Simple Factuality Probes Detect Hallucinations in Long-Form Natural Language Generation

Jiatong Han[1] **Neil Band**[2] **Muhammed Razzak**[1] **Jannik Kossen**[1]
Tim G. J. Rudner[3] **Yarin Gal**[1]

[1]University of Oxford [2]Stanford University [3]University of Toronto

julius.han@outlook.com nband@stanford.edu
{muhammed.razzak,jannik.kossen,yarin.gal}@cs.ox.ac.uk tim.rudner@utoronto.ca

Abstract

Large language models (LLMs) often mislead users with confident hallucinations. Current approaches to detect hallucination require many samples from the LLM generator, which is computationally infeasible as frontier model sizes and generation lengths continue to grow. We present a remarkably simple baseline for detecting hallucinations in long-form LLM generations, with performance comparable to expensive multi-sample approaches while drawing only a single sample from the LLM generator. Our key finding is that LLM hidden states are highly predictive of factuality in long-form natural language generation and that this information can be efficiently extracted at inference time using a lightweight probe. We benchmark a variety of long-form hallucination detection methods across open-weight models up to 405B parameters and demonstrate that our approach achieves competitive performance with up to 100x fewer FLOPs. Furthermore, our probes generalize to out-of-distribution model outputs, evaluated using hidden states of smaller open-source models. Our results demonstrate the promise of hidden state probes in detecting long-form LLM hallucinations.

1 Introduction

Modern large language model (LLMs) applications increasingly rely on frontier models producing tens of thousands of tokens: users generate codebase-scale edits with specialized coding assistants (OpenAI, 2025a), create multi-page technical reports with advanced models like GPT-5 (OpenAI, 2025c), or employ reasoning models such as GPT-5-Thinking (OpenAI, 2025d) and Gemini-2.5-Pro (DeepMind, 2025) as proof assistants. While these models generally produce accurate content, they remain susceptible to hallucinations, particularly as context lengths and conversation turns increase.

For users to safely leverage these frontier capabilities while mitigating potential harms, models could accompany their generations with fine-grained *factuality scores*, enabling users to identify which portions of the output are trustworthy and which require verification. Detecting hallucinations in long-form generations, however, presents unique challenges compared to short-form settings. Unlike evaluating the likelihood of a single short phrase being false, long-form texts contain numerous interdependent claims spanning multiple paragraphs or documents.

Current approaches to long-form hallucination detection rely primarily on sampling-based methods, which assess claim confidence through semantic entropy (Kuhn et al., 2023; Farquhar et al., 2024), self-consistency (Band et al., 2024), or graph uncertainty (Jiang et al., 2024). While effective for paragraph-length outputs, these approaches become computationally prohibitive for frontier models with hundreds of billions of parameters generating tens of thousands of tokens, requiring multiple generation calls to increasingly expensive reasoning models.

A promising alternative lies in leveraging an LLM's internal representations of factuality and confidence. Recent work has demonstrated that LLM activations—specifically, their per-token hidden states—encode information about answer correctness in short-form settings (Azaria and Mitchell, 2023; Burns et al., 2023). However, no prior research has systematically investigated whether this information could be extracted for long-form hallucination detection, to assign fine-grained confidence scores to interdependent claims in extended generations.

We address this crucial gap by exploring whether hidden states capture hallucination signals in long-form generations and whether lightweight probes trained on these states can enable efficient, claim-level hallucination detection. Specifically, we train linear and tree-based probes to map an LLM's hidden state representation of an atomic claim directly

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 16209–16226
November 4-9, 2025 ©2025 Association for Computational Linguistics

to its likelihood of hallucination. These **Factuality Probes** are inexpensive to train, interpretable by design, and enable a pipeline for detecting long-form hallucinations with just a single generation call to the LLM—dramatically reducing computational overhead compared to sampling-based methods.

Our empirical evaluations demonstrate that our approach achieves comparable hallucination detection performance (measured by AUROC) to established sampling-based baselines while reducing computational resources *by two orders of magnitude*. Moreover, we show robust scalability across models from 3B to 405B parameters, with detection performance scaling log-linearly in model size. Remarkably, probes trained on smaller open models retain strong performance when assessing outputs from significantly larger, proprietary models, despite the substantial disparity in model sizes. To summarize, our key contributions are as follows:

1. We introduce linear probes applied to hidden states as a computationally efficient method for hallucination detection in long-form generations, matching the performance of sampling-based approaches while reducing computational cost by over $100\times$.
2. We empirically establish that hidden state probes exhibit consistent, log-linear improvements in detection performance as model sizes increase from 3B to 405B parameters.
3. We show that probes trained on smaller open-weight models generalize reliably across diverse domains and to detect hallucinations in outputs from larger, proprietary models.

Code to reproduce our results can be found at: https://github.com/JThh/fact-probe.

2 Fine-Grained Hallucination Detection with Factuality Probes

Frontier LLMs produce tens of thousands of tokens per output in applications such as editing codebases, writing research reports, and answering technical questions (OpenAI, 2025a,b,c,d). To provide users with a fine-grained understanding of the reliability of such long-form generations, we aim to accurately report claim-level scores of factuality. Unlike prior approaches drawing many samples from the generator LLM, we seek an efficient approach that scales to modern frontier models.

The core empirical finding that enables our approach is that LLM hidden states encode rich representations of claim-level factuality. Leveraging this, we propose an approach to report fine-grained factuality scores at inference time by applying efficient probes on LLM hidden states. We refer to these probes as **Factuality Probes**.

Representing fine-grained confidence scores in long-form generations. Our high-level procedure for providing fine-grained factuality scores at inference time is shown in Figure 1. First, the generator LLM generates a single long-form generation. Next, we apply an (often smaller) auxiliary LM to decompose the long-form generation into atomic claims, and to associate each claim to a token span in the long-form generation. Then, we can obtain estimates of the factuality of each atomic claim using a Factuality Probe, and attribute this factuality to the corresponding span, providing users with a visual representation of confidence as in the green and red paragraph in Figure 1.

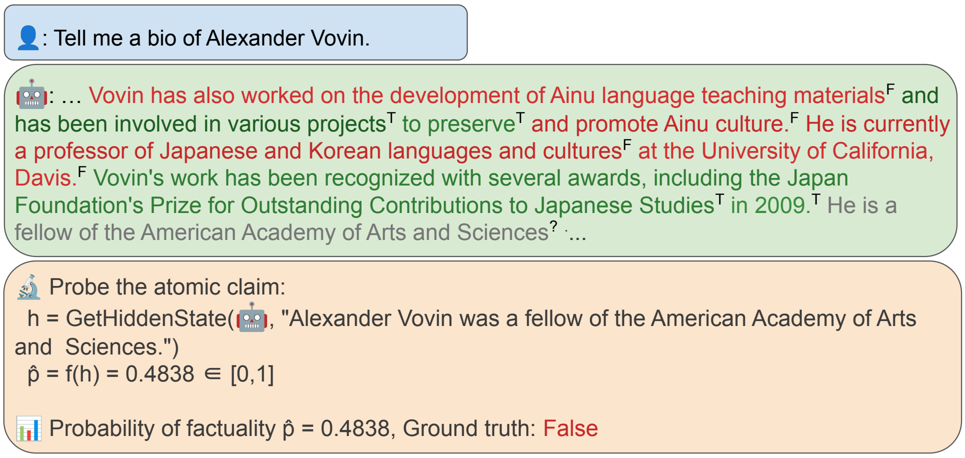
Estimating claim-level confidence with Factuality Probes. We obtain claim-level confidence estimates by passing atomic claims to either the generator LLM or a smaller LM, extracting hidden states, and applying a trained probe on them to predict the probability of claim-level factuality. In either case, Factuality Probes are significantly cheaper than existing sampling-based approaches for long-form hallucination detection, requiring only a single `.generate()` call to the generator LLM in the entire procedure.

In order to train hidden state Factuality Probes, we use retrieval as a scalable source of reliable factuality scores for atomic claims. We next describe our training and inference algorithms in detail.

2.1 Training Factuality Probes

Factuality Probes are lightweight classifiers that take as input the hidden states of an LM, and decide whether a single, self-contained statement—an atomic claim—is true. We train these probes in two stages (see Algorithm 1).

Stage 1: generate a supervised factuality dataset. Given a pool of prompts $\mathcal{D}_{\text{prompt}}$, we first query a generator LM π_{gen} to produce a long-form output z for each prompt. Because long-form outputs often interleave many facts, we next decompose z into individual claims with the help of a prompted auxiliary LM, π_{aux}. For example, if the long-form generation is on the history of the United States, an atomic claim could be: "George Washington was the first president of the United States." We



👤: Tell me a bio of Alexander Vovin.

🤖: … Vovin has also worked on the development of Ainu language teaching materials^{F} and has been involved in various projects^{T} to preserve^{T} and promote Ainu culture.^{F} He is currently a professor of Japanese and Korean languages and cultures^{F} at the University of California, Davis.^{F} Vovin's work has been recognized with several awards, including the Japan Foundation's Prize for Outstanding Contributions to Japanese Studies^{T} in 2009.^{T} He is a fellow of the American Academy of Arts and Sciences$^{?}$ ...

🔬 Probe the atomic claim:
h = GetHiddenState(🤖, "Alexander Vovin was a fellow of the American Academy of Arts and Sciences.")
p̂ = f(h) = 0.4838 ∈ [0,1]

📊 Probability of factuality p̂ = 0.4838, Ground truth: False

Figure 1: **Factuality Probes enable the fine-grained reporting of confidence in long-form generations.** Given a long-form generation from a frontier LLM, we decompose it into atomic claims, obtain hidden states for each claim using either the frontier LLM or a smaller LM, and apply a lightweight probe on hidden states to estimate claim-level factuality. By attributing atomic claims to token spans, we propagate claim-level factuality back to the long-form generation, providing users with a natural visualization of factuality throughout a long-form generation.

pass each extracted claim c into an encoder LM, for which we use either the generator LLM π_{gen} or a cheaper, smaller LM π_{small} to obtain hidden states h_c. The impact of different choices for the encoder and hidden state selection are analyzed in Section 4.3. Lastly, we obtain a binary label y_c indicating whether the claim is correct (in this case, $y_c = 1$). The label comes from a strong retrieval-augmented verifier f_{ret} which we treat as an oracle: $y_c = f_{\text{ret}}(c) \in \{0, 1\}$. f_{ret} refers to the retrieval-based evaluators of the benchmarks. They verify the factuality of claims using either document retrieval from Wikipedia (Min et al., 2023) or through a Web Search API (Wei et al., 2024). Collecting all (h_c, y_c) pairs yields a training set $\mathcal{D}_{\text{probe}}$ of paired hidden states and indicators of factuality.

Stage 2: fitting the Factuality Probe. We then train a simple classifier f that maps a hidden vector to a probability of factuality: $p(y = \text{true}|c) = f(h_c; \theta)$. Throughout the paper we consider two simple instantiations of f:

1. **Sparse logistic regression.** A linear probe with an L_1 penalty encourages the model to rely on a small subset of dimensions,
$$\min_{\theta} \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} \mathcal{L}\big(y_c, f(h_c; \theta)\big) + \lambda \|\theta\|_1; \quad (1)$$
$\mathcal{L}$ is the logistic loss and λ controls sparsity.
2. **XGBoost.** A more powerful gradient-boosted decision tree that can capture nonlinear interactions among hidden state coordinates.

Altogether, this two-stage procedure yields a probe that is cheap to evaluate yet highly accurate in flagging hallucinated statements generated by LLMs.

2.2 Representing Fine-Grained Factuality at Inference Time

At inference time, we use a trained Factuality Probe to provide a user with fine-grained estimates of factuality in a long-form generation, as in Figure 1. Overall, inference proceeds similarly to the training algorithm, with the key difference being our use of span attribution to propagate claim-level factuality scores back to a contiguous span of tokens in the long-form generation. The inference time algorithm is detailed in Algorithm 2, and is composed of five steps.

1. **Answer generation.** As in standard decoding, the generator LLM π_{gen} receives a prompt x and produces a long-form completion z.
2. **Claim decomposition with span attribution.** The auxiliary model π_{aux} decomposes z into a set of pairs $\{(c, S(c))\}$, where c is an atomic claim and $S(c) = [s(c), e(c)]$ denotes the contiguous token span in z that makes the claim. Spans are enforced to be non-overlapping; when two claims would otherwise share a token, the token is deterministically assigned to the first claim detected. This simple tie-breaking rule yields clean and interpretable attributions in practice (cf. Figure 1).

Algorithm 1: Training Long-Form Factuality Probes

```
Given: Generator LM $\pi_{\text{gen}}$; optional encoder LM
       $\pi_{\text{small}}$; auxiliary LM $\pi_{\text{aux}}$;
       retrieval-augmented factuality scorer $f_{\text{ret}}$;
       input prompts $\mathcal{D}_{\text{prompt}} = \{x_i\}_{i=1}^{N}$.
Result: A trained factuality probe $f$:
        $h_c \mapsto f(h_c) \in [0, 1]$.
/* Stage 1: Generate training dataset    */
$\mathcal{D}_{\text{probe}} \leftarrow \{\}$
for $x \in \mathcal{D}_{\text{prompt}}$ do
    /* Generate completion                */
    $z \sim \pi_{\text{gen}}(z \mid x)$
    /* Extract atomic claims              */
    $\mathcal{C} \sim \pi_{\text{aux}}(\cdot \mid \text{DecomposePrompt}(z))$
    for $c \in \mathcal{C}$ do
        /* Choose LM for encoding hidden
           states based on budget         */
        $\pi_{\text{enc}} \leftarrow \pi_{\text{gen}}$ or $\pi_{\text{small}}$
        /* Get hidden representation      */
        $h_c = \text{GetHiddenState}(\pi_{\text{enc}}, c)$
        /* Get factuality label           */
        $y_c = f_{\text{ret}}(c) \in \{0, 1\}$
        /* Add to training data           */
        $\mathcal{D}_{\text{probe}} \leftarrow \mathcal{D}_{\text{probe}} \cup (h_c, y_c)$
    end
end
/* Stage 2: Train probe                   */
$f \leftarrow$ supervised training on $\mathcal{D}_{\text{probe}}$, e.g., minimize
  logistic regression loss in Eq. 1.
return $f$
```

3. **Hidden state extraction.** As at training time, we obtain the hidden state h_c for atomic claim c by encoding c with the generator LLM π_{gen} or a smaller LM π_{small}.
4. **Probe evaluation.** The factuality probe f maps each hidden vector to a probability of factuality $\widehat{p}_c = f(h_c) \in [0, 1]$. Altogether, we obtain a set of claims and corresponding factuality scores $\mathcal{R} = \{(c, \widehat{p}_c) \mid c \in \mathcal{C}\}$.
5. **Fine-grained factuality visualization.** We use the spans $S(c) = [s(c), e(c)]$ for visualizing claim-level factuality. For each claim, we propagate its probe score $\widehat{p}_c$ back to the supporting span $S(c) = [s(c), e(c)]$ in the completion z. For example, in Figure 1, tokens in $S(c)$ are heatmapped from red-to-green according to $\widehat{p}_c$, producing an interpretable overlay that highlights which segments of the output are judged more or less reliable.

Altogether, Factuality Probes enable an end-to-end pipeline providing claim-level factuality annotations. In the following sections, we investigate their accuracy and computational overhead.

Algorithm 2: Inference-Time Scoring of Long-Form Factuality

```
Given: Generator LM $\pi_{\text{gen}}$; optional encoder LM
       $\pi_{\text{small}}$; auxiliary LM $\pi_{\text{aux}}$; trained factuality
       probe $f$; prompt $x$.
Result: Set of claims with factuality scores
        $\mathcal{R} = \{(c, \widehat{p}_c)\}$ and supporting spans $\mathcal{S}$.
/* Generate completion                     */
$z \sim \pi_{\text{gen}}(z \mid x)$
/* Extract claims & supporting spans       */
$(\mathcal{C}, \mathcal{S}) \sim \pi_{\text{aux}}(\cdot \mid \text{DecomposePrompt}(z))$
            $\triangleright\ \mathcal{S} = \{S(c) = [s(c), e(c)] \mid c \in \mathcal{C}\}$
$\mathcal{R} \leftarrow \{\}$
for $c \in \mathcal{C}$ do
    /* Choose LM for encoding hidden states
       based on budget                     */
    $\pi_{\text{enc}} \leftarrow \pi_{\text{gen}}$ or $\pi_{\text{small}}$
    /* Get hidden representation           */
    $h_c = \text{GetHiddenState}(\pi_{\text{enc}}, c)$
    /* Predict factuality                  */
    $\widehat{p}_c = f(h_c) \in [0, 1]$
    /* Add to results                      */
    $\mathcal{R} \leftarrow \mathcal{R} \cup (c, \widehat{p}_c)$
end
return $\mathcal{R}$
```

3 Experiment Setup

Let k denote the sample size for sampling-based methods. We define the model producing generations as π_{gen} and the auxiliary model used for claim breakdown, filtering, and possibly entailment assessment as π_{aux}. For evaluation, we sample a minimum of 25 topics from the LongFact objects dataset (Wei et al., 2024) for training our factuality probes and 30 entities from the Wikipedia-based dataset used by Min et al. (2023) for testing all methods. Detailed sample statistics are provided in Table 5. Generation lengths for all models were capped at 512 tokens per entity or object.

3.1 Baselines

We compare Factuality Probes with established baselines for LLM confidence estimation, including sampling– and verbalization-based methods.

Semantic Entropy (SE). Farquhar et al. (2024) generate long-form text using greedy decoding with π_{gen}, decompose into atomic claims, and assess uncertainty by generating stochastic questions asking about each claim using π_{aux}. For each question, we collect k high temperature responses from π_{gen}, compute SE of the response distributions, and average over the questions to estimate the uncertainty of the given claim.

P(**True**). We perform claim decomposition and then prompt π_{aux} to generate questions that would

Table 1: Fact probe in-domain and out-of-domain performance improves with larger training models and generalizes across generator LMs (π_{gen}). $\pi_{\textbf{gen}}$ **(f)** denotes the generator LM on whose activations FP was trained.

In-Domain	
$\pi_{\textbf{gen}}$ (f)	**AUROC**
Llama 3.2 3B	$0.7261_{\pm 0.0125}$
Llama 3.1 8B	$0.7357_{\pm 0.0113}$
Llama 3.1 70B	$0.7453_{\pm 0.0097}$
Llama 3.1 405B	$0.7579_{\pm 0.0082}$
OOD (π_{gen} = GPT-4o-Mini)	
$\pi_{\textbf{gen}}$ (f)	**AUROC**
Llama 3.2 3B	$0.6248_{\pm 0.0135}$
Llama 3.1 8B	$0.6531_{\pm 0.0142}$
Llama 3.1 70B	$0.6905_{\pm 0.0161}$
Llama 3.1 405B	$0.7076_{\pm 0.0114}$

yield each claim as an answer. π_{gen} produces multiple alternative answers to these questions, which are incorporated into a few-shot prompt. The model then evaluates its original claim against these brainstormed alternatives to determine factuality.

SelfCheckGPT. Following Manakul et al. (2023), we decompose π_{gen} generations into atomic claims with π_{aux} and prompt π_{gen} to assess each claim's correctness in the context of its previously generated text. This self-checking approach allows models to verify their own factual assertions without requiring external knowledge sources.

Graph-based Uncertainty. Following Jiang et al. (2024), we sample k high-temperature responses plus one greedily decoded response from π_{gen}, decompose each into atomic claims, and merge claims with equivalent meanings into one node, as evaluated by π_{aux}. We construct a bipartite graph between responses and claims, and quantify uncertainty using graph centrality metrics. Specifically, Self-Consistency (SC) can be considered as a variant of Jiang et al. (2024) which uses degree centrality for uncertainty measurement.

Verbalized Confidence. Following Tian et al. (2023a), we obtain Post-hoc Verbalized Confidence (PH-VC) by directly asking π_{gen} the likelihood of its own claim being true. As shown by Jiang et al. (2024), SC and VC can be combined (i.e., SC+VC) to outperform each individual algorithm.

4 Experiment

We evaluate factuality prediction methods on several π_{gen} including the Llama 3 series models (Llama3.2-3B, Llama3.1-8B, Llama3.1-70B, Llama3.1-405B) and Gemma2-9B, and utilize GPT-4o-mini as π_{aux} throughout. We choose Logistic Regression (LR) and XGBoost as the candidate classifiers f in our fact probing method. To test method robustness under data domain shift, we train probes on LongFact labels and evaluate them on FActScore labels. Additionally, we test robustness to out-of-distribution (OOD) models by applying probes trained on the activations of π_{small} on the generations of π_{gen} or π_{aux}, e.g., closed-source model generations (Min et al., 2023).

We train probes on the model's hidden states to predict fact labels, categorizing them as either supported or not supported. The inputs are the hidden representations of a single token—either the first token (FT), last token (LT), or second-last token (SLT) of a single hidden layer—or the concatenated hidden representations of 5-layer groups. All claims, regardless of topic, are pooled together, and the probe is trained with 3-fold stratified cross-validation. Overall, the statistics for training and testing data are presented in Table 5.

4.1 Probes Capture Long-form Factuality

In our experiments, we demonstrate that probing hidden representations effectively captures signals predictive of model factuality. Our results reveal that probing the final tokens of atomic claims yields superior performance compared to probing first or penultimate token positions. Additionally, our experiments show that using XGBoost as f marginally outperforms LR classifiers, while implementing layer grouping shows no significant improvement in probe performance. The implication is that claim token hidden representations can reliably predict factuality through either linear mappings or boosted tree models. We show further in Figure 3 that these fact probes not only predict factuality accurately but also produce well-calibrated confidence scores, with higher confidence predictions strongly correlating with improved factual accuracy across most model architectures.

4.2 Probes Generalize Across Domains

For a given model, can we train lightweight probes to detect factuality that generalize across diverse claim types? We investigate this by evaluating our probes under significant domain shift—training on long-form generations from various topics in LongFact (Wei et al., 2024) and testing on biographical claims from FactScore (Min et al., 2023).

Our results in Table 1 (top) and Figure 2 demonstrate that models encode generalizable represen-

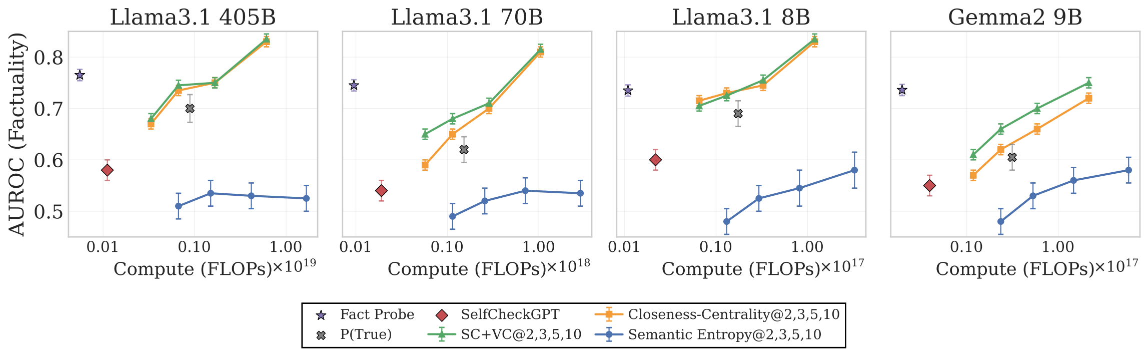


Figure 2: Factuality Probes outperform significantly more expensive sample-based methods such as Semantic Entropy, $P(\text{True})$, SelfCheckGPT, and perform on par with more costly Graph Uncertainty baselines (Jiang et al., 2024), such as Self-Consistency plus Verbalized Confidence (SC+VC) and Closeness Centrality (C_c).

Table 2: Probes trained on smaller models generalize to accurately predict the long-form factuality of generations from larger models.

π_{small} (f)	π_{gen}	**AUROC**
Llama 3.2 3B	Llama 3.2 3B	$0.7260_{\pm 0.0121}$
	Llama 3.1 8B	$0.7218_{\pm 0.0121}$
	Llama 3.1 70B	$0.7096_{\pm 0.0112}$
	Llama 3.1 405B	$0.6995_{\pm 0.0117}$
Llama 3.1 8B	Llama 3.1 8B	$0.7407_{\pm 0.0118}$
	Llama 3.1 70B	$0.7201_{\pm 0.0111}$
	Llama 3.1 405B	$0.7669_{\pm 0.0107}$
Llama 3.1 70B	Llama 3.1 70B	$0.7453_{\pm 0.0107}$
	Llama 3.1 405B	$0.7732_{\pm 0.0105}$

tations of factuality that transcend domain boundaries. These representations enable simple, efficient probes to perform well despite substantial differences between training and testing data distributions. Notably, our fact probes outperform significantly more expensive sample-based methods even when those approaches use 5 samples, at which point their computational costs are more than 10× higher than our approach. We additionally find that subjectivity filtering as employed in Jiang et al. (2024) has generally positive effects on probing performance across different π_{gen} in Figure 5.

4.3 Probes Generalize Across Model Tokens

A natural question is whether probes trained for a specific model can generalize beyond that model's generations to evaluate text from other distributions. To investigate this cross-model transferability, we fix our trained probes and assess their performance under distribution shift by evaluating claims generated by different models.

In Table 2, we evaluate generations from larger models π_{gen} using probes trained on their smaller counterparts (π_{small}) within the same model series (i.e., Llama). The evaluation process feeds tokens generated by π_{gen} through π_{small}, extracting hidden representations that are then analyzed by probes trained on the activations of π_{small}. Remarkably, these π_{small}-trained probes generalize robustly to out-of-distribution generations from π_{gen}; even probes trained on Llama 3.2-3B well-predict long-form factuality for outputs generated by Llama 3.1-405B, despite the substantial disparity in model sizes. Furthermore, the out-of-distribution performance within the same model family remains consistent with the in-distribution performances (when $\pi_{\text{small}} = \pi_{\text{gen}}$ as shown in Table 2).

We develop more challenging evaluations by training FPs on generations from open-weight models and testing them on claims produced by closed-source models. As shown in Table 1 (bottom), our probes make meaningful predictions on generations from GPT-4o-mini, with improving performance as the size of the training-time generator π_{gen} increases. We observe similar OOD generalization performance to other closed-source models, such as ChatGPT and Perplexity, in Table 3. This suggests that LLMs encode fairly universal representations of factuality that transcend specific model architectures, enabling our probes to effectively evaluate text from diverse generative sources. This generalization likely stems from the inherent representational consistency across language models. Despite architectural differences, LLMs develop similar internal representations of factual concepts through their training on overlapping data distributions.

4.4 Calibrated Fact Probing

We show in Figure 3 that fact probe predictions $\widehat{p}_c = f(h_c)$ exhibit strong calibration with actual model factuality across most tested architectures.

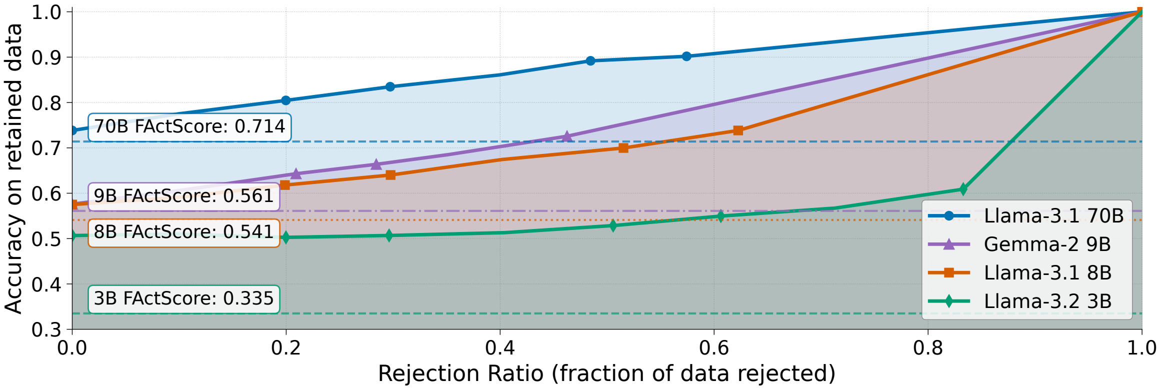


Figure 3: Fact probe predictions are well-calibrated with model factuality. In general, retaining only claims with fact probe scores (i.e., confidences) over a threshold leads to higher accuracies for all tested models. Larger models lead to improved calibration; rejection with Llama-3.2 3B fact probes only improves beyond random guessing with rejection ratio over 0.7, whereas larger models consistently improve across rejection ratios. The FActScore represents factual accuracies (i.e., the fraction of true claims).

This calibration manifests as a clear correlation between probe confidence and factual accuracy—as probe confidence increases (and rejection ratio rises), the accuracy of retained claims consistently improves. The factuality probes are well-calibrated for most models, with the exception of the smaller Llama-3.2 3B, which shows weaker calibration and only exceeds random chance performance at high rejection ratios (> 0.7). The consistent calibration observed across rejection ratios for larger models suggests that factuality signals become more distinctly encoded in hidden representations as model scale increases, enabling more reliable probing.

5 Discussion

5.1 Costs of Fact Evaluation Methods

Factuality verification methods can vary significantly in computational and monetary costs. Retrieval-based methods (e.g., Min et al. (2023)) require document searches; search-based approaches (e.g., Wei et al. (2024)) employ web searches; sampling-based methods (Jiang et al., 2024) demand many model samples with quadratically scaling costs. In contrast, our Factuality Probes require just a single forward pass per claim, resulting in dramatic efficiency gains—approximately two orders of magnitude fewer FLOPs than graph-based approaches like C_c@10 on Llama3.1-405B (cf. Figure 2). Retrieval-based methods also incur substantial API costs[1]. These efficiency advantages make FPs ideal for real-time factuality assessment without performance degradation or increased latency.

5.2 Linear Probes Are Highly Effective

The remarkable effectiveness of linear probes for factuality prediction can be attributed to three key factors. First, the standardized claim formatting (e.g., always ending in punctuation marks) provided by π_{aux} establishes a stable foundation, enabling probes to focus specifically on factuality signals rather than navigating format variations. Second, our analysis of final token representations leverages information bottlenecks where models necessarily encode their factual confidence before concluding claims. Third, Factuality Probes demonstrate exceptional versatility across diverse claim types—including definitions, behavioral statements, and propositional claims. This contrasts with methods such as Semantic Entropy or $P(\text{True})$, which perform adequately on simple definitions but struggle with complex factual assertions due to their reliance on answer consistency to direct questions—an approach poorly suited for nuanced claims where appropriate questions are difficult to formulate. By directly accessing internal model representations, FPs bypass these limitations, extracting factuality signals across diverse claim types without requiring intermediate question formulation (cf. Table 4).

6 Related Work

Short-Form Hallucination Detection. Early hallucination detection research focused primarily on detecting factual errors in brief model outputs.

[1] Based on our measurements, LongFact costs approximately \$6.6 per 1,000 claims, with about one-third of the cost for token generation and two-thirds for search queries.

Likelihood-based approaches (Desai and Durrett, 2020; Jiang et al., 2021; Malinin and Gales, 2021) leveraged model confidence scores but often suffered from overconfidence. Self-consistency methods (Manakul et al., 2023) addressed this by generating multiple responses and measuring agreement across outputs, with variations including semantic entropy (Farquhar et al., 2024), binary truth estimation (Kadavath et al., 2022), and eigenvalue-based metrics (Lin et al., 2023b). Verbalized uncertainty techniques (Lin et al., 2023a; Tian et al., 2023b; Xiong et al., 2023) emerged as a complementary strategy, directly prompting models to express confidence levels. Finally, retrieval-augmented methods (Feldman et al., 2023; Zhang et al., 2023; Peng et al., 2023) grounded outputs in external knowledge sources but introduced significant computational overhead. While establishing fundamental techniques for uncertainty estimation, these methods were created for single-claim outputs rather than complex multi-claim narratives.

Hallucination Detection in Long-Form Generations. Detecting hallucinations in long-form outputs presents unique challenges as generations contain intermingled factual and non-factual claims across multiple sentences and paragraphs. Approaches to this problem generally follow two decomposition strategies. At the sentence level, self-consistency techniques have been extended (Manakul et al., 2023; Band et al., 2024) to evaluate uncertainty by comparing multiple generated alternatives for each sentence. Atomic claim decomposition breaks long responses into minimal verifiable assertions and evaluates precision across these atomic units (Min et al., 2023). This paradigm has been expanded beyond biographical content (Wei et al., 2024; Zhao et al., 2024), establishing broader benchmarks across diverse domains. Building on atomic decomposition, recent approaches combine self-consistency with conformal prediction for claim-level uncertainty estimation (Mohri and Hashimoto, 2024), while others extract uncertainty scores directly from model internals (Duan et al., 2023; Band et al., 2024). Graph-based uncertainty metrics have been introduced to represent relationships between generated responses and their constituent claims as bipartite graphs, using centrality measures to quantify claim reliability (Jiang et al., 2024). Despite these advances, most existing methods require drawing multiple samples, creating substantial computational overhead that our probe-based approach addresses.

Latent-Space Probing of LLMs for Hallucination Detection. Recent research has demonstrated that LLM hidden states encode rich signals about output factuality, offering a promising avenue for hallucination detection. Specific directions in the latent space corresponding to "truthfulness" in model outputs have been identified (Marks and Tegmark, 2023). Probes trained on hidden states have been shown to effectively predict factuality when supervised with accuracy labels (Azaria and Mitchell, 2023; Liu et al., 2024; Ji et al., 2024). More recently, these methods have been extended to achieve fine-grained factuality prediction at the word level (He et al., 2024). Unsupervised approaches have also emerged, with Su et al. (2024) introducing real-time detection using contextualized embeddings from different Transformer layers, and Zablocki and Gajewska (2024) providing some analysis of hallucination risks through internal state examination, though most remain limited—restricted to binary questions (Burns et al., 2023), requiring accuracy labels for tuning (Zou et al., 2023), or demanding multiple costly generations for training (Chen et al., 2024; Kossen et al., 2024) all on short-form generations. Unlike these methods, our approach efficiently processes complex long-form text with a single forward pass for each paragraph and atomic claim, and generalizes across both domains and model architectures—addressing key limitations in previous probing techniques.

7 Conclusion

We introduce Factuality Probes, a method that leverages LLM hidden representations to efficiently assess the factuality of generated claims. Our lightweight linear probes achieve comparable or superior performance to current approaches while requiring up to 100× fewer computational resources, making real-time factuality assessment practical. These probes generalize effectively across both subject domains and model architectures, even evaluating claims from closed-source models when trained on open-weight architectures. Efficient factuality assessment methods like ours will be essential for responsible AI deployment.

Acknowledgements

NB acknowledges funding from an NSF Graduate Research Fellowship, a Quad Fellowship, and a Mercor Graduate Fellowship. TGJR acknowledges support from the Foundational Research Grants program at Georgetown University's Center for Security and Emerging Technology.

Contributions

JH, NB, and MR started the project. JH performed most of the experimental work and wrote a first draft of the manuscript. NB and MR provided guidance on experiments, substantially developed the paper's argument, and revised all sections. MR conducted the FP experiments using the Llama 3.1-405B model. JK contributed to conceptualization and experimental design during the initial phase of the project. TR suggested methodological improvements and provided extensive edits to the final manuscript. YG provided feedback and guidance on the paper's motivation.

References

Amos Azaria and Tom Mitchell. 2023. The internal state of an llm knows when it's lying. In *EMNLP*.

Neil Band, Xuechen Li, Tengyu Ma, and Tatsunori Hashimoto. 2024. Linguistic calibration of long-form generations. In *Forty-first International Conference on Machine Learning*.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. 2023. Discovering latent knowledge in language models without supervision. In *ICLR*.

Chao Chen, Kai Liu, Ze Chen, Yi Gu, Yue Wu, Mingyuan Tao, Zhihang Fu, and Jieping Ye. 2024. Inside: Llms' internal states retain the power of hallucination detection. *Preprint*, arXiv:2402.03744.

Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 785–794. ACM.

Google DeepMind. 2025. Gemini 2.5 pro. [Online; accessed 19-May-2025].

Shrey Desai and Greg Durrett. 2020. Calibration of pre-trained transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 295–302.

Jinhao Duan, Hao Cheng, Shiqi Wang, Chenan Wang, Alex Zavalny, Renjing Xu, Bhavya Kailkhura, and Kaidi Xu. 2023. Shifting attention to relevance: Towards the uncertainty estimation of large language models. *arXiv:2307.01379*.

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, and 6 others. 2021. A mathematical framework for transformer circuits. *Transformer Circuits Thread*. Https://transformer-circuits.pub/2021/framework/index.html.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. 2024. Detecting Hallucinations in Large Language Models Using Semantic Entropy. *Nature*.

Philip Feldman, James R Foulds, and Shimei Pan. 2023. Trapping llm hallucinations using tagged context prompts. *arXiv:2306.06085*.

Gabriel Goh. 2016. Decoding the thought vector.

Jiatong Han, Jannik Kossen, Muhammed Razzak, and Yarin Gal. 2024. Semantic entropy neurons: Encoding semantic uncertainty in the latent space of llms. In *NeurIPS 2024 Workshop on Mechanistic Interpretability (MINT)*.

Jinwen He, Yujia Gong, Zijin Lin, Cheng'an Wei, Yue Zhao, and Kai Chen. 2024. LLM factoscope: Uncovering LLMs' factual discernment through measuring inner states. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 10218–10230, Bangkok, Thailand. Association for Computational Linguistics.

Ziwei Ji, Delong Chen, Etsuko Ishii, Samuel Cahyawijaya, Yejin Bang, Bryan Wilie, and Pascale Fung. 2024. LLM internal states reveal hallucination risk faced with a query. In *Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pages 88–104, Miami, Florida, US. Association for Computational Linguistics.

Mingjian Jiang, Yangjun Ruan, Prasanna Sattigeri, Salim Roukos, and Tatsunori Hashimoto. 2024. Graph-based uncertainty metrics for long-form language model outputs. *Preprint*, arXiv:2410.20783.

Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. 2021. 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.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield Dodds, Nova DasSarma, Eli Tran-Johnson, and 1 others. 2022. Language models (mostly) know what they know. *arXiv:2207.05221*.

Jannik Kossen, Jiatong Han, Muhammed Razzak, Lisa Schut, Shreshth Malik, and Yarin Gal. 2024. Semantic entropy probes: Robust and cheap hallucination detection in llms. *Preprint*, arXiv:2406.15927.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *ICLR*.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. 2024. Inference-time intervention: Eliciting truthful answers from a language model. *NeurIPS*, 36.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2023a. Teaching models to express their uncertainty in words. *TMLR*.

Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. 2023b. Generating with confidence: Uncertainty quantification for black-box large language models. *arXiv:2305.19187*.

Linyu Liu, Yu Pan, Xiaocheng Li, and Guanting Chen. 2024. Uncertainty estimation and quantification for LLMs: A simple supervised approach. ArXiv preprint `arXiv:2404.15993`. Preprint (Apr 2024).

Andrey Malinin and Mark Gales. 2021. Uncertainty estimation in autoregressive structured prediction. *ICLR*.

Potsawee Manakul, Adian Liusie, and Mark JF Gales. 2023. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. In *Conference on Empirical Methods in Natural Language Processing*.

Samuel Marks and Max Tegmark. 2023. The Geometry of Truth: Emergent Linear Structure in Large Language Model Representations of True/False Datasets. *arXiv 2310.06824*.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. *EMNLP*.

Christopher Mohri and Tatsunori Hashimoto. 2024. Language models with conformal factuality guarantees. *Preprint*, arXiv:2402.10978.

Neel Nanda, Senthooran Rajamanoharan, János Kramar, and Rohin Shah. 2023. Fact finding: Attempting to reverse-engineer factual recall on the neuron level.

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

OpenAI. 2025a. Introducing codex. [Online; accessed 19-May-2025].

OpenAI. 2025b. Introducing deep research. [Online; accessed 19-May-2025].

OpenAI. 2025c. Introducing gpt-5. [Online; accessed 19-Sept-2025].

OpenAI. 2025d. Introducing openai o3 and o4-mini. [Online; accessed 19-May-2025].

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, and 1 others. 2011. Scikit-learn: Machine learning in Python. *JMLR*, 12.

Baolin Peng, Michel Galley, Pengcheng He, Hao Cheng, Yujia Xie, Yu Hu, Qiuyuan Huang, Lars Liden, Zhou Yu, Weizhu Chen, and 1 others. 2023. Check your facts and try again: Improving large language models with external knowledge and automated feedback. *arXiv:2302.12813*.

Weihang Su, Changyue Wang, Qingyao Ai, Yiran Hu, Zhijing Wu, Yujia Zhou, and Yiqun Liu. 2024. Unsupervised real-time hallucination detection based on the internal states of large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14379–14391, Bangkok, Thailand. Association for Computational Linguistics.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D. Manning. 2023a. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. *Preprint*, arXiv:2305.14975.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D. Manning. 2023b. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. *Preprint*, arXiv:2305.14975.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. 2024. Activation addition: Steering language models without optimization. *Preprint*, arXiv:2308.10248.

Jerry Wei, Chengrun Yang, Xinying Song, Yifeng Lu, Nathan Hu, Jie Huang, Dustin Tran, Daiyi Peng, Ruibo Liu, Da Huang, Cosmo Du, and Quoc V. Le. 2024. Long-form factuality in large language models. *Preprint*, arXiv:2403.18802.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. 2023. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. *Preprint*, arXiv:2306.13063.

Piotr Zablocki and Zofia Gajewska. 2024. Assessing hallucination risks in large language models through internal state analysis. *Authorea Preprints*.

Shuo Zhang, Liangming Pan, Junzhou Zhao, and William Yang Wang. 2023. Mitigating language model hallucination with interactive question-knowledge alignment. *arXiv:2305.13669*.

Wenting Zhao, Tanya Goyal, Yu Ying Chiu, Liwei Jiang, Benjamin Newman, Abhilasha Ravichander, Khyathi Chandu, Ronan Le Bras, Claire Cardie, Yuntian Deng, and Yejin Choi. 2024. Wildhallucinations: Evaluating long-form factuality in llms with real-world entity queries. *Preprint*, arXiv:2407.17468.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, and 1 others. 2023. Representation engineering: A top-down approach to ai transparency. *arXiv:2310.01405*.

Appendix

A Limitations

We discuss several limitations of our factuality probing approach that we aspire to address in future work.

Use of auxiliary model for claim breakdown and refinement. As is common with many approaches to assessing factuality (Section 3.1), our fact probe also relies on this critical preprocessing step. The quality and consistency of claim decomposition directly impacts probe performance, and variations in how the auxiliary model segments and refines claims may affect factuality assessments. While this dependency is shared across factuality assessment methods, it represents a computational bottleneck and potential source of inconsistency in our pipeline (cf. Table 4).

Factual dependencies in long-form generation claims. We do not currently leverage the contextual relationships between claims when predicting factuality. Long-form text often contains interdependent claims where the factuality of one assertion may influence the likelihood of another being correct. Our token-level approach, while computationally efficient, treats each claim independently and does not model these higher-order dependencies that could potentially enhance factuality prediction.

B Potential Risks

While our research aims to promote safer and more responsible LLM deployment, we acknowledge potential risks associated with factuality probing techniques. Notably, malicious actors could exploit these insights to deliberately compromise generator LM calibration, potentially increasing model confidence in false outputs (Han et al., 2024). Such vulnerabilities highlight the importance of responsible disclosure and implementation of safeguards when deploying such factuality assessment tools.

C Additional Results

C.1 Generalization to Closed-source Models

Table 3 demonstrates our Factuality Probes' strong generalization capabilities to closed-source models. Probes trained on smaller, open-weight models achieve promising AUROC scores when detecting hallucinations in outputs from proprietary models. Consistently across training models, performance is strongest on InstructGPT, followed by ChatGPT, with PerplexityAI showing the lowest scores. Notably, the 3B Llama model's probes rival those of the 70B variant, suggesting efficient lightweight probing solutions remain viable even for evaluating closed-source models.

C.2 Reversed Out-of-Domain Data Test

We reverse our training and testing paradigm by training fact probes on biographical claims annotated with FActScore labels, then evaluating their performance on the more topically diverse claims measured with LongFact. We show in Figure 4 that reversing the datasets makes the fact probes generalize worse.

C.3 Probe with Subjectivity Filtering

Following the procedure outlined in Jiang et al. (2024), we apply subjectivity filtering to our test claims. As shown in Figure 5, this filtering generally improves probing performance across models.

C.4 Probing on Different Token Positions

To better understand how probing different token positions affects performance, we train fact probes on the activations of the first, last, and second-to-last output tokens. As shown in Appendix C.4, our results validate the claim that probing the final tokens is highly effective, which we explain intuitively in Section 5.2.

Table 3: Fact Probes generalize their use across LLMs. AUROC scores for probes trained on open-weight models and evaluated on closed-source models. Evaluation data is taken from human-annotated examples published in Min et al. (2023).

Training π_{gen} (f)	**Testing π_{gen}**	**AUROC**
Llama 3.1-8B	InstructGPT	0.732 ± 0.008
	ChatGPT	0.698 ± 0.008
	PerplexityAI	0.628 ± 0.010
Llama 3.2-3B	InstructGPT	0.692 ± 0.008
	ChatGPT	0.689 ± 0.008
	PerplexityAI	0.623 ± 0.010
Llama 3.1-70B	InstructGPT	0.701 ± 0.008
	ChatGPT	0.650 ± 0.008
	PerplexityAI	0.605 ± 0.011
Gemma 2-9B	InstructGPT	0.612 ± 0.009
	ChatGPT	0.604 ± 0.008
	PerplexityAI	0.551 ± 0.011

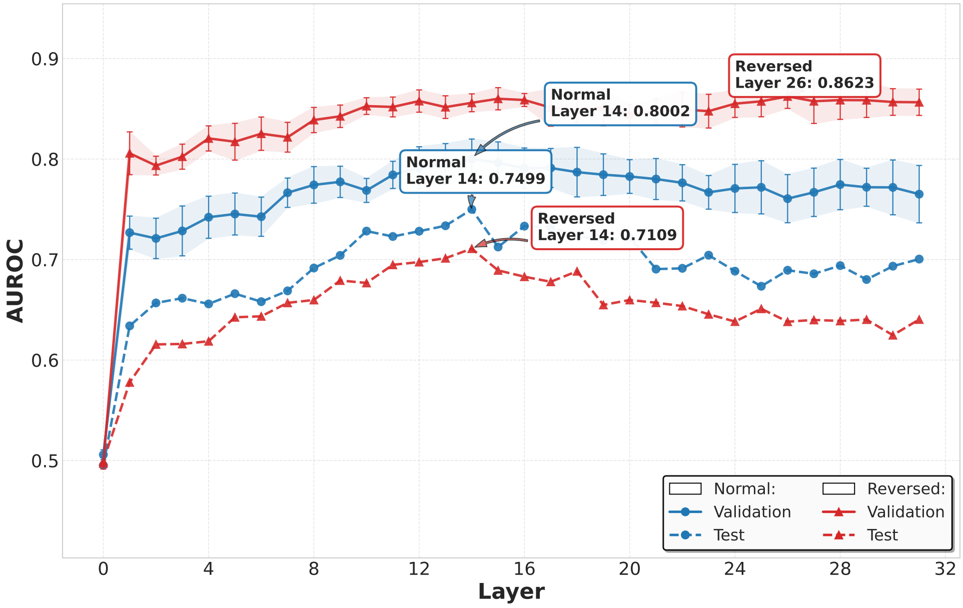
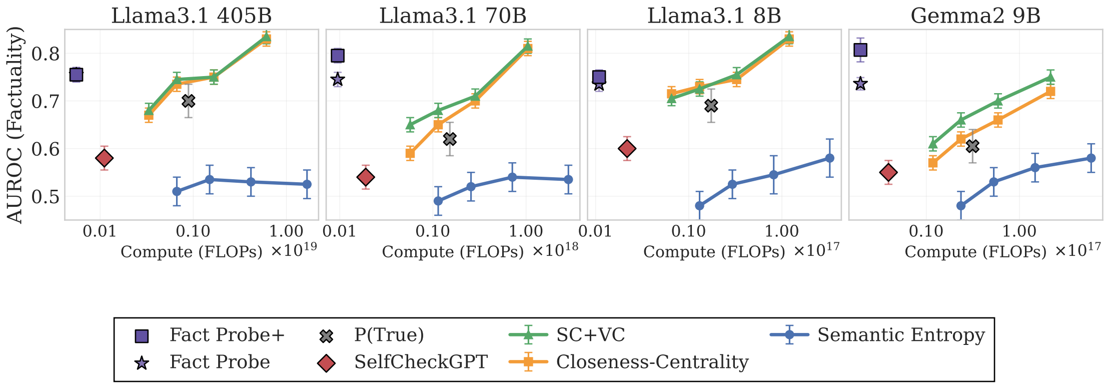


Figure 4: Reversing training and testing dataset can cause worse probe generalization (π_{gen} is Llama3.1-8B).



FP+ represents FP with subjectivity filtering applied

Figure 5: Subjectivity filtering generally boosts fact probing performances across LLMs.

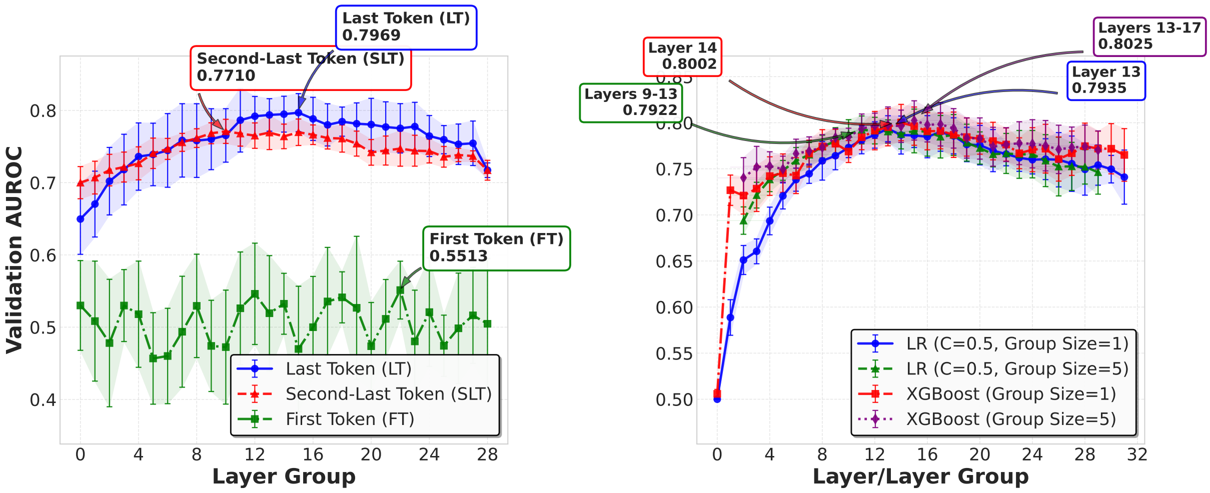


Figure 6: (Left) probing the hidden states of the last tokens of the claims yields the best performances. (Right) we observe comparable performances of linear probe instantiations f with different layer group sizes. Experiments conducted on Llama 3.1 8B.

D Experiment Details

We set the maximum number of new tokens to 512 for all LLMs, and sample under the temperature of 0.7 using top-k sampling with $k = 50$.

Factual Evaluation and Claim Processing Pipeline. Evaluating factuality in long-form generations requires decomposing complex texts into atomic claims that can be individually verified. Current approaches differ in how they process and evaluate these claims. FActScore (Min et al., 2023) focus primarily on precision—the ratio of supported claims to all evaluated claims—without accounting for recall. As noted by its authors, this approach may favor models that generate fewer facts or abstain from making claims, potentially missing important information expected by users. LongFact (Wei et al., 2024) address this limitation by evaluating both precision and recall, providing a more balanced assessment of factual completeness. It employs web search-based verification and handles a broader range of topics beyond the biographical focus of FActScore.

The processing pipeline for factuality assessment typically includes several essential steps, as illustrated in Table 4. First, the long-form generation is broken down into atomic claims that represent the smallest self-contained factual assertions. These claims then undergo refinement to ensure they are well-formed and contain all necessary context (e.g., replacing pronouns with their referents). Following refinement, claims may be filtered for relevance and subjectivity to focus evaluation on verifiable factual content. Next, depending on the assessment methodology, the pipeline may include question generation, high-temperature sampling, or entailment checking to determine claim factuality.

Our Factuality Probes approach maintains the critical initial steps of claim breakdown and refinement but streamlines the evaluation process by eliminating the need for additional sampling, question generation, or explicit entailment checking. Instead, we directly leverage the model's internal representations to assess factuality, substantially reducing computational requirements while maintaining strong evaluation performance (cf. Figure 2).

Processing Step	SE	G-Unc.	SelfCheckGPT	P(True)	Fact Probe
Long-form Gen.	$\mathrm{M_t}$	$\mathrm{M_t}$	$\mathrm{M_t}$	$\mathrm{M_t}$	$\mathrm{M_t}$
Claim Breakdown	$\mathrm{M_a}$	$\mathrm{M_a}$	$\mathrm{M_a}$	$\mathrm{M_a}$	$\mathrm{M_a}$
Claim Revision	$\mathrm{M_a}$	$\mathrm{M_a}$	$\mathrm{M_a}$	$\mathrm{M_a}$	$\mathrm{M_a}$
Irrel. Filtering	$\mathrm{M_a}$	$\mathrm{M_a}$	$\mathrm{M_a}$	$\mathrm{M_a}$	$\mathrm{M_a}$
Subj. Filtering	N/A	$\mathrm{M_a}$	N/A	N/A	N/A
Claim Merging	N/A	$\mathrm{M_a}$	N/A	N/A	N/A
Question Gen.	$\mathrm{M_a}$	N/A	N/A	$\mathrm{M_a}$	N/A
HT Sampling	$\mathrm{M_t}$	$\mathrm{M_t}$	N/A	$\mathrm{M_t}$	N/A
Entail./Equiv. Check	$\mathrm{M_a}$	$\mathrm{M_a}$	$\mathrm{M_t}$	$\mathrm{M_t}$	N/A

Table 4: FP has the least claim processing steps. The supportedness scoring is uniformly applied to all approaches after possible claim revision, and relevance or subjectivity filtering. Graph Uncertainty (G. Unc) methods include SC+VC and C_c.

Hyperparameters. For our fact probing experiments, the logistic regression model employed L1 regularization with the liblinear solver and a configurable regularization strength (C) which we set as 0.5 throughout. For XGBoost, we used 1000 estimators, learning rate of 0.1, and maximum tree depth of 6, configured for binary classification with AUC as the evaluation metric. XGBoost was accelerated using GPU computation where applicable to handle the large-scale hidden state analysis efficiently. For both models, we used the standard implementations as provided in the `scikit-learn` (Pedregosa et al., 2011) and `xgboost` (Chen and Guestrin, 2016) PyPI libraries, with parameter configurations as mentioned above.

Layer Group Size. We employ 5-layer groups when concatenating hidden state representations to validate their effectiveness across most models. For Llama-3.1 405B, we reduce this to 3-layer groups to manage memory consumption while maintaining convergence efficiency. This adjustment balances computational feasibility with model performance for our largest architecture.

Sparse Probes. We show the clamped neuron activations in Table 6. We demonstrate the performance comparisons between probes trained only on sparse activation values (in Table 6) and those trained on full-neuron activations with a high l_1 penalty.

Computing FLOPs. To compare the computational efficiency of different factuality prediction methods, we calculate the floating-point operations (FLOPs) required by each method. Concretely, fact probing require only a single forward pass per claim, while methods like $P(\text{True})$ require multiple passes. Sample-based methods scale with the sampling size k, with costs growing quadratically (in the worst case) at different rates depending on the algorithm. For instance, sampling-based methods like SE incur costs proportional to k^2, while graph-based approaches scale with $k(k + 1)$.

Sampling for SE. We randomly sampled 20% of the test claims without replacement to evaluate SE from those used to test fact probes. This was mainly due to the prohibitively high costs of SE, which is quadratic to the number of claims.

Training Samples (rated by Jiang et al. (2024))		
Model	**Number of Claims**	**Number of Topics**
Llama3.2-3B	1,531	25
Llama3.1-8B	3,374	50
Llama3.1-70B	4,204	75
Llama3.1-405B	4,469	40
Gemma2-9B	1,508	25
Test Samples (rated by Min et al. (2023))		
Model	**Number of Claims**	**Number of Entities**
Llama3.2-3B	1,867	30
Llama3.1-8B	1,732	30
Llama3.1-70B	2,028	30
Llama3.1-405B	2,138	30
Gemma2-9B	1,587	30
GPT-4o-Mini (OOD)	1,386	30

Table 5: Number of training and testing samples for evaluating fact probe generalization across subject domains. **Notes:** The number of topics vary across LLMs since they tend to (consistently) respond in very different token lengths, even though instructed with the same amount of maximum output sequence tokens. More claims are required for probes on larger LLMs to take better effect empirically.

Computational Budgets. Converting to hardware requirements, our experiments consumed approximately 440 GPU hours on A100 nodes from OATML lab machine and cloud machines from third-party vendors, amounting to approximately 2×10^{20} FLOPs (cf. Figure 2, multiplied with the number of test entities).

Reproducibility. All reported test metrics (i.e., AUROC) include bootstrapped results with standard errors on a 95% confidence interval to ensure statistical reliability of our comparisons.

E Steering LLMs for Better Factuality

Prior works such as (Goh, 2016; Olah et al., 2020; Elhage et al., 2021; Nanda et al., 2023) seek to find the internal workings of neural networks by interpreting individual neurons and their interactions in deciding model behavior. Olah et al. (2020) suggest that neural networks develop legible internal representations of features, which can be connected to form interpretable circuits. These features are causally meaningful variables that can be leveraged to steer the model, much like steering vectors. Meanwhile, it has been demonstrated that model behaviors can be steered by adding a vector to the model hidden states (Li et al., 2024; Turner et al., 2024), derived by calculating the differences in activation averages between specific model behavioral classes, similar to the activation clamping approach we will employ.

E.1 Finding Factuality Neurons

We identify a subset of neurons predominantly responsible for encoding *common* factuality (Table 6). We locate neuron activations such that supported and unsupported claims can be linearly separated by some threshold. And using only sparse neuron activations we can train probes performing comparably well with full-neuron probes (Figure 7).

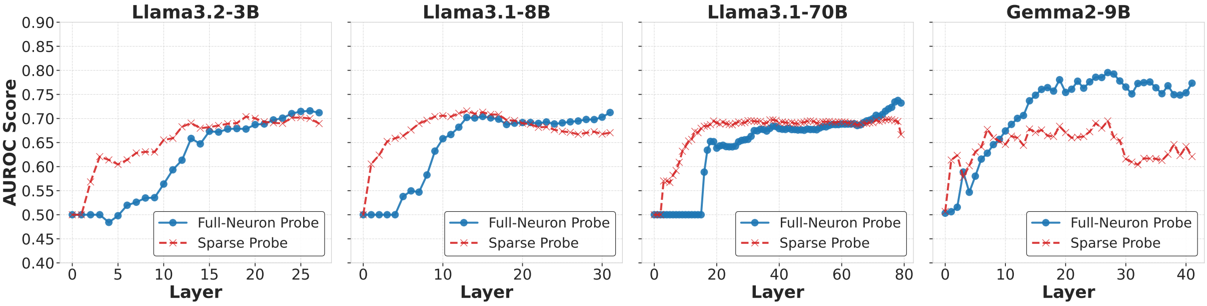


Figure 7: Probes trained only on selected neuron activations (or sparse probes) have comparable validation AUROC as compared with full-neuron probes trained with high sparsity constraints (or full probes).

By clamping neuron activations to the tail range of their "supported" class activation values (Table 6), we can causally reduce hallucinations in model outputs. As demonstrated in Table 7, this neuron-level steering approach significantly improves factuality across all evaluated LLMs on diverse topics. This presents valuable opportunities for enhancing LLM factuality through activation steering during inference, without requiring additional prompting or fine-tuning.

Table 6: Activation statistics for factuality neuron clamping. We select neurons based on linear weights of probes trained on grouped hidden states by layers, and the ‘Dimension‘ field is counted based on concatenated hidden dimensions.

Model	Layer	Neuron	Dimension	Avg. Act. (Supported)	Avg. Act. (Not Supported)	Clamping Value
Llama 3.2-3B	9	103	103	0.329 ± 0.204	0.205 ± 0.193	0.737
	13	103	12391	0.476 ± 0.362	0.289 ± 0.319	1.200
	13	859	13147	0.095 ± 0.320	-0.017 ± 0.310	0.736
	13	1534	13822	0.262 ± 0.307	0.429 ± 0.287	-0.351
	13	1947	14235	-0.820 ± 0.228	-0.735 ± 0.226	-1.277
Gemma 2-9B	27	169	10921	-0.077 ± 2.746	-1.151 ± 2.596	5.415
	27	2841	13593	-0.949 ± 2.887	-1.278 ± 3.010	4.825
	28	852	15188	0.984 ± 2.940	-0.078 ± 2.750	6.864
	28	1913	16249	1.574 ± 3.664	0.019 ± 3.723	8.902
	28	2612	16948	-0.064 ± 3.140	1.314 ± 3.160	-6.344
Llama 3.1-8B	14	133	12421	0.323 ± 0.202	0.214 ± 0.188	0.727
	14	709	12997	-0.311 ± 0.231	-0.196 ± 0.214	-0.773
	14	1162	13450	-0.301 ± 0.177	-0.356 ± 0.165	-0.054
	15	2485	18869	0.821 ± 0.200	0.729 ± 0.210	1.222
	15	2629	19013	-0.367 ± 0.254	-0.267 ± 0.252	-0.875
Llama 3.1-70B	77	4030	28606	3.229 ± 0.925	2.885 ± 0.958	5.080
	77	6890	31466	-1.355 ± 0.766	-1.099 ± 0.822	-2.887
	78	1071	33839	-0.065 ± 1.357	0.370 ± 1.494	-2.779
	78	4030	36798	3.549 ± 0.981	3.203 ± 1.007	5.511
	78	4994	37762	1.190 ± 0.866	0.921 ± 0.856	2.922

Table 7: Steering factuality neurons significantly improves LLM correctness. Evaluated on biographies of 10 random entities from WikiPedia rated by FactScore, or 10 random topics from LongFact. Statistical significance: paired t-test, $t(5) = 4.76, p < 0.01$.

Model	Metric	Before	After	Improvement (%)
Llama 3.2-3B	FactScore	0.286	0.316	+10.5%
	LongFact F1	0.538	0.553	+2.8%
Llama 3.1-8B	FactScore	0.599	0.664	+10.9%
	LongFact F1	0.588	0.622	+5.8%
Llama 3.1-70B	FactScore	0.568	0.609	+7.2%
Gemma 2-9B	FactScore	0.606	0.627	+3.5%