PREDICTING REASONING ERRORS BEFORE THEY OCCUR: A PROMPT-SIDE ACTIVATION DIRECTION ANTICIPATES CORRECTNESS IN A LARGE LANGUAGE MODEL

Anonymous authors
Paper under double-blind review

ABSTRACT

Responses produced by large language models are eloquent in a way that makes right answers indistinguishable from wrong ones. Predicting the correctness of the model's response is critical, as it helps either stop the model from answering or route it to a larger model instead of producing eloquent but incorrect answers. In this study, we use activation patching to identify a candidate layer for arithmetic reasoning. We use the hidden-state activations at the candidate layer to develop a difference-in-means measure. We test the approach across three models: Mistral-7B, Llama-2-13B, and Qwen 27B. The measure predicts subsequent correctness across all three models (AUC .627, .673, and .786 from smallest to largest model), and the signal emerges around the middle of the network and strengthens toward the task-relevant layers. The learned direction transfers to new formulations and an independent benchmark without retraining (AUC .810 and .864 on the largest model), although its decision threshold shifts across datasets and requires recalibration. These results place correctness information earlier in the computation, before the model commits to an answer, suggesting that reliability can be assessed before generation rather than only diagnosed after it. The principal methodological contribution of this paper is to utilize the activations of causally identified candidate layers and develop a light-weight difference-in-means measure to predict the correctness of models' response before the generation of answer tokens.

1 INTRODUCTION

Large Language Models can be a source of misinformation if the model provides wrong answers as eloquently as the right ones. This is especially important because the usage is not limited to experts but also to laymen. It is essential to identify whether the model is going to generate an accurate answer or not. Two strands of literature have attempted to address this question. The first strand of literature can be classified into post-hoc approaches (Kadavath et al., 2022; Jiang et al., 2021; Lin et al., 2022; Xiong et al., 2024; Wang et al., 2023; Kuhn et al., 2023; Farquhar et al., 2024). These responses, though promising, require generating a full response, increasing inference cost. The latter strand of literature assesses response accuracy by examining the model's internal state(Belinkov, 2022; Azaria & Mitchell, 2023; Burns et al., 2023; Marks & Tegmark, 2024; Orgad et al., 2025). This approach shows that models internally encode their own errors, and the accurate and inaccurate internal state representations are linearly separable.

Thus, the existing literature suggests that model representations contain information about the correctness of the eventual response. Although prior work has probed for this signal pre-generation, especially difference-in-means directions and linear probes these approaches typically search across all layers, training or aggregating a direction per layer and selecting whichever performs best, without committing to a specific layer.

In this study, we question whether the activations in the causally identified layer can be used as a predictor of the correctness. We test the hypothesis across on aritmetic reasoning. We causally localize task-based computation using layer-wise activation patching (Vig et al., 2020; Meng et al.,

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.

2022; Zhang & Nanda, 2024). Then, at the identified layer, we estimate a single linear direction by contrasting prompt-side activations for correct and incorrect responses and use the difference-of-means estimator, as used in the representation steering literature (Marks & Tegmark, 2024; Zou et al., 2023; Rimsky et al., 2024).

We find that a light-weight difference-in-means estimator developed based on activations at the causally identified layer predicts model correctness and we test it across the three different models belonging to different model families. Across three models from different families, the direction predicts whether the subsequent response will be correct, reaching an AUC of 0.783 in the largest model. Although it is read before any answer token is generated, it matches or exceeds post-generation confidence measures and performs comparably to trained linear probes, and it transfers without refitting to an independent benchmark.These results indicate that the activations at the candidate layer act as a direction to predict the correctness of models' response before the generation of answer tokens. The results have both theoretical and practical implications. Theoretically, activations at the causally identified candidate layer could be utilised to predict the correctness of the models' response. Practically, identifying appropriate causal layer reduces the training costs in predicting the correctness which is further useful in either terminating the response or in routing.

2 Related Work

2.1 Self-Knowledge and Internal Representation of Truth

Extant literature had demonstrated that language models can generate useful estimates regarding the likelihood of their predictions being accurate. As Kadavath et al. (2022) demonstrated, models are able to be trained or prompted to make predictions about the accuracy of their predictions by self-evaluation, which shows that information regarding the correctness of their predictions is available prior to the completion of the prediction itself. Nonetheless this information comes from an explicit confidence measure produced by the model rather than an informational task-specific signal. In contrast, we explore a different approach and seek to extract an informational, task-specific signal from the activations of the layer that performs computation, identified via activation patching.

A second line of work asks whether information about truth and correctness is encoded in the internal representations of language models. Azaria & Mitchell (2023) find that classifiers trained on hidden activations can distinguish true from false statements, while Burns et al. (2023) recover a truth-related direction in activation space with no labelled data. These works show that correctness-related information can be extracted from relatively simple representations learned from model activations, albeit generally by probing or searching for predictive representations across layers, without first establishing what layer is involved in the relevant task computation. In this study, we use activation patching to causally identify a task-relevant candidate layer, then use activations at that layer to extract a lightweight signal of subsequent correctness as a task-specific quantity. This gives the model a signal of whether it is likely to produce a correct answer before generating it.

In the above-mentioned literature, Marks & Tegmark (2024) is the work most closely related to our study. They demonstrate that, provided enough scale, language models encode the truth or falsehood of a given proposition in a linearly oriented way, and that a relatively simple difference-of-means method recovers this direction as well as more complex alternatives. Our approach is similar in principle: it derives a direction from a difference-of-means on contrastive activations, but differs in both the target and the timing. Marks & Tegmark (2024) decode the truthfulness of a proposition the model has already been given; our study instead decodes, from the final prompt position and before the model produces any reasoning, whether a multi-step computation it is about to perform will succeed.

2.2 Causal Localisation via Activation Patching

A separate methodological tradition establishes internal computation causally rather than correlationally, by intervening on activations and observing the effect on model behaviour. Vig et al. (2020) introduce causal mediation analysis to language models, using targeted activation substitutions to identify which components mediate a given behavior rather than merely correlate with it. Meng et al. (2022) apply this logic to factual recall, localizing the storage of individual facts to a narrow band of middle-layer feed-forward modules and showing that a single rank-one edit

at the identified site is sufficient to alter the stored association. The activation patching procedure used here to localise arithmetic computation is a direct descendant of this approach, applied at the granularity of whole decoder layers rather than individual components, and to a different behaviour; single-step arithmetic recovery rather than factual recall. This distinction between correlational and causal localization matters for our design: probe accuracy identifies where a signal is decodable, whereas activation patching tests whether intervening on a representation changes the computation itself. We rely on patching for this reason, to avoid selecting a layer by decodability alone.

2.3 Steering and Contrastive Directions

A third line of work uses mean-difference directions not to detect a property but to control it, by adding a computed direction to a model's activations during inference. Rimsky et al. (2024) construct steering vectors from the difference in residual-stream activations between contrastive positive and negative examples of a target behavior and show that adding this vector during generation reliably alters model outputs. Li et al. (2023) similarly identify a small number of attention heads whose activations, shifted along a truth-related direction found via linear probing, substantially improve truthfulness on held-out benchmarks. Our estimator for the correctness direction is constructed in the same way, a normalised difference of means between contrastive activation sets, but we use it purely as a predictive readout: we project onto the direction and measure downstream correctness, without adding the direction back into the forward pass. Whether the direction identified here is causally implicated in producing correct answers, in the sense established for these steering methods, or is instead a correlate of some property the model has already resolved, is a question our design does not answer; we regard Rimsky et al. (2024)- and Li et al. (2023)-style intervention on this direction as the most direct test of that distinction, and note it as a natural extension of the present work.

3 Data and Methodology

3.1 Models and Datasets

We evaluated three open-weight autoregressive transformers on grade-school arithmetic word problems: Qwen3.6-27B, Llama-2-13B-chat, and Mistral-7B-Instruct-v0.3. All models were loaded using 4-bit NF4 quantization (Dettmers et al., 2023). We tested across three different model families across three different model sizes. Because every activation used in the analysis was obtained from a quantized model, the absolute magnitudes of the projection scores reported below are specific to this configuration and are not expected to transfer numerically to a full-precision instantiation. The same four-stage method was applied to each model, but several implementation parameters, the number of patching pairs, the construction of the labelling sample, the calibration/evaluation split, and the layer windows for the derived measures were set per model. These are summarized in Table 1 and described in the relevant subsections below.

The primary dataset was GSM8K (Cobbe et al., 2021), which pairs grade-school mathematical word problems with natural-language solutions. Two additional datasets, Socratic GSM8K (Cobbe et al., 2021) and SVAMP (Patel et al., 2021), were used to evaluate whether the signal transferred to other formulations of grade-school arithmetic.

Construction of the labeling sample. The procedure for assembling the labeled analysis sample differed between the largest model and the two smaller ones, because their natural error rates differed by more than an order of magnitude, and the method requires a sufficient number of incorrect responses to estimate the correctness direction. Qwen3.6-27B answered GSM8K correctly roughly 96% of the time, so a random draw would have yielded too few incorrect examples to estimate the direction reliably. We therefore enriched its sample for errors: we generated responses from GSM8K until 160 incorrect responses had been collected, retained all of them, and randomly subsampled an equal-sized set of correct responses, yielding a balanced analysis sample of 480 examples, divided into calibration and evaluation halves of 240 each while preserving the correct-to-incorrect ratio. Llama-2-13B and Mistral-7B, by contrast, answered incorrectly on a large fraction of problems (64.7% and 55.0% on a random draw), so no enrichment was needed: for each we randomly sampled 1,000 problems (without replacement, seed 42) from the combined GSM8K train–test pool and divided them by a stratified 70/30 split into 700 calibration and 300 evaluation examples.

Table 1: Per-model configuration. Layer windows for the average-projection and trajectory-slope measures were set from each model's own candidate layer and separation profile, not carried over as fixed layer numbers.

	Qwen-27B	Llama-2-13B	Mistral-7B
Decoder layers	64	40	32
Patching minimal pairs	56	56	56
Candidate layer	61	37	29
Pilot accuracy (50 problems)	96%	42%	42%
Labeling protocol	enriched	random	random
Labeling sample size	480	1,000	1,000
Observed error rate	3.94%	64.7%	55.0%
Calibration / evaluation	240 / 240	700 / 300	700 / 300
Avg-projection window	44–62	30–38	22–30
Trajectory-slope window	32–62	20–38	16–30

Pilot validation. Before collecting each main sample, we evaluated the model on the first 50 GSM8K test problems to confirm that generation and answer extraction were functioning correctly. Qwen3.6-27B answered 96% correctly with one unparseable response; Llama-2-13B and Mistral-7B each answered 42% correctly with no unparseable responses (for Llama-2-13B, after correcting an answer-extraction pattern that initially missed negative dollar amounts). In every case the pilot error rate was consistent with that observed in the larger data collection.

3.2 Overview of the Method

The analysis proceeded in four stages. First, we identified where the arithmetic computation was represented, using layer-wise activation patching. Second, at the resulting candidate layer, we extracted each model's representation of the problem before generation and estimated a direction separating prompts that subsequently received correct and incorrect responses. Third, we followed this direction across network depth and compared its predictive performance against linear probes, token-level confidence, and simple measures of problem difficulty. Finally, we applied the direction and analysis procedure to held-out reformulations and independent arithmetic datasets without retraining.

The distinction between the first two stages is important. Activation patching is an intervention and was used to *locate* the computation; the correctness direction is a difference of observed means and was used to *measure* information about the eventual outcome. The direction was therefore evaluated as a predictive signal, not as a causal intervention;a point we return to when interpreting the results.

3.3 Locating Arithmetic Computation with Activation Patching

To identify the layers involved in arithmetic computation without selecting them, we used a clean–corrupted minimal-pair design. We constructed synthetic addition problems of the form:

> "There are a red balls and b blue balls in a box. How many balls are there in total?"

For each pair, a and b were sampled uniformly from $[2, 50]$. The corrupted version replaced a with a different integer a', leaving the rest of the prompt unchanged, so that the clean and corrupted prompts shared structure, length, and syntax and differed only in the quantity entering the arithmetic. Table 1 shows the number of pairs used per model.

For each pair, we recorded the log probability assigned to the correct answer under the clean prompt, $\mathrm{LP}_{\mathrm{clean}}$, and under the corrupted prompt, $\mathrm{LP}_{\mathrm{corrupt}}$. We then patched the clean activation into the corrupted computation, separately at each layer L: we replaced the residual-stream activation at the final prompt-token position in the corrupted forward pass with the activation from the same position in the clean forward pass, leaving all other positions and layers unchanged, and re-scored the corrupted prompt to obtain $\mathrm{LP}_{\mathrm{patched}}(L)$. Each layer was patched independently, so an intervention at one layer did not affect the measurement at any other.

We quantified recovery at layer L as

$$R(L) = \frac{\mathrm{LP}_{\mathrm{patched}}(L) - \mathrm{LP}_{\mathrm{corrupt}}}{\mathrm{LP}_{\mathrm{clean}} - \mathrm{LP}_{\mathrm{corrupt}}}. \tag{1}$$

A value of zero indicates no recovery relative to the corrupted prompt, and a value of one indicates full recovery of the clean-prompt log probability. Pairs for which the clean–corrupted difference was smaller than 10^{-6} were excluded. We averaged recovery across pairs and designated the layer with the highest mean recovery as the candidate layer for that model.

This procedure localized the computation at the level of whole transformer layers; individual attention heads and sub-components were not separately analyzed.

3.4 Response Labeling and Prompt-Side Activation Extraction

For each GSM8K example in the analysis sample, we generated a response with greedy decoding under a three-shot chat template and assigned a binary correctness label based on whether the extracted final answer matched the reference answer.

The activation used to predict this label was obtained *without* the generated response. Each problem was passed through the model in a single forward pass, with hidden-state outputs enabled, using the same three-shot chat template used for generation, and we extracted the residual-stream activation at the final prompt token from the candidate layer. Let this activation be $mathbfa_i\ in\mathbb{R}^d$ for problem i. Under causal attention, the final prompt token can attend to the entire preceding problem statement, so $\mathbf{a}_i$ summarizes the model's representation of the problem after it has read the complete question. No answer tokens were present when $\mathbf{a}_i$ was extracted.

This is one of the central features of the design: the model has processed the problem, but has not yet generated the response whose correctness is being predicted.

3.5 Estimating the Correctness Direction

We divided the analysis sample into calibration and evaluation sets, keeping the correct-to-incorrect ratio fixed across both (proportions in Table 1). All parameters of the predictive measure were estimated on the calibration set alone.

Let $\bar{\mathbf{a}}_+$ and $\bar{\mathbf{a}}_-$ denote the mean calibration activations for prompts that subsequently received correct and incorrect responses, respectively. We defined the correctness direction as follows:

$$\mathbf{d} = \frac{\bar{\mathbf{a}}_+ - \bar{\mathbf{a}}_-}{\|\bar{\mathbf{a}}_+ - \bar{\mathbf{a}}_-\|}. \tag{2}$$

For a new prompt activation $\mathbf{a}$, the scalar correctness score was

$$s(\mathbf{a}) = \mathbf{a}^\top \mathbf{d}, \tag{3}$$

with higher values indicating greater alignment with the direction associated with correct responses in the calibration data. This difference-of-means estimator follows approaches used to identify interpretable directions in activation space (Zou et al., 2023; Rimsky et al., 2024; Marks & Tegmark, 2024). The direction was estimated from prompt-side activations alone; the generated response contributed only the correctness label used to define the two class means, not any part of the input from which the score was computed.

3.6 Layerwise Analysis of the Correctness Signal

After estimating $\mathbf{d}$ at the candidate layer, we examined whether the same direction was expressed elsewhere in the network. We extracted residual-stream activations from two sets of layers: a dense window of 13 layers centered on the candidate layer (truncated at the network boundaries), and a coarser set sampling every fourth layer across the network.

The direction was *not* re-estimated at each layer. The single direction obtained at the candidate layer on the calibration set was held fixed, and activations from other layers were projected onto it. This is motivated by the residual-stream architecture, in which layers' writes additively into a

shared representational space, so projecting different layers' activations onto one common direction is meaningful and lets us track how information aligned with that direction accumulates with depth, rather than fitting a separate classifier at each layer.

At each layer we computed the mean projection separately for correct and incorrect responses and examined their separation, from these trajectories we derived three scalar measures:

- **Single-layer projection:** the projection onto $\mathbf{d}$ at the candidate layer.
- **Average projection:** the mean projection across a window of late layers over which class separation widened.
- **Trajectory slope:** the ordinary least-squares slope of projection against layer index across a broader late-network window.

The two window ranges were set per model from its own separation profile (Table 1); because these ranges were chosen by inspecting the analysis sample, they are post hoc hyperparameters, and the out-of-distribution evaluations provide the primary unbiased assessment of these two measures.

3.7 Comparison Baselines

We compared the activation-based score against three families of alternatives.

Linear probes. We trained ℓ_2-regularized logistic-regression classifiers to predict correctness from raw residual-stream activations, both per layer under five-fold cross-validation and on concatenated activations from the later layers under a held-out split. These provide a comparison against a discriminatively trained predictor operating on the same representations.

Post-generation confidence. We computed two response-level measures after generation: mean token log probability and mean predictive entropy over the generated tokens (entropy negated so that larger values indicate greater confidence). Unlike the activation score, both require the response to have already been generated.

3.8 Out-of-Distribution Evaluation

Finally, we tested whether the learned direction transferred beyond the examples used to estimate it. The correctness direction, candidate-layer choice, layer windows, and in-distribution decision threshold were all derived from the GSM8K analysis data and applied without refitting to Socratic GSM8K (Cobbe et al., 2021) and SVAMP (Patel et al., 2021). The same prompt construction, response generation, correctness labeling, and activation extraction procedures were used for these datasets.

The primary transfer metric was AUC, which measures whether the score ranks correct responses above incorrect ones without reference to any threshold, because a direction can preserve its ranking ability even if the numerical boundary between correct and incorrect examples shifts across datasets. This let us test two distinct forms of generalization: whether the direction still carries information about correctness, and whether the same numerical score can serve directly as a decision rule on new data.

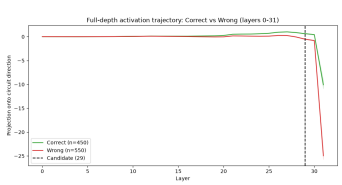
4 Results

4.1 Localizing the Computation

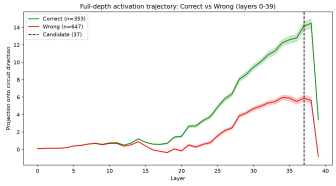
In all three models, activation patching identified a candidate layer in the late region of the network for the arithmetic task: layer 61 of 64 (Qwen), layer 37 of 40 (Llama), and layer 29 of 32 (Mistral). In Qwen3.6-27B, patching the final-position residual stream produced no measurable recovery through layer 50, rose to a plateau of approximately 0.13 across layers 51–59, and peaked at layer 61 with a recovery of 0.24. Similar observations are made for other models.

As shown in Figure 1, the projection of activations from every layer onto the candidate layer has a consistent depth profile across all three models. The direction is therefore not specific to single

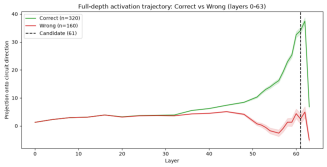
model family. Although the strength differs across the models, with strength being highest for the largest model Qwen-27B and least for the smallest model Mistral-2-7B, the separation between the right and wrong answers is negligible in the early layers and grows monotonically toward the candidate layer. In the next subsection, we test whether this simple linear measure predicts the model correctness.



(a) Mistral-7B (32 layers, layer 29).



(b) Llama-2-13B (40 layers, layer 37).



(c) Qwen3.6-27B (64 layers, layer 61).

Figure 1: Mean projection onto the fixed correctness direction as a function of layer, for prompts subsequently answered correctly and incorrectly (red). The direction is estimated once at each model's candidate layer (dashed line) and held fixed across all layers.

4.2 Prompt-Side Activations Predict Correctness Before Generation

Table 2: Discrimination (AUC) of activation-based measures and baselines, in-distribution and out-of-distribution, across three models

Model	Measure	GSM8K (test)	Socratic (Near-OOD)	SVAMP (Far-OOD)
Mistral	Linear probe (best layer, layer 16)	0.683	0.736	0.720
	Single-layer projection (layer 29)	0.621	0.705	0.724
	Average projection	0.627	0.704	0.725
	Trajectory slope	0.608	0.704	0.719
	Negative mean entropy	0.639	0.677	0.683
	Mean token log-probability	0.643	0.666	0.702
Llama	Linear probe (best layer, layer 18)	0.693	0.771	0.612
	Single-layer projection (layer 37)	0.673	0.690	0.571
	Average projection (layers 30–38)	0.672	0.688	0.568
	Trajectory slope (layers 20–38)	0.658	0.671	0.564
	Negative mean entropy	0.638	0.640	0.561
	Mean token log-probability	0.597	0.593	0.546
Qwen	Linear probe (Best Layer, Layer 50)	0.751	0.863	0.901
	Single-layer projection (layer 61)	0.783	0.809	0.869
	Average projection (layers 44–62)	0.786	0.810	0.864
	Trajectory slope (layers 32–62)	0.781	0.806	0.861
	Negative mean entropy	0.756	0.774	0.835
	Mean token log-probability	0.742	0.765	0.825

Projecting across the candidate layer direction, successfully predicted if subsequent response will be correct for all three models (Table 2). We saw the held-out AUCs of 0.783 for Qwen3.6-27B, 0.673 for Llama-2-13B, and 0.621 for Mistral-7B (approximate 95% CIs [0.726, 0.840], [0.607, 0.739], and [0.557, 0.685]; Hanley & McNeil, 1982).Here, when we averaged the projection over window of layers or took the slomodels'ss depth, AUC changed by maximum of 0.015.Hence the single candidate layer captures signal about models' correctness.

This signal is available before the model generates answer tokens, whereas the post-generation methods need full response. Despite the method relying on the pre-generation activations the measure matches or exceeds the post-generation confidence across the three models. In Qwen3.6-27B the AUC of simple lintoken'ssure developed is 0.783 while it is 0.756 for negative entropy (CI [0.695, 0.817]) and 0.742 for token's log probability. In the case of Llama-2-13B, it is 0.673 against 0.638 and 0.597. However, Mistral-7B is an exception here (0.621 against 0.639 and 0.643). Intervals overlap heavily here and thus we read these results as parity instead of superiority. When adding the log probability to the prompt-side score in Qwen3.6-27B, AUC remains unchanged (0.787 versus

0.786). This indicates that the post-generation measure provides no extra information beyond what is available before generation. Moreover, we performed linear probing (a linear classifier trained on internal activations at the best layer) to compare performance with the developed measure. While the linear probing accuracy is slightly higher in the smaller models Mistral and Llama, in the larger model Qwen the measure outperforms linear probing. However, on out-of-distribution datasets, although the measure outperforms negative mean entropy and the mean token log probability, linear probing has a slightly better AUC than the developed measure. However, it is critical to note that we use only the activations at a single light-weight layer, and the measure is light-weight compared to light-weight probing. Thus, a simple light-weight measure developed based on the activations at the candidate layer predicts the model's correctness. These findings have both theoretical and practical implications. Theoretically, causal methods identify activations at the candidate layer that help predict the model's correctness. Practically, a simple light-weight measure can serve as an alternative approach to predict the model's correctness, which significantly helps with either model termination or routing to a better model.

5 Discussion

The principal finding of this study is that a model's success on an arithmetic word problem can be predicted before it generates any answer token, using a single linear direction read from the causally identified candidate layer. Previous approaches obtain this information in one of two ways:

- Post-hoc methods estimate correctness from the generated response or from repeated sampling (Kadavath et al., 2022; Kuhn et al., 2023; Farquhar et al., 2024), therefore, requiring generation first.
- Internal-state methods such as linear probes can decode activation correctness (Azaria & Mitchell, 2023; Marks & Tegmark, 2024; Orgad et al., 2025), but they typically fit a readout at every layer and then select the one that performs best.

However, in this approach the layer is fixed by activation patching, without reference to correctness labels, and the direction is read before generation begins. The computational cost to obtain the signal is negligible; activation is computed in the prompt's forward pass during generation, so scoring a prompt adds one dot product.

The direction also compared favourably with post-generation confidence, which is available only after the full response has been produced. Across the three models and three evaluation sets, it exceeded both token-level entropy and log probability in eight of nine comparisons, including every out-of-distribution comparison; the exception was Mistral-7B on GSM8K (0.621 versus 0.639 and 0.643). In distribution, the confidence intervals overlap, so these results indicate parity rather than a reliable advantage. Parity is nonetheless the relevant result in practice: the direction provides this information before the first token is generated, when abstaining or deferring can still save computation.

Trainable linear probes did not reliably outperform the direction, even though the direction requires no discriminative training. In distribution, the probes exceeded it by 0.062 AUC in Mistral-7B and 0.020 in Llama-2-13B, within the approximate intervals and with the probe layer selected for its performance, while in Qwen3.6-27B the direction exceeded the probe (0.783 versus 0.738; the probe used a different split, so this comparison is indicative). Out of distribution, the direction matched the probe on SVAMP in Mistral-7B, whereas the probes scored higher on Socratic GSM8K and, in Llama-2-13B, on SVAMP.

What exactly this direction encodes is still an open question. It might reflect some model-specific sense of uncertainty, directionality, or perceived difficulty, rather than capturing something strictly about arithmetic. Also, our current estimator is strictly correlational. The extension of this study can test whether it participates directly in computation or just reflects expected outcome, which requires an intervention along direction. This is similar to recent steering work (Rimsky et al., 2024; Li et al., 2023).

Overall discrimination was highest with Qwen3.6-27B. But these three models differ widely in family, training setup, and error rates (roughly 4% compared to 55–65%). Their analysis samples were constructed differently too. Therefore this ordering shouldn't be interpreted as scaling trend.

6 LIMITATIONS

The limitations of the study can broadly be divided into three categories: Model families, task families, causality vs prediction. Although the study shows the performance of the developed measure across the three model families at different model sizes. The study does not test performance differences within the model family based on different model sizes. The findings are limited to only one task family i.e., mathematical reasoning. The measure developed discusses only the correlation and not the causation. Each of these limitations can further be tested and extended in the future work.

7 CONCLUSION

The current study tests whether the measure developed based on the activations at the causally identified layer can predict the model correctness. The results indicate a such an approach acts as a predictor. At the candidate layer identified by activation patching, a single difference-of-means direction, computed before any answer token is generated, predicts whether the answer will be correct in all three models. It exceeds post-generation confidence in eight of nine comparisons, performs comparably to trained linear probes without discriminative training, and transfers to an independent benchmark in two of three models. The signal emerges in the middle of the network, before the candidate layer. These findings have three implications: first, reliability assessment need not wait for generation: part of it can be performed before the first token, at negligible computational cost; second, activations at the causally identified layer can be used as a predictor of model correctness, this study explores one of the methods; finally, the light-weight measure can be used in routing or terminating the model from generating the answer when the model likely generates wrong answer.

AI USE STATEMENT

In this work, we used generative AI tools for checking overlaps with previous works, searching and summarizing prior literature and understanding implementations of the same. We have not used generative AI tools for generating experimental results or figures. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

To enable reproducibility we detail models, datasets, quantization settings and sample construction in Section 3.1, and provide a summary of the settings used for each model in Table 1. We describe our methods for activation patching, label responses, prompt-side activation extraction, and estimation of the correctness direction and other measures in Sections 3.3-3.6. Finally, in Sections 3.7–3.8 we present the comparison baselines and our procedure for performance evaluation on out-of-distribution datasets.

AUTHOR CONTRIBUTIONS

ACKNOWLEDGMENTS

REFERENCES

Amos Azaria and Tom Mitchell. The internal state of an LLM knows when it's lying. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 967–976, 2023.

Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219, 2022.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. Discovering latent knowledge in language models without supervision. In *Proceedings of the 11th International Conference on Learning Representations*, 2023.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. QLoRA: Efficient finetuning of quantized LLMs. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630, 2024.

James A. Hanley and Barbara J. McNeil. The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology*, 143(1):29–36, 1982.

Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. How can we know when language models know? on the calibration of language models for question answering. *Transactions of the Association for Computational Linguistics*, 9:962–977, 2021.

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, and Jared Kaplan. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *Proceedings of the 11th International Conference on Learning Representations*, 2023.

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

Stephanie Lin, Jacob Hilton, and Owain Evans. Teaching models to express their uncertainty in words. *Transactions on Machine Learning Research*, 2022.

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

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Hadas Orgad, Michael Toker, Zorik Gekhman, Roi Reichart, Idan Szpektor, Hadas Kotek, and Yonatan Belinkov. LLMs know more than they show: On the intrinsic representation of LLM hallucinations. In *Proceedings of the 13th International Conference on Learning Representations*, 2025.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. Are NLP models really able to solve simple math word problems? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 2080–2094, 2021.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Matt Turner. Steering Llama 2 via contrastive activation addition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, 2024.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and Stuart Shieber. Investigating gender bias in language models using causal mediation analysis. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *Proceedings of the 11th International Conference on Learning Representations*, 2023.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. Can LLMs express their uncertainty? an empirical evaluation of confidence elicitation in LLMs. In *Proceedings of the 12th International Conference on Learning Representations*, 2024.

Fred Zhang and Neel Nanda. Towards best practices of activation patching in language models: Metrics and methods. In *Proceedings of the 12th International Conference on Learning Representations*, 2024.

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