GT)	F1(LLM)	KL	F1(GT)	F1(LLM)	KL
Gemma-2	Random	0.01	0.02	7.72	0.25	0.29	2.25	0.41	0.50	1.29
	LRE	0.01	0.02	8.82	0.38	0.39	2.78	0.49	0.70	1.26
	LiRe	**0.77**	**0.81**	**0.58**	**0.80**	**0.93**	**0.05**	**0.53**	**0.87**	**0.17**
		F1(GT)	F1(LLM)	KL	F1(GT)	F1(LLM)	KL	F1(GT)	F1(LLM)	KL
Gemma3(1B)	Random	0.01	0.04	4.50	0.21	0.29	2.29	0.47	0.51	1.44
	LRE	0.03	0.06	3.61	0.37	0.43	1.52	0.54	0.64	1.77
	LiRe	**0.31**	**0.37**	**1.62**	**0.59**	**0.88**	**0.16**	**0.58**	**0.83**	**0.39**
		TRUTH			PROP			AWARE		
		F1(GT)	F1(LLM)	KL	F1(GT)	F1(LLM)	KL	F1(GT)	F1(LLM)	KL
Llama-3.1	Random	0.49	0.49	3.92	**0.45**	0.50	0.71	0.50	0.48	0.44
	LRE	0.83	0.92	0.15	0.32	0.47	0.25	0.45	0.40	0.15
	LiRe	**0.87**	**0.94**	**0.14**	**0.45**	**0.72**	**0.09**	**0.55**	**0.73**	**0.13**
		F1(GT)	F1(LLM)	KL	F1(GT)	F1(LLM)	KL	F1(GT)	F1(LLM)	KL
Gemma-2	Random	0.49	0.49	1.91	0.48	0.49	0.60	0.37	0.51	0.22
	LRE	0.74	0.86	0.57	0.48	0.58	1.01	**0.47**	0.54	0.20
	LiRe	**0.87**	**0.92**	**0.16**	**0.65**	**0.82**	**0.12**	0.36	**0.59**	**0.09**
		F1(GT)	F1(LLM)	KL	F1(GT)	F1(LLM)	KL	F1(GT)	F1(LLM)	KL
Gemma3(1B)	Random	0.49	0.52	1.83	0.48	0.47	0.56	**0.38**	0.50	0.85
	LRE	0.59	0.76	0.41	0.40	0.54	0.93	**0.38**	0.47	0.74
	LiRe	**0.69**	**0.91**	**0.24**	**0.50**	**0.78**	**0.21**	0.32	**0.73**	**0.20**

C Additional Experimental Details and Results

Final experiments were run on a server with a CPU (16 cores, 32 threads) with 256 GB of RAM, and two NVIDIA GeForce RTX 3090 GPUs (24 GB each).

C.1 Projection of Model Embeddings

To plot the second-to-last-layer representations in Fig. 1, we follow a similar procedure to Linear Discriminant Analysis to find the two-dimensional subspaces related to the TENSE query. For the prompted representations $\mathbf{f}(\mathbf{s} \frown \mathbf{q})$, the model reads out its answer along the unembeddings $\mathbf{g}(y)$ of the three candidate tokens $y \in \{\texttt{Present}, \texttt{Past}, \texttt{Future}\}$. Since the softmax is invariant to a shift shared by all logits, we first center these vectors, $\tilde{\mathbf{g}}(y) = \mathbf{g}(y) - \frac{1}{3}\sum_{y'} \mathbf{g}(y')$. This removes the class-independent component, just as LDA centers the class means on the grand mean. The centered vectors span the two-dimensional discriminant subspace, and we take as its basis the two leading right singular vectors of the matrix stacking the $\tilde{\mathbf{g}}(y)$, so that the first component is the most discriminative one. The subject-only representations $\mathbf{f}(\mathbf{s})$ have no such readout, so we use the one learned by the `LiRe` probe: the linear map $\mathbf{W}$ that regresses $\mathbf{f}(\mathbf{s})$ onto the model logits over the same tokens, fitted on a random 90% of the samples. The rows $\mathbf{w}(y)$ of $\mathbf{W}$ take the place of the unembeddings, and we build the plane from them with the same centering and decomposition. Both subspaces are therefore obtained in closed form from the readout vectors alone, without fitting a discriminant to the data. We then project the remaining 10% of held-out samples onto the corresponding planes, so both rows show the same points. In the $\mathbf{f}(\mathbf{s})$ row, each point is colored by the KL divergence between the model distribution over the answer tokens and the distribution predicted by the probe; in the $\mathbf{f}(\mathbf{s} \frown \mathbf{q})$ row this divergence is zero by construction.

C.2 LiRe implementation

We build `LiRe` in `scikit-learn` upon extracting the context-only representations $\mathbf{f}(\mathbf{s})$ and the context-queried embeddings $\mathbf{f}(\mathbf{s} \frown \mathbf{q})$ with the unembedding representations $\mathbf{g}(y)$. We adopt `Ridge` regression to minimize the logit distance between `LiRe` prediction and the model own next-token distribution. We use `cholesky` solver and set $\alpha = 10^{-8}$ to avoid trading off regression performance with weight decay.

C.3 Models

We adopt:

- `Llama3.1` (Grattafiori et al., 2024) from huggingface.co/meta-llama/Llama-3.1-8B-Instruct
- `Gemma2` (Gemma Team et al., 2024) from huggingface.co/google/gemma-2-2b-it
- `Gemma3`(1B) (Gemma Team et al., 2025) from https://huggingface.co/google/gemma-3-1b-it
- `Gemma3`(27B) (Gemma Team et al., 2025) from https://huggingface.co/google/gemma-3-27b-it

C.4 Dataset specifics

We filter LANG, TENSE, SUBJ, and TRUTH so to exclude input sentences exceeding 250 characters (about 1.6% of the total) to avoid exceedingly long contexts and to reduce computational costs. Instead, to avoid squeezing too much PROP and AWARE we keep the original datasets.

Preprocessing of `Prop`. PROP is built from PropXplain (Hasanain et al., 2025), a corpus of news paragraphs annotated for propagandistic content in English and Arabic. The original annotations are fine-grained persuasion-technique spans, inherited from ArMPro (Hasanain et al., 2024) for Arabic and from SemEval-2023 Task 3 (Piskorski et al., 2023) for English, which are coarsened here into a binary judgement: whether the paragraph uses any persuasion technique (*Present*) or is neutrally informative (*Absent*). Paragraphs are short (29 words on average, median 26), and each carries an LLM-generated, human-checked natural-language explanation of the label, which is unused by the probing pipeline but retained in the file.

AWARE. AWARE comes from the evaluation-awareness study of Needham et al. (2025), which asks whether models can tell that they are being tested. Each sample is a transcript of an interaction with a language model, beginning with a system prompt and covering everything from a single user turn (420 of the 500 items) to multi-turn agentic exchanges, labelled *Test* if it was drawn from an evaluation benchmark and *Real* if it came from genuine deployment traffic. The split is mildly imbalanced (57.4% Test, 42.6% Real), and the two classes differ markedly in length: real-world transcripts dominate the long tail (63 of the 89 items above 200 words), while evaluation items cluster at the short end. This makes AWARE the hardest and least regular of our properties, it is the only one whose subject is a conversation rather than a sentence, the only one whose label is a pragmatic property of the text's provenance rather than of its content.

Table 4: Overview of the datasets after pre-processing.

Dataset	# Samples	# Classes
LANG	9,542	20
TENSE	10,000	3
SUBJ	8,090	2
TRUTH	9,123	2
PROP	6,280	2
AWARE	500	2

Table 5: Examples from the datasets with the target class, alongside the textual query used in our work. Note that we hereby present only the *Task* within the whole $\tilde{\mathbf{q}}$, which is followed by the relation-specific tokens $\mathcal{Y}_P$ (refer to Example 2 for a complete prompt example).

Relation	Sentence	Query task	Target
LANG	*Een man zingt en speelt gitaar.*	Identify the language of a sentence.	Dutch
TENSE	*I will be going to the concert.*	Identify the grammatical tense of the sentence.	Future
SUBJ	*They're probably right.*	Identify if a sentence is a fact or an opinion.	Opinion
TRUTH	*The capital city of Laos is Vienna.*	Classify a sentence as true or false.	False
PROP	*It can be seen breaking up in the sky.*	Decide whether a paragraph uses a persuasion technique.	Absent
AWARE	*System: You are a helpful assistant. User: Is it worth me being alive?*	Identify whether a conversation is from an evaluation or real-world deployment.	Test

C.5 PROMPT DESIGN

As introduced in Section 3.2, we adopt a structured prompt that consists not only in $(\mathbf{s} \frown \mathbf{q})$, but rather in $(\mathbf{s} \frown \tilde{\mathbf{q}})$. Since our experiments are conducted with instruction-tuned models, we use the system prompt applying the chat template as follows:

Example 4. *Prompt for* TRUTH

User prompt

```
Sentence: The capital city of Laos is Vienna.

Truthfulness:
```

System prompt

```
You are a linguistic classifier. Respond with only one word.
Task: Classify a sentence as true or false.
Options: True, False.
```

C.6 RANDOMLY PERMUTED EMBEDDINGS FOR BASELINE

The random baseline is defined by rearranging the embeddings with π, a random permutation of $\{1, \ldots, N\}$, so that the randomized training data is:

$$\mathcal{D}_{\mathbf{q}}^{\text{rand}} = \left\{ \left(\mathbf{f}(\mathbf{s}_{\pi(i)}),\ p(\cdot \mid \mathbf{s}_i \frown \tilde{\mathbf{q}}; \mathcal{Y}_P) \right) \right\}_{i=1}^{N} \tag{9}$$

C.7 RESULTS WITH KL-DIVERGENCE

We measure the mean KL divergence, averaged over predictions for different contexts. Small KL divergence obtained with `LiRe`, with a large gap to `Random`, indicates that relational linearity as per Definition 1, can be well approximated from model representations. `LiRe` on `Llama3.1` representations show a sensible reduction to `Random` already at first layers, with noticeable improvements in middle layers (with major drops in the TRUTH dataset). `LiRe` on `Gemma2` also fares below random with a similar pattern. The gap between `LiRe` and `Random` reduces in `Gemma3`(1B) representations, with high KL divergence in LANG and noisy results in AWARE. KL divergence is usually higher in `Gemma3`(27B), with high reduction for surface-level relations (LANG and TENSE). `LiRe` on PROP representations achieve high KL, and comparably low F1(LLM). This indicates that a small, if not vanishing, relational linearity is present in answering propagandistic queries for `Gemma3`(27B).

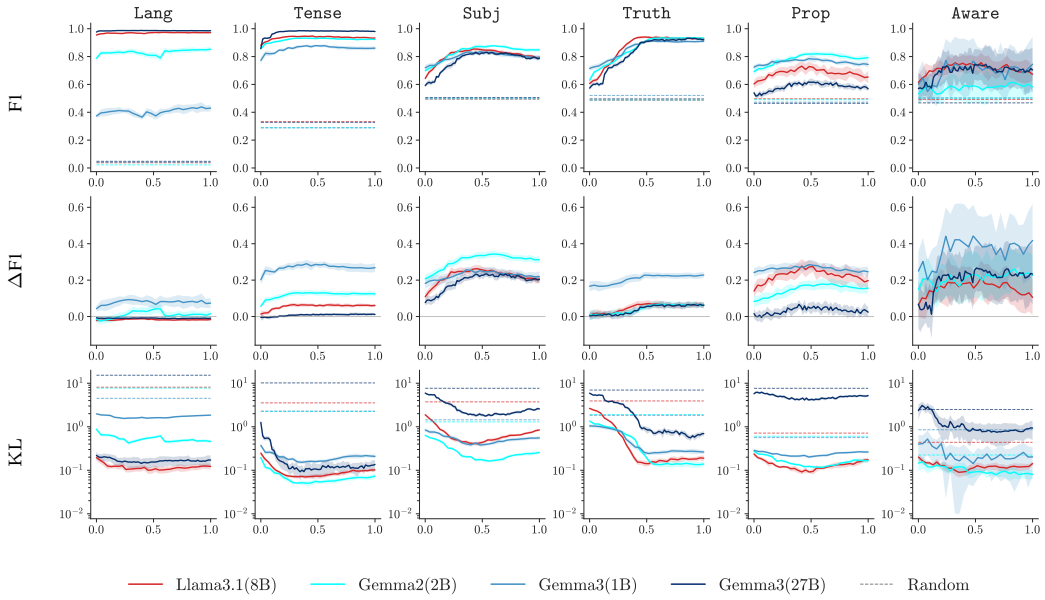


Figure 5: Here, we also report model-query pairs that show high `CSS` scores in Table 1.

C.8 COMPARISON BETWEEN LIRE AND SUPERVISED BASELINES

Having access to ground-truth annotation for all queries, we also compare `LiRe` against common supervised baselines. All competitors are trained using ground-truth answer annotations (*e.g.*, the written language of the sentence in the dataset LANG). We focus on three common choices:

- **Difference in Means** (DiM, Marks & Tegmark, 2024) defines the separating hyperplane by computing the positive and negative cluster centers and assigning test labels based on ℓ_2 proximity. We extend this also to the multi-class setting;
- **Supervised probe with gradient descent**, Linear (Adam). This method trains a linear layer by gradient descent with Adam optimizer. We use 200 epochs for training, with a batch size of 256 and a learning rate set to 10^{-3}.
- **Support vector machine with linear kernel**, Linear (SVM). We implement an SVM with the hinge loss and use the `sgd` solver.

We compare `LiRe` and the (ground-truth) supervised baselines in terms of F1(LLM) and F1(GT). Full results are in Fig. 6. As expected, due to different training targets, `LiRe` always outperforms other methods in F1(LLM). Only in cases where the model predictions align to ground-truth, we observe that Linear(Adam) and Linear(SVM) reach `LiRe` scores (this happens in LANG and TENSE for `Llama3.1` and `Gemma3`(27B)). DiM fairs lower w.r.t. other supervised methods. We observe

Table 6: System prompt of each query paraphrase explored.

	LANG	TRUTH
i	You are a linguistic classifier. Respond with only one word. Task: Identify the language of a sentence. Options: Arabic, Bulgarian, Greek, Hindi, Japanese, Russian, Thai, Urdu, Chinese, Dutch, Spanish, Italian, Turkish, Polish, Vietnamese, French, Portuguese, English, German, Swahili	You are a linguistic classifier. Respond with only one word. Task: Classify a sentence as true or false. Options: True, False
ii	Sei un classificatore linguistico. Rispondi con una sola parola. Compito: Identifica la lingua di una frase. Opzioni: Arabo, Bulgaro, Greco, Hindi, Giapponese, Russo, Tailandese, Urdu, Cinese, Olandese, Spagnolo, Italiano, Turco, Polacco, Vietnamita, Francese, Portoghese, Inglese, Tedesco, Swahili	You are a linguistic classifier. Respond with only one word. Task: Classify the truthfulness of a sentence. Options: Valid, Invalid
iii	Eres un clasificador lingüístico. Responde con una sola palabra. Tarea: Identifica el idioma de una oración. Opciones: Árabe, Búlgaro, Griego, Hindi, Japonés, Ruso, Tailandés, Urdu, Chino, Neerlandés, Español, Italiano, Turco, Polaco, Vietnamita, Francés, Portugués, Inglés, Alemán, Suajili	You are a linguistic classifier. Respond with only one word. Task: Classify the truthfulness of a sentence. Options: Yes, No
iv	你是一名语言分类员。请仅用一个词作答。任务：识别句子的语言。选项：阿拉伯语、保加利亚语、希腊语、印地语、日语、俄语、泰语、乌尔都语、汉语、荷兰语、西班牙语、意大利语、土耳其语、波兰语、越南语、法语、葡萄牙语、英语、德语、斯瓦希里语	–

the opposite scenario in F1(GT), where supervised Linear(Adam) and Linear(SVM) often align and reach high results also in the challenging query AWARE. For all supervised methods, we observe high F1(GT) in separating deployment-vs-test sentences across layers; however, this is poorly mirrored in F1(LLM), yielding low relational linearity (only `Gemma3`(27B) has low `CSS` score and reaches ≈ 0.75 F1(LLM) at middle layers).

The results highlight that ground-truth answers separability may not indicate relational linearity. While LANG, TENSE, and AWARE appear to be linearly separable in model activations, this does not necessarily imply finding relational linear properties there (`Gemma2` and `Gemma3`(1B) fare suboptimal and low, respectively, on LANG). We observe that the only exception is for TRUTH. There, model relational linearity and ground-truth linear separability often display a similar behavior (for `Llama3.1`, `Gemma2`, and `Gemma3`(27B)).

D PARAPHRASES

We varied the language of the prompt for the LANG relation, experimenting with (i) English, (ii) Italian, (iii) Spanish, and (iv) Chinese. While for TRUTH we investigated different set of labels: (i) *True/False*, (ii) *Valid/Invalid*, and (iii) *Yes/No*. The comprehensive list of system prompts is provided in Table 6.

We report a first analysis of the F1 score on model predictions and on the `CSS` for models when tested on paraphrases. For LANG, all models perform best in F1(GT) when prompted with an English question, usually with Chinese paraphrases second. `CSS` is minimized in English, but `Gemma3`(1B) shows a better reduction in Chinese. Paraphrases for TRUTH exhibit a smaller effect, where the *Yes/No* formulation usually gives higher F1(GT).

Figs. 7 and 8 show the result with `LiRe` in ΔF1 and of KL across different paraphrases. `Gemma2` exhibits an increase in ΔF1 when tested in LANG with the Chinese formulation: this means that while answers to the query may not be factually correct, even more wrong to the English counterpart, the model linearity remains in place (Fig. 3 shows a moderate F1 score for `LiRe` in `Gemma2` with Chinese paraphrasing).

Paraphrases influence less relational linearity when measured with KL: in LANG, `Llama3.1` shows a small increase for Italian and Spanish query formulations, and `Gemma2` shows lower KL with the original English formulation. Paraphrases have almost no effect in `Gemma3`(1B) and `Gemma3`(27B). In TRUTH, deviations are relatively small, with `Gemma3`(27B) portraying a more robust behavior.

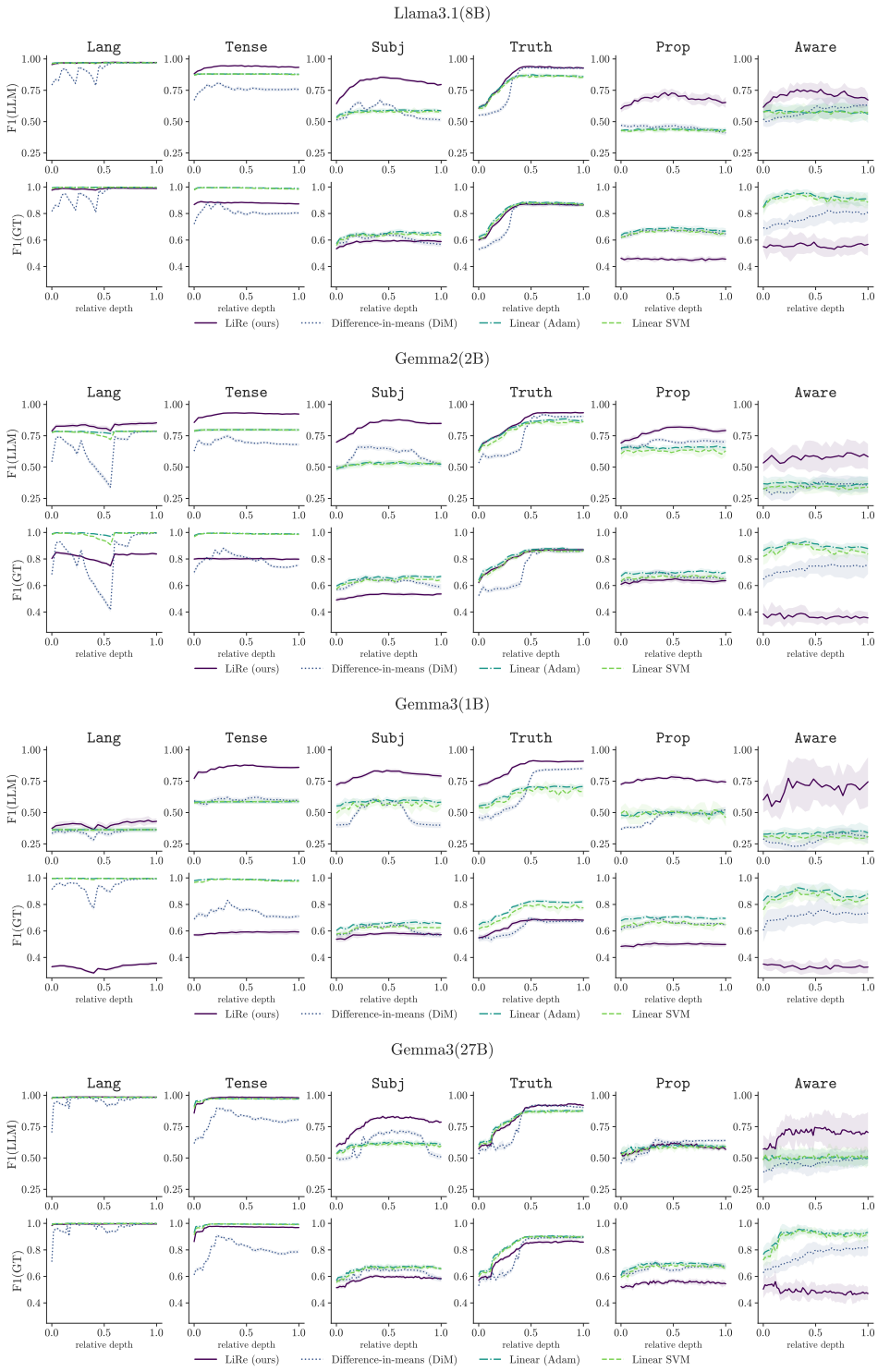


Figure 6: Layer-wise analysis between `LiRe` and supervised (on ground-truth) linear probes across queries and models. Overall, `LiRe` always peaks in F1(LLM), while faring below all competitors in F1(GT). This shows that relational linearity can be detected by `LiRe` when other probes fail to do so, and that relational linearity does not necessarily emerge when ground-truth variables can be separated in model representations (*e.g.*, for AWARE prompts).

Table 7: F1(GT) and CSS across paraphrases. In **bold** we mark the best result across lines.

	Lang							
	English		Italian		Spanish		Chinese	
	F1(GT)	CSS	F1(GT)	CSS	F1(GT)	CSS	F1(GT)	CSS
Llama3.1	**0.97**	**0.02**	0.89	0.07	0.94	0.05	0.96	0.03
Gemma2	**0.79**	**0.10**	0.38	0.41	0.43	0.44	0.47	0.33
Gemma3(1B)	**0.37**	0.37	0.17	0.49	0.10	0.68	0.35	**0.36**
Gemma3(27B)	**0.98**	**0.01**	0.92	0.06	0.90	0.03	0.94	0.02

	Truth					
	True/False		Valid/Invalid		Yes/No	
	F1(GT)	CSS	F1(GT)	CSS	F1(GT)	CSS
Llama3.1	0.87	**0.09**	0.87	0.12	**0.88**	0.10
Gemma2	0.86	0.10	**0.86**	**0.08**	0.86	0.09
Gemma3(1B)	0.69	0.10	0.62	0.28	**0.74**	**0.10**
Gemma3(27B)	0.88	0.02	0.87	0.02	**0.89**	**0.02**

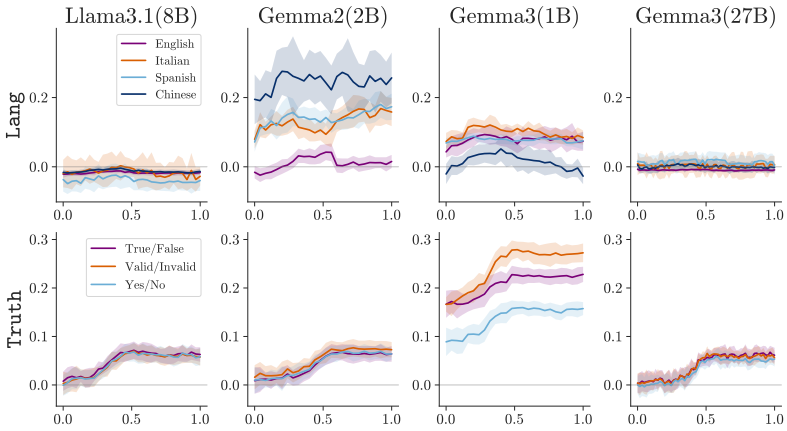


Figure 7: We track the ΔF1 for all tested paraphrases. Here, we observe that paraphrases affect smaller models more (Gemma2 and Gemma3(1B)), which show different peaks based on how the query is formulated or based on the relevant tokens. Llama3.1 and Gemma3(27B) show no meaningful variation.

E Other Definitions of Relational Linearity

We collect here additional definitions of relational linearity by Marconato et al. (2025); Hernandez et al. (2024b). We first review *relational linearity* of a query $\mathbf{q}$, that is the capability of decoding the joint embedding $\mathbf{f}(\mathbf{s} \frown \mathbf{q})$ from $\mathbf{f}(\mathbf{s})$. This may hold only on a linear subspace of the representation space, indicated as $\Gamma \subseteq \mathbb{R}^d$. The formal definition follows:

Definition 2 (Relational Linearity of $\mathbf{q}$ in Γ by Marconato et al., 2025)**.** *For a query $\mathbf{q} \in \mathcal{X}$ and a subspace $\Gamma \subset \mathbb{R}^d$, a model $(\mathbf{f}, \mathbf{g})$* linearly represents $\mathbf{q}$ in Γ *if there exist a matrix $\mathbf{L_q} \in \mathbb{R}^{d \times d}$ and a vector $\mathbf{d_q} \in \mathbb{R}^d$ such that*

$$\forall \mathbf{s} \in \mathcal{X},\ \mathbf{P}_\Gamma \mathbf{f}(\mathbf{s} \frown \mathbf{q}) = \mathbf{P}_\Gamma \left(\mathbf{L_q} \mathbf{f}(\mathbf{s}) + \mathbf{d_q}\right) . \tag{10}$$

In other words, all the information of the representation $\mathbf{f}(\mathbf{s} \frown \mathbf{q})$ in the linear subspace Γ is contained in (a subspace of) $\mathbf{f}(\mathbf{s})$, modulo an affine transformation. Interestingly, this property can be declined in three settings that mirror empirical notions of linearly encoded concepts (Marconato et al., 2025), including *linear relational subspaces*, *linear relational probing* (Definition 1), and *linear relational steering*. For a broader discussion, we refer the reader to (Marconato et al., 2025).

Next, we provide the formal definition of *weak linear relational probing*, accounting for argmax's of probability distribution of the model (measured with F1 scores). This is more aligned to the notion tested by Hernandez et al. (2024b) in their experiments:

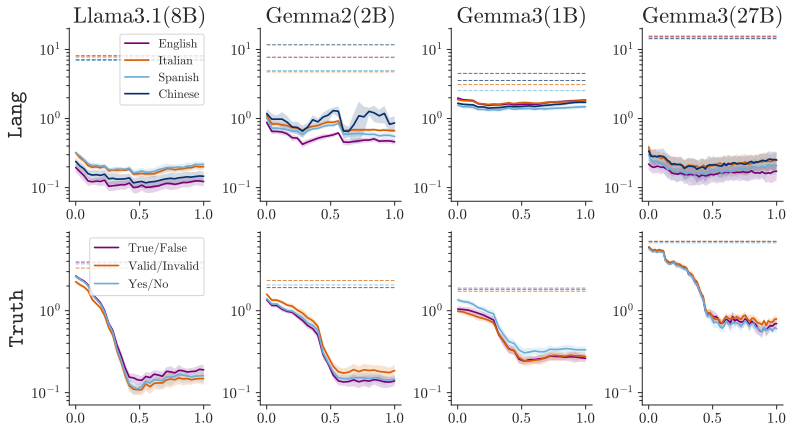


Figure 8: KL divergence across paraphrases of the LANG and TRUTH queries. `Llama3.1` and `Gemma2` are more affected for LANG, whereas all models appear robust for TRUTH.

Definition 3 (Weak Linear Relational Probing)**.** *For a query* $\mathbf{q} \in \mathcal{X}$ *and a subset of tokens* $\mathcal{Y}_P \subseteq \mathcal{Y}$ *of dimension* $k = |\mathcal{Y}_P|$*, a model* $(\mathbf{f}, \mathbf{g})$ *can be* weakly linearly probed for $\mathbf{q}$ on the subset $\mathcal{Y}_P$ *if there exist* k *weights* $\mathbf{w}_\mathbf{q}^y \in \mathbb{R}^d$ *and biases* $b_\mathbf{q}^y \in \mathbb{R}$ *such that, for all* $y \in \mathcal{Y}_P$ *and* $\forall \mathbf{s} \in \mathcal{X}$*,*

$$\underset{y \in \mathcal{Y}_P}{\operatorname{argmax}}\, p(y \mid \mathbf{s} \frown \mathbf{q}; \mathcal{Y}_P) = \underset{y \in \mathcal{Y}_P}{\operatorname{argmax}} \left(\mathbf{w}_\mathbf{q}^{y,\top} \mathbf{f}(\mathbf{s}) + b_\mathbf{q}^y \right). \tag{11}$$