Hallucination Detection via Attribution-Based Information Flow Analysis

Anonymous ACL submission

Abstract

Large language models (LLMs) inevitably generate hallucinations—factually incorrect or contextually inconsistent outputs—that undermine their reliability in high-stakes applications. Existing probe-based hallucination detectors rely on static snapshots of hidden states or coarse-grained aggregations of attention weights, overlooking the rich, dynamic signals encoded in how information flows through the model's internal components. We propose the **Information Flow Probe (IFP)**, which extracts three complementary features at every layer: per-head attention contribution (AC), FFN contribution (FC), and residual contribution (RC), computed via an attribution-based decomposition of the residual stream. We first demonstrate empirically that these features exhibit systematically distinct patterns for factual versus hallucinated answers even without any classifier training, then combine them as input to a lightweight logistic regression probe. Experiments across three mainstream open-source LLMs and seven QA datasets show that IFP consistently outperforms both training-free and training-based baselines and achieves strong cross-dataset generalization.

1 Introduction

Large Language Models (LLMs) achieve outstanding performance across various natural language processing domains, including question answering, information extraction, summarization, and more (Yang et al., 2025; Grattafiori et al., 2024; Team et al., 2024). However, they inevitably generate responses that are factually incorrect, contextually inconsistent, or fabricate nonexistent information—a phenomenon known as hallucination (Zhang et al., 2025a; Sahoo et al., 2024; Alansari and Luqman, 2025). In practical applications, hallucinations severely undermine the reliability of LLMs, particularly in fields with low tolerance for error, such as healthcare and law. Consequently, hallucination detection techniques have received significant attention.

Existing approaches for hallucination detection in LLMs are broadly divided into black-box and white-box paradigms. Black-box approaches typically involve prompting the model for self-critique or estimating uncertainty through semantic clustering of sampled responses. White-box methods instead exploit internal model transparency; among these, probing-based classifiers have emerged as a dominant direction owing to their high diagnostic accuracy. Current probes primarily use static snapshots of hidden states or attention maps as features, treating each layer's representation as an independent signal. More recent work such as ICR-Probe (Zhang et al., 2025b) takes a step toward dynamic features by measuring, at each layer, the Jensen-Shannon divergence between the head-averaged attention distribution and the direction of the residual stream update—yielding a single scalar per layer that reflects the relative dominance of attention versus FFN in driving the update. However, averaging attention weights across all heads before computing this divergence discards the fine-grained, per-head routing structure—precisely the information that reveals *where* the model's attention is directed when it hallucinates.

To address this gap, we propose the **Information Flow Probe (IFP)**, which captures the complete information flow pattern across all three architectural pathways—per-head attention contribution (AC), FFN contribution (FC), and residual contribution (RC)—at every layer of the LLM. We first show empirically that these features exhibit systematically different patterns for factual versus hallucinated answers even before any classifier is trained, then use them as input to a lightweight logistic regression probe. Experiments across three mainstream open-source LLMs and seven datasets demonstrate that IFP consistently outperforms existing methods and exhibits strong cross-dataset generalization.

Our contributions are as follows:

1. We analyze information flow patterns inside LLMs and show that per-head attention contribution, FFN contribution, and residual contribution carry inherent discriminative signal for hallucination detection.

2. We propose IFP, a probe that leverages the complete layer-wise information flow pattern as features. IFP achieves state-of-the-art performance among model-internal probing methods across three LLMs and seven datasets.

3. Ablation studies and analyses demonstrate that IFP is robust, generalizable across datasets, and interpretable, offering a promising direction for understanding and mitigating LLM hallucinations.

2 Related Work

2.1 Hallucination Detection via Probing

Probing classifiers use a model's internal representations to predict its output behavior, and have become a principal tool for hallucination detection. Early work established that hallucinated and factual responses are often linearly separable in the hidden-state space (Marks and Tegmark, 2024; Azaria and Mitchell, 2023a). Subsequent methods diversified the feature space: some probes operate on the hidden state of the question alone, detecting hallucination risk before generation begins (Snyder et al., 2024; Gottesman and Geva, 2024); others adopt richer supervision signals (Kossen et al., 2024) or multi-layer hidden state aggregation (Azaria and Mitchell, 2023b). More recently, attention to dynamic signals has grown: Li et al. (2025) exploits token-level hidden-state dynamics, and ICR-Probe (Zhang et al., 2025b) introduces a per-layer feature that measures the Jensen-Shannon divergence between the head-averaged attention distribution and the direction of the residual stream update, capturing the relative dominance of attention versus FFN contributions. While ICR-Probe represents an important step toward capturing architectural interactions, averaging attention weights across all heads before computing this divergence discards the fine-grained, per-head routing structure that encodes how the model distributes its attention across question and answer tokens. IFP addresses this gap by training a probe on the *complete* layer-wise information flow pattern—retaining all attention heads, FFN pathways, and residual streams—and demonstrates that this richer representation yields consistent improvements in both in-distribution and cross-dataset evaluation.

2.2 Mechanistic Interpretability

Mechanistic interpretability aims to explain model behavior at the level of individual components and circuits (Rai et al., 2025). Causal mediation analysis provides an influential framework for this: by intervening on intermediate representations and measuring the effect on outputs, it attributes model behavior to specific components. Meng et al. (2022) use this approach to identify intermediate-layer FFNs as primary sites of factual knowledge storage. More recent work has linked internal representations directly to hallucination: Gao et al. (2025) identify *hallucination neurons* whose activation correlates with hallucinated outputs, and show that suppressing them reduces hallucination rates. Ferrando et al. (2025) employ Sparse Autoencoders (SAEs) to characterize the knowledge boundaries of LLMs at the entity level. On the attribution side, Ferrando and Voita (2024) propose an information flow routing framework that efficiently estimates each component's contribution to a target token in a single forward pass. IFP builds directly on this attribution framework, using the resulting per-layer, per-head contribution scores as probe features to connect mechanistic interpretability with practical hallucination detection.

3 Preliminaries

3.1 Transformer

As shown in Figure 1, each layer of an LLM updates token representations through three additive pathways: multi-head attention, the FFN block, and the residual connection. Understanding how information flows through these pathways across layers provides a fine-grained picture of the model's inference process—and, as we show, a discriminative signal for hallucination detection.

3.2 Information Flow in LLMs

Inside a Transformer-based LLM, the computation for each token can be understood as a progressive refinement of its representation through a series of additive updates. At every layer, the token's hidden state is carried forward by the *residual stream* (**?**) and simultaneously updated by two parallel path-

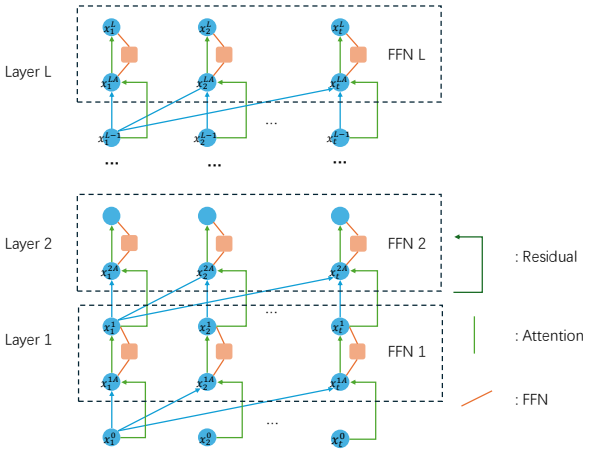

Figure 1: Information flow through a single transformer layer. Each token's representation is updated via attention edges (from all attending source tokens), the FFN edge, and the residual connection.

ways: the multi-head attention block, which gathers information from all preceding token positions, and the FFN block, which injects new content derived from parametric knowledge. This additive structure naturally induces a computation graph in which nodes correspond to per-layer token representations and directed edges correspond to the vector contributions flowing between them (Ferrando and Voita, 2024). Rather than relying on activation patching—which requires manually crafted contrastive templates and a separate forward pass per intervention—Ferrando and Voita (2024) propose to quantify edge importance through attribution, recovering task-relevant information flow routes in a single forward pass.

Concretely, each node y is expressed as a sum of incoming edge vectors $y = z_1 + \cdots + z_m$, and the importance of edge z_j is defined as its normalized proximity to y:

$$\text{importance}(z_j, y) = \frac{\text{proximity}(z_j, y)}{\sum_k \text{proximity}(z_k, y)}, \tag{1}$$

where proximity is measured using negative ℓ_2 distance (Ferrando et al., 2022):

$$\text{proximity}(z_j, y) = -\|z_j - y\|. \tag{2}$$

Attention and FFN Sub-layer Decomposition Each Transformer layer applies two successive residual updates. In the *attention sub-layer*, the input x_t^{l-1} is carried forward via a residual connection while the attention block aggregates information from preceding tokens:

$$x_t^{lA} = x_t^{l-1} + \sum_{h=1}^{H} W_O^h \sum_s \alpha_{h,t,s}^l \, W_V^h \, x_s^{l-1}, \tag{3}$$

where $\alpha_{h,t,s}^l$ is the attention weight from token t to source s at head h, and W_V^h, W_O^h are the value and output projection matrices. The vector contributed by source token s to target token t through head h is:

$$z_{s\to t}^{l,h} = W_O^h(\alpha_{h,t,s}^l \, W_V^h \, x_s^{l-1}), \tag{4}$$

and its importance is computed relative to the full attention sub-layer output x_t^{lA}.

In the *FFN sub-layer*, the attention output x_t^{lA} serves as the new residual, and the FFN block adds transformed content:

$$x_t^l = x_t^{lA} + \text{FFN}_l(x_t^{lA}), \tag{5}$$

yielding two additive edges at this stage: the *residual edge* (carrying x_t^{lA} forward) and the *FFN edge* (introducing new parametric content). Applying Equation (1) to Equation (5), the importance of the FFN edge quantifies how much new knowledge the FFN injects, while the importance of the residual edge quantifies how much of the attention sub-layer's output is carried forward unmodified into x_t^l.

The resulting decomposition provides a fine-grained view of how information is routed through the model: which source tokens a given target attends to (attention edges, resolved per head), how much new parametric knowledge is retrieved at each layer (FFN edge), and how much of the attention output is carried over into the final layer representation (residual edge). Crucially, this characterization is substantially richer than the scalar used by ICR-Probe (Zhang et al., 2025b), which aggregates the entire attention block into a single per-layer value and thereby discards the head-level routing structure. By retaining the complete layer-by-layer trajectories across all three pathways—including the head-granularity attention contribution—the framework captures a more complete picture of how question-relevant information propagates toward the generated answer token. As we show in Section 5.5, this completeness translates into inherently discriminative features: factual answers and hallucinations exhibit systematically different information flow patterns even before any probe training. This observation motivates the design of IFP, described in the following section.

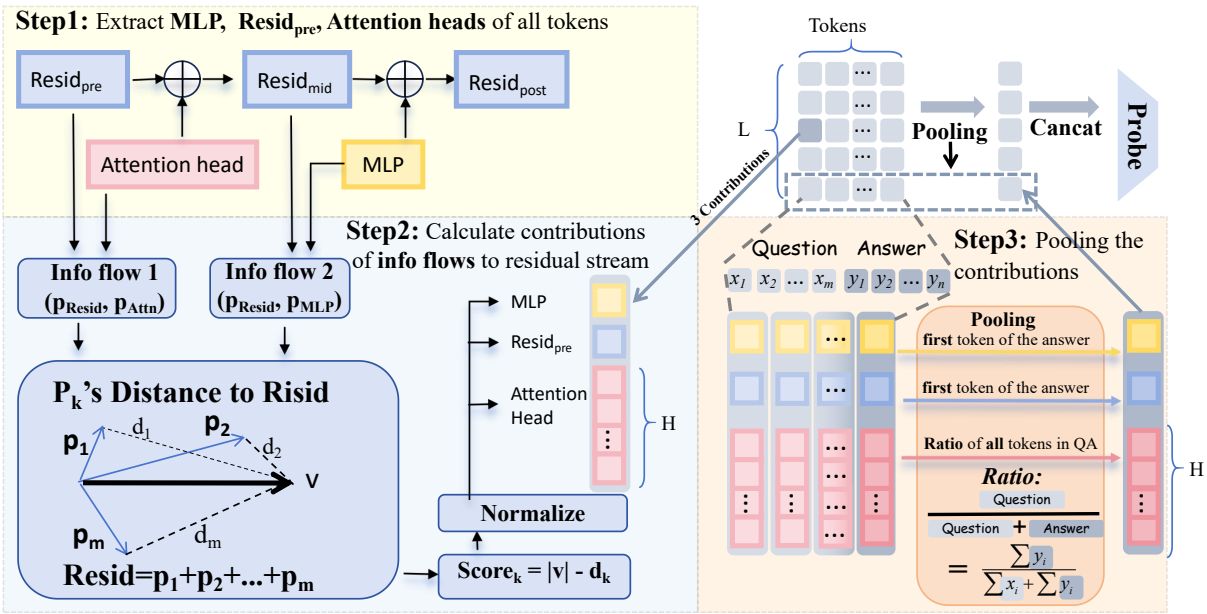

Figure 2: Overview of IFP. For each QA pair, we extract hidden representations at every layer and compute three information flow features: per-head attention contribution (AC) at the last answer token, and FFN contribution (FC) and residual contribution (RC) at the first answer token. These features are concatenated into a single vector and fed to a logistic regression classifier for hallucination detection.

4 Information Flow Probe

4.1 Attention Contribution

Given an LLM with L layers and H attention heads per layer, consider a QA pair where the question occupies N tokens $Q = \{q_1, q_2, \ldots, q_N\}$ and the generated answer occupies M tokens $A = \{a_1, a_2, \ldots, a_M\}$. We characterize how each attention head routes information by computing, for head h at layer l, the total attention weight received from the question and answer tokens at the *last* answer token a_M:

$$C^{l,h}\,(\text{question}) = \sum_{i=1}^{N} \alpha^{l}_{h,i}, \tag{6}$$

$$C^{l,h}\,(\text{answer}) = \sum_{j=1}^{M} \alpha^{l}_{h,j}, \tag{7}$$

where $\alpha^{l}_{h,i}$ is the attention weight assigned by head h at layer l from a_M to the i-th question token, and $\alpha^{l}_{h,j}$ is the corresponding weight to the j-th answer token. The attention contribution (AC) for head h at layer l is the fraction of attention directed to the question:

$$\text{AC}^{l,h} = \frac{C^{l,h}\,(\text{question})}{C^{l,h}\,(\text{question}) + C^{l,h}\,(\text{answer})} \tag{8}$$

We collect per-head scores for all layers into a single feature vector:

$$\text{AC} = [\text{AC}^{1,1}, \ldots, \text{AC}^{L,H}] \tag{9}$$

4.2 FFN and Residual Contribution

We extract the FFN and residual information flow contributions at the *first* answer token a_1, which captures how the model begins to generate the response before any answer tokens influence the context. As described in Equation (5), the FFN sub-layer decomposes the final layer output into two additive edges: the FFN edge $\text{FFN}_l(x^{lA}_{a_1})$ and the residual edge $x^{lA}_{a_1}$ (i.e., the output of the attention sub-layer carried forward). Applying the importance measure from Equation (1) to this decomposition, the FFN contribution score and residual contribution score at layer l are:

$$\text{FC}_l = \text{importance}\left(\text{FFN}_l(x^{lA}_{a_1}),\ x^{l}_{a_1}\right), \tag{10}$$

$$\text{RC}_l = \text{importance}\left(x^{lA}_{a_1},\ x^{l}_{a_1}\right). \tag{11}$$

Here RC_l measures the relative contribution of the attention sub-layer's output $x^{lA}_{a_1}$ to the final layer representation $x^{l}_{a_1}$—it should therefore be interpreted as the degree to which the layer's attention processing is carried forward, rather than the contribution of the raw input $x^{l-1}_{a_1}$. Since FC_l and RC_l

are derived from a two-way decomposition, they satisfy $\mathrm{FC}_l + \mathrm{RC}_l = 1$ (up to normalization), and together quantify the balance between parametric knowledge injection (FFN) and attention output carry-over (residual) at each layer. We concatenate these scores across all L layers to form the feature vectors:

$$\mathrm{FC} = [\mathrm{FC}_1, \cdots, \mathrm{FC}_L], \tag{12}$$

$$\mathrm{RC} = [\mathrm{RC}_1, \cdots, \mathrm{RC}_L]. \tag{13}$$

4.3 Empirical Motivation

As a preliminary validation, we compute the per-layer AUROC of each raw feature type (AC, FC, RC) without any probe training (see Figure 4 and Section 5.5). All three features show meaningful discriminative power in the intermediate layers, confirming that information flow patterns carry inherent signal for hallucination detection and motivating their use as probe features.

4.4 Probe Definition

To use these information flow contributions in detecting hallucination, we concatenate the attention contributions, FFN contributions, and residual contributions into a feature vector:

$$z = [\mathrm{AC}, \mathrm{FC}, \mathrm{RC}] \tag{14}$$

We employ a logistic regression classifier LR to predict whether the answer is a hallucination based on the information flow score vector z:

$$P(y = 1 \mid z) = LR(z) = \sigma\left(w^\top z + b\right) \tag{15}$$

where σ denotes the sigmoid function, w is the weight vector, and b is the bias term of the classifier. The full procedure for computing z is summarized in Algorithm 1.

Algorithm 1 Computing IFP Feature Vector

Require: $Q = \{q_1, \ldots, q_N\}$, $A = \{a_1, \ldots, a_M\}$
Require: LLM with L layers and H heads per layer
Ensure: Feature vector z

```
let imp(·,·) denote the importance function (Eq. 1)
for l = 1, ..., L do
    for h = 1, ..., H do
        C^{l,h}(Q) ← Σ_{i=1}^{N} α^l_{h,i}                ▷ at a_M
        C^{l,h}(A) ← Σ_{j=1}^{M} α^l_{h,j}                ▷ at a_M
        AC^{l,h} ← C^{l,h}(Q) / (C^{l,h}(Q) + C^{l,h}(A))
    end for
    AC_l ← [AC^{l,1}, ..., AC^{l,H}]
    FC_l ← imp(FFN_l(x^{lA}_{a_1}), x^l_{a_1})            ▷ at a_1
    RC_l ← imp(x^{lA}_{a_1}, x^l_{a_1})                   ▷ at a_1
end for
AC ← [AC_1, ..., AC_L]
FC ← [FC_1, ..., FC_L]
RC ← [RC_1, ..., RC_L]
return z ← [AC, FC, RC]
```

5 Experiments

5.1 Experimental Setting

Models We conduct experiments on various hallucination detection methods across three distinct series of open-source models: Llama3-8B-Instruct (Grattafiori et al., 2024), Qwen3-8B (Yang et al., 2025), and Gemma-2-9B-it (Team et al., 2024).

Datasets We evaluate various hallucination detection methods across multiple datasets: **HaluEval** (Li et al., 2023), evaluating the performance of LLMs in recognizing hallucination; **SQuAD** (Rajpurkar et al., 2018) comprising reading comprehension questions; **TriviaQA** (Joshi et al., 2017), which comprises general domain questions; **Natural Questions** (**NQ**) (Kwiatkowski et al., 2019), which comprises questions in search engines; **PopQA** (Mallen et al., 2023), which comprises entity-centric QA pairs from Wikidata; **SciQ** (Welbl et al., 2017), which comprises science exam questions across multiple disciplines; and **MovieQA** (Orgad et al., 2025), which comprises questions about factual knowledge about movies.

Baselines We compare hallucination detection with two types of methods. The training-free methods: **PPL** (Ren et al., 2023) and **LN-Entropy** (Malinin and Gales, 2021). The training-based methods: **SAPLMA** (Azaria and Mitchell, 2023b), **Mass Mean Probe (MM)** (Marks and Tegmark, 2024) and **ICR-Probe** (Zhang et al., 2025b).

Evaluation For the binary classification task of hallucination detection, we evaluate performance using a widely adopted metric: area under the receiver operating characteristic curve (AUROC). We employ few-shot prompts to generate responses. We utilize LLM-as-a-judge to evaluate the correctness of answers, specifically whether the LLM ex-

Model	Method	HaluEval	MovieQA	NQ	PopQA	SciQ	SQuAD	TriviaQA
Gemma-2	PPL	0.505	0.668	0.503	0.606	0.610	0.568	0.604
	LN-Entropy	0.610	0.658	0.523	0.676	0.615	0.510	0.678
	SAPLMA	0.744	0.707	0.735	0.863	0.764	0.725	0.807
	MM	0.750	0.726	0.736	0.859	0.736	0.679	0.801
	ICR-Probe	0.719	0.655	0.646	0.571	0.574	0.666	0.677
	IFP (Ours)	**0.836**	**0.757**	**0.803**	**0.900**	**0.818**	**0.765**	**0.868**
Qwen3	PPL	0.506	0.617	0.561	0.534	0.576	0.546	0.599
	LN-Entropy	0.614	0.764	0.575	0.590	0.591	0.615	0.622
	SAPLMA	0.841	0.858	0.775	0.930	0.788	0.724	0.777
	MM	0.854	0.818	0.797	**0.946**	0.779	0.697	0.853
	ICR-Probe	0.801	0.779	0.626	0.601	0.621	0.648	0.713
	IFP (Ours)	**0.895**	**0.884**	**0.845**	0.941	**0.849**	**0.800**	**0.898**
Llama3	PPL	0.582	0.713	0.671	0.534	0.650	0.578	0.622
	LN-Entropy	0.624	0.770	0.625	0.729	0.605	0.683	0.709
	SAPLMA	0.816	0.749	0.678	0.878	0.803	0.738	0.883
	MM	0.791	0.726	0.686	0.860	0.785	0.704	0.868
	ICR-Probe	0.762	0.673	0.590	0.545	0.691	0.667	0.724
	IFP (Ours)	**0.864**	**0.833**	**0.777**	**0.898**	**0.849**	**0.798**	**0.888**

Table 1: Main results (AUROC) across three LLMs and seven QA datasets. Best results are in **bold**.

hibits hallucinations.

5.2 Main Results

As shown in Table 1, IFP consistently outperforms all baselines across all three LLMs and seven QA datasets. The advantage over ICR-Probe—which distills each layer's attention–FFN interaction into a single scalar—confirms that retaining the full per-head attention routing structure provides substantially richer discriminative information. SAPLMA and MM, which rely solely on static hidden-state snapshots without modeling how representations evolve across layers, also fall short, highlighting the value of dynamic, flow-based features.

5.3 Generalization Analysis

To evaluate the generalization ability of hallucination detectors, we conduct cross-dataset experiments where the probe is trained on one dataset and evaluated on the remaining datasets. The results are shown in Figure 3. Overall, the proposed Information Flow Probe (IFP) consistently outperforms the ICR Probe across nearly all training–testing combinations. Specifically, IFP achieves AUROC scores ranging from 0.731 to 0.941, whereas ICR Probe typically falls between 0.554 and 0.801. This performance gap is particularly evident when transferring to knowledge-intensive datasets such as NQ and PopQA, where IFP maintains strong detection capability while ICR Probe experiences a substantial performance drop. Another important observation is the stability of cross-domain transfer. When trained on different datasets, IFP maintains relatively consistent performance across test datasets. For instance, training on TriviaQA yields AUROC scores above 0.76 on all other datasets, while training on PopQA produces strong results across both open-domain QA datasets and hallucination benchmarks such as HaluEval. In contrast, ICR Probe shows larger performance variance, suggesting that its learned representations are more dataset-specific.

5.4 Ablation Studies

Ablation Study 1: Contribution of Each Component. As defined in Equation (14), IFP consists of three components: AC (attention contribution), FC (FFN contribution), and RC (residual contribution). The ablation results are shown in Table 2. Removing AC causes the largest performance drop, confirming that per-head attention routing carries the dominant discriminative signal—a finding consistent with the high per-head AUROC observed in Ablation Study 3. FC and RC individually yield moderate performance (AUROC ≈ 0.69), showing

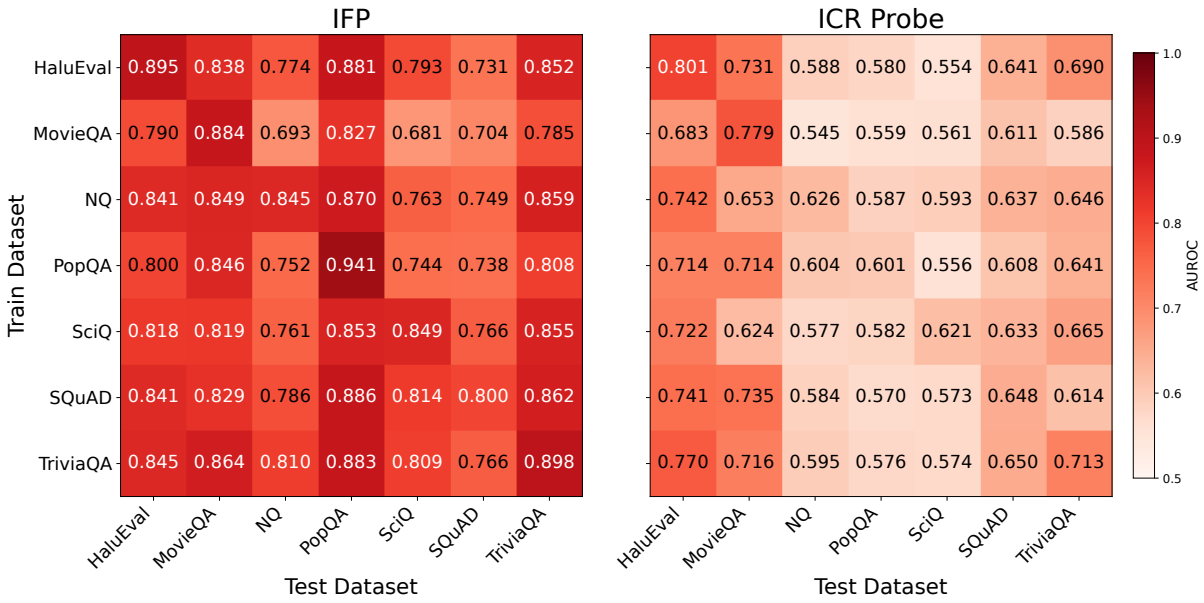

Figure 3: Cross-dataset AUROC heatmap for Qwen3-8B: the probe is trained on the row dataset and evaluated on the column dataset. Higher AUROC is better.

Setting	Feature Dim	Train AUROC	Test AUROC
IFP	1088	0.9087	0.8882
-w/o resid	1056	0.8937	0.8844
-w/o MLP	1056	0.8936	0.8840
-w/o attention	64	0.7216	0.6905
Merged attention	96	0.7868	0.7742

Table 2: Ablation study using the Llama3-8B-Instruct on the TriviaQA dataset, comparing feature dimension and the resulting AUROC on train and test splits. Merged attention denotes computing attention without distinguishing individual heads.

Layers	HaluEval	PopQA	TriviaQA
Layer 1-4	0.783	0.789	0.739
Layer 5-8	0.788	0.829	0.750
Layer 9-12	**0.820**	0.864	0.811
Layer 13-16	0.816	0.865	**0.841**
Layer 17-20	0.818	**0.877**	0.827
Layer 21-24	0.817	0.856	0.814
Layer 25-28	0.791	0.863	0.790
Layer 29-32	0.808	0.850	0.827
Layer 1-32	0.864	0.898	0.888

Table 3: AUROC results from layer-wise ablation experiments on three datasets using Llama3-8B-Instruct. Best sublayers are **bold**.

that FFN and residual features are informative on their own, though less powerful than AC. Replacing multi-head AC with a single-head (head-averaged) variant also causes a substantial performance decline, further validating that head-level granularity is essential rather than redundant.

Ablation Study 2: Contribution of Layers. To identify which layers contribute most to hallucination detection, we run layer ablation experiments that restrict features to four consecutive layers per setting. As shown in Table 3, the optimal layer range varies across datasets, but intermediate layers consistently outperform both early and late layers. On PopQA, for instance, layers 17–20 alone achieve AUROC only 0.021 below the full-model result. This finding aligns with prior work (Zhang et al., 2025b) suggesting that intermediate layers are the primary locus of factual knowledge processing.

Ablation Study 3: Attention Head Informativeness and Efficiency. Every attention head's AC score yields above-chance AUROC as a standalone classifier, confirming a consistent pattern: when the model answers correctly, the last answer token attends more heavily to the question tokens; when it hallucinates, it attends more heavily to its own previously generated answer tokens. A plausible interpretation is that during hallucination the model lacks sufficient question-relevant knowledge

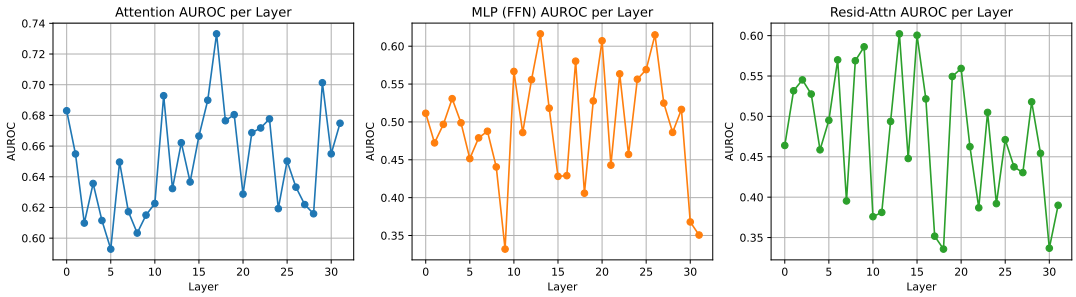

Figure 4: Per-layer AUROC of raw information flow features (attention contribution AC, FFN contribution FC, and residual contribution RC) without probe training, shown for Llama3-8B-Instruct on TriviaQA.

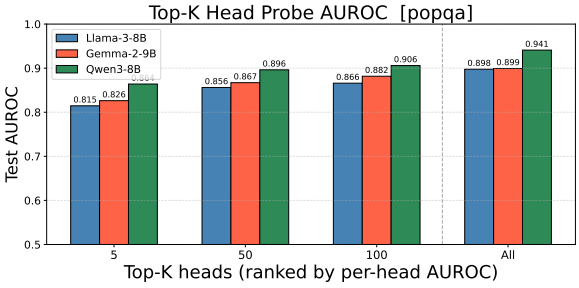

Figure 5: Probe AUROC as a function of the number of top-ranked attention heads included in the AC feature (ranked by individual AUROC). Performance saturates at around 100 heads (∼10% of all heads), indicating that a small subset suffices for near-optimal detection.

in its parametric memory, causing it to rely on the answer context it has already produced rather than on the original question. As shown in Figure 5, probe AUROC rises quickly when the top-ranked heads are included and plateaus at around 100 heads—roughly 10% of all heads in a 32-layer, 32-head model—already matching near-full performance. This suggests that IFP is robust to head selection: in resource-constrained settings, one could reduce feature dimensionality by approximately 90% with minimal performance cost.

5.5 Information Flow Pattern Analysis

Beyond trained probe performance, we examine whether individual information flow features carry discriminative signals in isolation. As shown in Figure 4, we compute the per-layer AUROC of each feature type (AC, FC, RC) using the raw scalar values directly as classifiers, without any probe training. AC features yield above-chance AUROC at every layer and peak in the intermediate layers. The underlying reason is intuitive: when a model retrieves a fact confidently, the last token attends heavily to the question; when it hallucinates, it instead recycles context from its own previously generated tokens—resulting in systematically lower AC values. FC and RC features show layer-dependent patterns, with discriminative power alternating in sign across layers, yet remaining meaningful in the middle layers. Taken together, these observations confirm that information flow patterns carry inherent signal for distinguishing hallucinations from factual responses, independent of any learned classifier.

6 Conclusion

We presented the Information Flow Probe (IFP), a lightweight logistic regression classifier for hallucination detection that operates on three complementary information flow features—per-head attention contribution (AC), FFN contribution (FC), and residual contribution (RC)—extracted at every layer via an attribution-based decomposition of the residual stream. A key finding is that these features carry inherent discriminative signal even without classifier training: factual answers and hallucinations exhibit systematically different information flow patterns across layers, with the last answer token attending more heavily to question tokens during correct generation and more to self-generated context during hallucination. By retaining full per-head granularity rather than collapsing attention into a coarse scalar, IFP captures richer routing information than prior dynamic probes such as ICR-Probe. Extensive experiments across three LLMs and seven datasets confirm consistent state-of-the-art performance and strong cross-dataset generalization. We hope this work encourages further use of mechanistic interpretability tools as a principled foundation for hallucination detection and mitigation.

Limitations

Indistinguishable hallucinations. Some hallucinated content is encoded with the same internal representations as factual knowledge within LLMs—for example, confidently memorized but incorrect facts or systematic biases baked in during pre-training (Guo and Li, 2026). In such cases, the information flow patterns at inference time are indistinguishable from those of correct answers, and no probe operating on a single forward pass can reliably detect them.

Computational overhead. Compared to probe methods such as SAPLMA and MM that extract only a single hidden-state vector, IFP requires collecting per-head attention weights, FFN outputs, and residual vectors at every layer. This increases memory consumption roughly proportionally to the number of layers and attention heads. Although the feature extraction remains efficient (one forward pass), the overhead may be non-trivial for very large models or high-throughput serving scenarios.

Restricted to open-source models. IFP relies on full access to intermediate activations—attention weights, FFN outputs, and residual streams—which are not exposed by closed-source API-only models such as GPT-5, Gemini-3. This limits applicability to open-weight models where internal representations can be inspected directly.

References

Aisha Alansari and Hamzah Luqman. 2025. Large language models hallucination: A comprehensive survey. *Preprint*, arXiv:2510.06265.

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

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

Javier Ferrando, Gerard I. Gállego, and Marta R. Costa-jussà. 2022. Measuring the mixing of contextual information in the transformer. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8698–8714, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Javier Ferrando, Oscar Balcells Obeso, Senthooran Rajamanoharan, and Neel Nanda. 2025. Do i know this entity? knowledge awareness and hallucinations in language models. In *The Thirteenth International Conference on Learning Representations*.

Javier Ferrando and Elena Voita. 2024. Information flow routes: Automatically interpreting language models at scale. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17432–17445, Miami, Florida, USA. Association for Computational Linguistics.

Cheng Gao, Huimin Chen, Chaojun Xiao, Zhiyi Chen, Zhiyuan Liu, and Maosong Sun. 2025. H-neurons: On the existence, impact, and origin of hallucination-associated neurons in llms. *Preprint*, arXiv:2512.01797.

Daniela Gottesman and Mor Geva. 2024. Estimating knowledge in large language models without generating a single token. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 3994–4019, Miami, Florida, USA. Association for Computational Linguistics.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Anxin Guo and Jingwei Li. 2026. Hallucination is a consequence of space-optimality: A rate-distortion theorem for membership testing. *Preprint*, arXiv:2602.00906.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

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.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Junyi Li, Xiaoxue Cheng, Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2023. HaluEval: A large-scale hallucination evaluation benchmark for large language

models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6449–6464, Singapore. Association for Computational Linguistics.

Qing Li, Jiahui Geng, Zongxiong Chen, Derui Zhu, Yuxia Wang, Congbo Ma, Chenyang Lyu, and Fakhri Karray. 2025. HD-NDEs: Neural differential equations for hallucination detection in LLMs. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6173–6186, Vienna, Austria. Association for Computational Linguistics.

Andrey Malinin and Mark Gales. 2021. Uncertainty estimation in autoregressive structured prediction. In *International Conference on Learning Representations*.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9802–9822, Toronto, Canada. Association for Computational Linguistics.

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

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

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

Daking Rai, Yilun Zhou, Shi Feng, Abulhair Saparov, and Ziyu Yao. 2025. A practical review of mechanistic interpretability for transformer-based language models. *Preprint*, arXiv:2407.02646.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

Jie Ren, Jiaming Luo, Yao Zhao, Kundan Krishna, Mohammad Saleh, Balaji Lakshminarayanan, and Peter J Liu. 2023. Out-of-distribution detection and selective generation for conditional language models. In *The Eleventh International Conference on Learning Representations*.

Pranab Sahoo, Prabhash Meharia, Akash Ghosh, Sriparna Saha, Vinija Jain, and Aman Chadha. 2024. A comprehensive survey of hallucination in large language, image, video and audio foundation models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 11709–11724, Miami, Florida, USA. Association for Computational Linguistics.

Ben Snyder, Marius Moisescu, and Muhammad Bilal Zafar. 2024. On early detection of hallucinations in factual question answering. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '24, page 2721–2732, New York, NY, USA. Association for Computing Machinery.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, Johan Ferret, Peter Liu, Pouya Tafti, Abe Friesen, Michelle Casbon, Sabela Ramos, Ravin Kumar, Charline Le Lan, Sammy Jerome, and 179 others. 2024. Gemma 2: Improving open language models at a practical size. *Preprint*, arXiv:2408.00118.

Johannes Welbl, Nelson F. Liu, and Matt Gardner. 2017. Crowdsourcing multiple choice science questions. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 94–106, Copenhagen, Denmark. Association for Computational Linguistics.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, Longyue Wang, Anh Tuan Luu, Wei Bi, Freda Shi, and Shuming Shi. 2025a. Siren's song in the ai ocean: A survey on hallucination in large language models. *Computational Linguistics*, 51(4):1373–1418.

Zhenliang Zhang, Xinyu Hu, Huixuan Zhang, Junzhe Zhang, and Xiaojun Wan. 2025b. ICR probe: Tracking hidden state dynamics for reliable hallucination detection in LLMs. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 17986–18002, Vienna, Austria. Association for Computational Linguistics.

A Prompts

We prompt LLMs to answer questions using a 6-shot prompt.

ICL Prompt

Answer the following questions as briefly as possible.

Question: `{demo question 1}`
Answer: `{demo answer 1}`

Question: `{demo question 2}`
Answer: `{demo answer 2}`

...

Question: `{input question}`
Answer:

We use DeepSeek-V3.2 as the LLM-as-a-judge, with the following prompt. We manually reviewed the results of LLM Judge on 500 QA samples, and its error rate was 0.8%.

LLM judge Prompt

We are assessing the quality of answers to the following question: *input question*
The following are expected answers to this question: *input ground truth*
The proposed answer is *proposed answer*
Within the context of the question, does the proposed answer mean the same as the expected answer?
Respond only with yes or no.
Here are some examples:
Question: *demo question 1*
Expected answer: *demo ground truth 1*
Proposed answer: *demo correct answer 1*
Response: *yes*

Question: *demo question 2*
Expected answer: *demo ground truth 2*
Proposed answer: *demo wrong answer 2*
Response: *no*

...

Now evaluate the following:
Question: *input question*
Expected answer: *input ground truth*
Proposed answer: *input proposed answer*
Response:

B Experiment Details

B.1 Implementation Details

We implement the logistic regression model using the scikit-learn library [1]. We conduct experiments on an Nvidia L40 GPU. We use five different random seeds and report the average value.

B.2 Model Details

- **Llama3-8B-Instruct** (Grattafiori et al., 2024): A decoder-only model with 32 layers, 32 query heads, and 8 key-value heads (grouped query attention, GQA), with a hidden dimension of 4096. HuggingFace ID: `meta-llama/Meta-Llama-3-8B-Instruct`.

- **Qwen3-8B** (Yang et al., 2025): A decoder-only model with 36 layers, 32 query heads, and 8 key-value heads (GQA), with a hidden dimension of 4096. HuggingFace ID: `Qwen/Qwen3-8B`.

- **Gemma-2-9B-it** (Team et al., 2024): A decoder-only model with 42 layers, 16 query heads, and 8 key-value heads (GQA), with a hidden dimension of 3584. Gemma-2 uses interleaved local (sliding-window) and global attention layers. HuggingFace ID: `google/gemma-2-9b-it`.

B.3 Dataset Details

The details of the dataset used in this paper are as follows:

- **Halueval**: 10000 samples from QA subset. Apache 2.0 license. https://huggingface.co/datasets/pminervini/HaluEval

- **MovieQA**: 10000 samples. MIT license. https://github.com/technion-cs-nlp/LLMsKnow/blob/main/data/movie_qa_train.csv

- **NQ**: sampled 10000 samples from the training set. CC-BY-SA-3.0 license. https://huggingface.co/datasets/google-research-datasets/nq_open

- **PopQA**: sampled 10000 samples from the test set. MIT license. https://huggingface.co/datasets/akariasai/PopQA

[1] https://scikit-learn.org

- **SciQ**: sampled 10000 samples from the training set. CC-BY-SA-3.0 license. https://huggingface.co/datasets/allenai/sciq
- **Squad**: sampled 5000 samples from the validation set. CC BY-SA 4.0 license. https://huggingface.co/datasets/rajpurkar/squad_v2
- **TriviaQA**: sampled 10000 samples from the validation set. Apache 2.0 license. https://huggingface.co/datasets/mandarjoshi/trivia_qa

B.4 Baseline Details

- **PPL** (Ren et al., 2023): Uses the perplexity of the generated answer sequence as a training-free uncertainty signal. Perplexity is computed as the exponentiated mean negative log-likelihood over the answer tokens; a higher value indicates lower model confidence and is treated as a hallucination indicator.
- **LN-Entropy** (Malinin and Gales, 2021): Computes the mean token-level entropy of the output distribution over the generated response, normalized by sequence length. This training-free method treats high entropy as indicative of uncertain, potentially hallucinated content.
- **SAPLMA** (Azaria and Mitchell, 2023b): Trains a logistic regression probe on the hidden states extracted from the last generated token. In our evaluation we concatenate hidden states across all L layers to give SAPLMA the same feature coverage as IFP (resulting in a $d \times L$-dimensional input vector, where d is the model's hidden size).
- **Mass Mean Probe (MM)** (Marks and Tegmark, 2024): Identifies a linear truthfulness direction in the hidden-state space by computing the difference between the mean representations of correct and incorrect examples (the "mass mean" direction). The projection of a test example onto this direction serves as the hallucination score. As with SAPLMA, we use all-layer concatenated hidden states of the last token for a fair comparison.
- **ICR-Probe** (Zhang et al., 2025b): At each layer, computes the Jensen-Shannon divergence between the head-averaged attention distribution and the direction of the residual stream update ($\Delta x^l = x^l - x^{l-1}$), yielding one scalar per layer. The resulting L-dimensional vector is fed to a multi-layer perceptron (MLP) classifier for hallucination detection.

For SAPLMA and MM, which originally use the hidden state of a single layer or the last token, we follow the same setting as IFP and use concatenated hidden states from the last token across all layers to ensure a fair comparison under equal feature coverage.