

# Do Androids Know They’re Only Dreaming of Electric Sheep?

Sky CH-Wang<sup>o\*</sup> Benjamin Van Durme<sup>•</sup> Jason Eisner<sup>•</sup> Chris Kedzie<sup>•</sup>

<sup>o</sup>Department of Computer Science, Columbia University

<sup>•</sup>Microsoft

skywang@cs.columbia.edu, chriskedzie@microsoft.com

## Abstract

We design probes trained on the internal representations of a transformer language model to predict its hallucinatory behavior on three grounded generation tasks. To train the probes, we annotate for span-level hallucination on both sampled (organic) and manually edited (synthetic) reference outputs. Our probes are narrowly trained and we find that they are sensitive to their training domain: they generalize poorly from one task to another or from synthetic to organic hallucinations. However, on in-domain data, they can reliably detect hallucinations at many transformer layers, achieving 95% of their peak performance as early as layer 4. Here, probing proves accurate for evaluating hallucination, outperforming several contemporary baselines and even surpassing an expert human annotator in response-level detection F1. Similarly, on span-level labeling, probes are on par or better than the expert annotator on two out of three generation tasks. Overall, we find that probing is a feasible and efficient alternative to language model hallucination evaluation when model states are available.

## 1 Introduction

*Do language models know when they’re hallucinating?*<sup>1</sup> Detecting hallucinations in grounded generation tasks (such as document summarization) is commonly framed as a textual entailment problem (Ji et al., 2023), with prior work largely focused on creating or applying secondary detection models trained on and applied to surface text (see Falke et al. (2019); Huang et al. (2020); Kryscinski et al. (2020); Goyal and Durrett (2020); Mishra et al. (2021); and others). In this work, we explore the degree to which the generation model itself—a decoder-only transformer—*already* encodes the desired diagnostic information in its feed-forward

\* Work performed as an intern at Microsoft Semantic Machines. Code and data available at [https://github.com/microsoft/llm\\_generation\\_probes](https://github.com/microsoft/llm_generation_probes).

<sup>1</sup>That is, *do they know their electric sheep are ungrounded?*

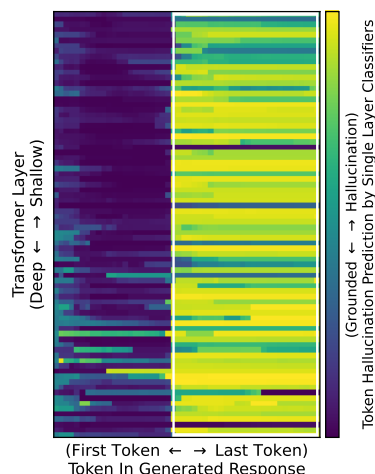


Figure 1: Prediction of hallucination in one generated response by probing the hidden states of a transformer during decoding. The true annotated span of hallucination within the response is boxed in white; rows represent probes trained to detect hallucination from different transformer layers, with columns representing their prediction for each token in a generated response.

and self-attention output states. We train span-level probes supervised at the token-level and response-level probes for hallucination on several grounded generation tasks and show that hallucination with respect to grounding data is detectable from an transformer’s hidden states, in line with similar results on probing ungrounded (i.e., closed-book QA) generation (Mielke et al., 2022; Azaria and Mitchell, 2023).

As a language model generates a response, we apply our probes to its encoded tokens to determine whether the current token is likely to be a hallucination. Related work has argued that these encodings are full of meta-information about the model’s generation process. For example, salient information for generation can be localized to specific model layers and time steps (Geva et al., 2021; Meng et al., 2022, 2023), and hidden states can encode the model’s confidence about the correctness

of subsequent response generation (Mielke et al., 2022). This suggests that a probe on model hidden states will have access to a rich set of features for detecting hallucination—features that would not be directly and efficiently available to a secondary entailment or classification model seeing only the input and generated output tokens.

We develop three probes of increasing complexity across three grounded generation tasks: abstractive summarization, knowledge-grounded dialogue generation, and data-to-text. For each task, we collect hallucinations in two ways: (1) from ordinary sampled responses, where we generate outputs from a large language model (LLM) conditioned on the inputs, or (2) by editing reference inputs or outputs to create discrepancies. We refer to these as organic and synthetic data, respectively. In both cases, we have human annotators mark hallucinations in the outputs. Method (2) produces hallucination annotations at a higher rate, though we find that the utility of these synthetic examples is lower as they do not come from the test distribution.

We summarize our contributions as follows:

- We produce a high-quality dataset of more than 15k utterances with hallucination annotations for organic and synthetic output texts across three grounded generation tasks.
- We propose three probe architectures for detecting hallucinations and demonstrate improvements over multiple contemporary baselines while also surpassing an expert human annotator on response-level detection F1. On two out of three tasks, the best probe achieves over 90% F1.
- We show probe performance on span-level detection is competitive with an expert human annotator, performing on par or better on two out of three generation tasks in our dataset.
- We analyze how probe accuracy is affected by annotation type (synthetic/organic), hallucination type (extrinsic/intrinsic), model size, and which part of the encoding is probed.

## 2 Related Work

**Definitions.** We center our study of hallucinations in the setting of in-context generation (Lewis et al., 2020) where grounding knowledge sources are provided within the prompt itself. This is in contrast to settings where a language model is expected to generate a response entirely from its parametric

knowledge (i.e., grounded in the model’s training data, as in Azaria and Mitchell (2023)). Though the term has often appeared in reference to factuality (Gabriel et al., 2021), hallucinations—as we use the term—are not necessarily factually incorrect, only ungrounded. We follow the taxonomy of Ji et al. (2023): in their survey, they classify hallucinations as either *intrinsic*, where generated responses directly contradict the knowledge sources, or *extrinsic*, where generated responses are neither entailed nor contradicted by the sources. We follow this taxonomy. For example, given the following task prompt for abstractive summarization,

... Deckard is now able to buy his wife Iran an authentic Nubian goat—from Animal Row, in Los Angeles—with his commission ... TL;DR:

a generated summary sentence of “*Deckard can now purchase an animal for his wife*” is faithful, “*Deckard can purchase a goat in Iran*” is an intrinsic hallucination, and “*A Nubian goat is a British domestic goat breed*” is an extrinsic hallucination. While models may leverage latent knowledge to *inform* their responses, such as recognizing that a Nubian goat is indeed a British domestic goat breed, directly *stating* such knowledge in a response—as in the last sentence above—is considered an extrinsic hallucination when it is neither entailed nor contradicted by the knowledge sources. Thus, non-hallucinatory generation in this setting involves balancing the language model’s retrieval and manipulation of knowledge, both from the given prompt context and from its parameters.

**Automated Diagnosis.** Lexical metrics commonly used in the evaluation of NLG fail to correlate with human judgments of faithfulness (Kryscinski et al., 2019). More useful are natural language inference (NLI) approaches that either directly measure the degree of entailment between a generated response and a given source text (Hanselowski et al., 2018; Kryscinski et al., 2020; Goyal and Durrett, 2020), or else break down texts into single-sentence claims that are easily verifiable (Min et al., 2023; Semnani et al., 2023). These NLI methods have been successfully used for hallucination detection (Laban et al., 2022; Bishop et al., 2024). Similarly, approaches that leverage question-answer (QA) models to test if claims made by a response can be supported by the grounding text have also been applied to measure hallucination (Wang et al., 2020; Durmus et al.,

Task	History	Knowledge	Instruction	Response
<b>Abs. Summ.</b>	–	...dialled down that initial forecast to predict 13 to 19 named storms, including six to nine hurricanes...	TL;DR:	The National Weather Service forecast called for 13 to 19 named storms including six to <b>11</b> hurricanes.
<b>KGDG</b>	...B: ok how can I get a turkey I think that may be hard to get though	Three consecutive strikes are known as a “turkey.”	Response that uses the information in knowledge:	A: Indeed, you have to get <b>four</b> consecutive strikes to get one.
<b>D2T</b>		name[The Wrestlers], priceRange[high]	Restaurant Description:	<b>Located by the riverside.</b> The Wrestlers is a restaurant with high prices, <b>and near Raja Indian Cuisine.</b>

Table 1: From top to bottom, examples of organic hallucinations in model responses for **abstractive summarization**, **knowledge-grounded dialogue generation** (KGDG), and **data-to-text** tasks. Minimum annotated hallucination spans are highlighted in **cyan** for intrinsic hallucinations and **green** for extrinsic hallucinations.

2020; Scialom et al., 2021). To the extent that models are less confident when they are hallucinating, hallucination may also be reflected in metrics that measure generation uncertainty (Malinin and Gales, 2021; Guerreiro et al., 2023; Ren et al., 2023; Ngu et al., 2024; Madsen et al., 2023), relative token contributions to the overall output probability (Xu et al., 2023), or signatures in the attention matrices during generation (Lee et al., 2018; Yuksekgonul et al., 2023). Similarly, conflicts between contextual knowledge provided in a prompt and parametric knowledge as learned during pre-training can contribute to this behavior (Xu et al., 2024).

**Predicting Transformer Behavior.** In small transformer language models, it is possible to ascribe behaviors like in-context learning to specific layers and modules (Elhage et al., 2021). This becomes harder in larger models, but Meng et al. (2022) provide evidence that the feed-forward modules function as key-value stores of factual associations and that it is possible to modify weights in these modules to edit facts associated with entities. Transformer internal states have also been shown to be predictive of the likelihood that a model will answer a question correctly in the setting of closed-book QA (Mielke et al., 2022) and in the prediction of the truthfulness of a statement (Azaria and Mitchell, 2023). Interestingly, these internal representations of truthfulness can disagree systematically with measures of model confidence (Liu et al., 2023). These works measure hidden states in the absence of supporting evidence (i.e. no passages are retrieved for either task), making them effectively probes of knowledge stored in model

weights and not the manipulation of novel or contingent information provided in a prompt. In contrast, this work examines related questions specifically in the context of grounded generation (e.g., retrieval augmented generation (Lewis et al., 2020)).

Concurrently to this work, Monea et al. (2023) find that there are separate computational processes in transformer LLMs devoted to the recall of associations from memory versus grounding data. Slobodkin et al. (2023) train a probe to detect the (un)answerability of a question given a passage, which is similar in some respects to our task in §4; however, we detect span-level hallucination during answer generation, which could happen even in theoretically answerable questions.<sup>2</sup>

### 3 Grounded Generation Tasks

We test hallucination probes for autoregressive grounded generation in three distinct tasks: abstractive summarization, knowledge-grounded dialogue generation, and data-to-text. We use the same high-level prompt structure across all tasks to organically generate model responses to tasks and to probe for hallucination. Each prompt contains an optional interaction *history*, a *knowledge* source, and an *instruction* resulting in the conditionally generated *response*. Task prompts and examples of organic hallucinations are shown in Table 1.

#### 3.1 Organic Hallucinations

We use llama-2-13b (Touvron et al., 2023) as the response model to generate outputs for each task.

<sup>2</sup>Reviewers asked to distinguish our work from that of Azaria and Mitchell (2023) and Kadavath et al. (2022). For a detailed discussion of the differences, see Appendix D.

Response	Original Knowledge	Changed Knowledge
Yes, Elvis was also one of the <b>most famous musicians</b> of the 20th century.	Elvis Presley is regarded as one of the <b>most significant cultural icons</b> of the 20th century.	Elvis Presley is regarded as one of the <b>least well-known music-ians</b> of the 20th century.
Close to The Sorrento you can find Clowns <b>pub</b> .	name[Clowns], <b>eatType[pub]</b> near[The Sorrento]	name[Clowns], near[The Sorrento]

Table 2: Examples of changed knowledge statements for our datasets of synthetic hallucinations. Changes made to Conv-FEVER (top) and E2E (bottom) are shown in **bold**. The changed *knowledge* statements make the top *response* an **intrinsic** predicate error and the bottom *response* an **extrinsic** entity error (highlighted as in Table 1).

To ensure that a non-negligible number of both hallucinations and grounded responses were present in the generations, each model response is generated using top- $k$  random sampling (Fan et al., 2018) with  $k = 2$  and temperature 1. The number of few-shot in-context learning examples for each task (see below) was chosen through a pilot exploration, optimizing for the presence of a balance of hallucinations and fully grounded responses, as determined via manual inspection by the authors. Full prompt templates are shown in Appendix C.

**Abstractive Summarization.** The CNN / DailyMail (CDM) dataset (Hermann et al., 2015) is a standard benchmark in the setting of abstractive summarization (Lin and Ng, 2019). It consists of a set of news articles paired with reference summaries in the form of human-written bullet points that were originally published alongside the articles. For automatic summarization, we follow Radford et al. (2019) and prompt our model to generate summaries of news articles by appending TL;DR: after the article content (a 0-shot prompt),<sup>3</sup> taking the first three generated sentences as the summary.

**Knowledge-Grounded Dialogue Generation.** For a given conversation history and a provided knowledge sentence obtained via Wizard-Of-Wikipedia, the Conv-FEVER (Santhanam et al., 2021) task (CF) asks a respondent to create a next-turn response to the conversation that is grounded in the provided knowledge sentence. Here, we prompt our response model to generate grounded responses under 2-shot in-context learning with example grounded responses.

<sup>3</sup>In Internet posts that appear in LLM pretraining data, this string often marks the start of a summary immediately following the main content of the post. Prior work (Radford et al., 2019) has found that appending this string after texts in prompts to autoregressive LLMs is capable of producing abstractive summaries of the former.

**Data-to-Text Generation.** The E2E task (Novikova et al., 2017) asks respondents to write restaurant descriptions that are congruent with a provided set of attributes. Here, we prompt our response model to perform this task with 9-shot in-context learning, with in-context examples manually validated for groundedness.<sup>4</sup>

### 3.2 Synthetic Hallucinations

Recent work (Ma et al., 2023; Monea et al., 2023) has carefully created synthetic ungrounded datasets by hand to evaluate metrics used to measure unfaithfulness and models used to detect it. The BUMP dataset (Ma et al., 2023) is a version of CDM in which reference summaries have been manually edited to be ungrounded. It includes both freestyle edits and systematic coverage of four specific types of errors: (1) *predicate errors*, where the predicate in a generation is inconsistent with the source knowledge; (2) *entity errors*, where the subject or object of a predicate is inconsistent; (3) *circumstance errors*, or errors in the time, duration, or location of an event of the predicate; and (4) *coreference errors*. Types (1)–(3) may be intrinsic or extrinsic, while (4) may be ambiguous; as such, we don’t consider coreference errors in our analysis of intrinsic or extrinsic hallucinations.

We use BUMP in our CDM experiments. For our Conv-FEVER and E2E experiments, we similarly created errors in a uniformly sampled subset of these datasets. However, rather than changing an example’s reference response, we changed its provided knowledge sentence so that the existing reference response would be unfaithful to it. This simulates the common case of hallucination where

<sup>4</sup>As the original E2E task had omitted or missing information in up to 40% of its data, we use the cleaned version by Dušek et al. (2019). As even the cleaned version had instances of omitted or missing information, 9 in-context examples were manually selected and verified for groundedness.

Task	N-o	H-o	$\kappa$ -o	N-s	H-s
CDM	4684	28%	0.39	1778 <sup>6</sup>	50%
CF	924	70%	0.50	2846	50%
E2E	4998	54%	0.55	9950	50%

Table 3: **Dataset statistics.** **N-o** is the number of annotated organic generations, **H-o** is the percentage of organic generations with at least one hallucination in the final adjudicated dataset, and  $\kappa$ -o is the response-level Fleiss’s  $\kappa$  of individual annotations before adjudication. **N-s** and **H-s** refer to the same for synthetic responses. Note that grounded synthetic responses were not generated by a model but come from gold reference data.

a large language model generates statements that may be plausible but are not technically grounded.

For Conv-FEVER, we prompted ChatGPT to modify a given example’s provided knowledge sentence to introduce intrinsic and extrinsic errors of types (1)–(4) above; we then manually judged these attempts and edited the unsuccessful ones.<sup>5</sup> In this way, we created equal numbers of examples with intrinsic and extrinsic predicate, entity, circumstance, and coreference errors, as well as freestyle errors.

For E2E, we modified a random subset of the provided grounding attributes. Specifically, for an example with  $n$  attributes, we drew  $k$  uniformly from  $[1, n - 1]$ , uniformly selected one of the subsets of size  $k$ , and then flipped a fair coin to decide whether to remove or perturb—by randomly sampling an alternative attribute from the entire dataset—these  $k$  attributes. Finally, we manually verified that the reference response was unfaithful with respect to the modified attributes; if not, we further edited the attributes. Examples of changed knowledge statements are shown in Table 2; statistics of our synthetic datasets are shown in Table 3.

### 3.3 Annotation

We ask annotators to label the *minimal hallucinatory spans* in each response, if any. These are the smallest text spans that would need modification to transform an ungrounded response into a grounded one. A team of 17 annotators completed a pilot annotation exercise of 50 annotations per task. These pilot annotations were checked and verified by the authors to pinpoint common sources of error and disagreement. This feedback was integrated into a

<sup>5</sup>In hindsight, this workflow proved inefficient; significant human editing and validation were needed.

<sup>6</sup>Obtained via BUMP (Ma et al., 2023).

revised set of annotation guidelines, shown in Appendix A, and the full dataset was then re-annotated fully by three annotators such that each example was annotated three times (example annotations are shown in Table 1). Annotations were then reconciled at the token-level into the final gold annotation, such that a token is labeled as a hallucination if at least two out of three annotators included it in a hallucination-span annotation.

Annotation statistics are shown in Table 3. Note that for each dataset, over one-fifth of all annotated responses were annotated as containing at least one hallucinated span (columns **H-o** and **H-s**), which we refer to as response-level annotations. Annotators exhibited moderate agreement (Landis and Koch, 1977), achieving response-level Fleiss’  $\kappa$  scores between 0.39 and 0.55 across the three tasks.

## 4 Probing

Probes (Alain and Bengio, 2017) are tools used to analyze a neural network’s internal representations. Often taking the form of linear classifiers, they are applied to the internal representations of a network and are trained to discriminate between different types of inputs or outputs. Here, we are interested in detecting hallucinatory outputs during an LLM’s generation steps when conditioned on a prompt that requires a response grounded in the prompt.

### 4.1 Design

Let  $u$  be a prompt as described at the start of §3. To determine whether a response  $x$  to a prompt  $u$  contains hallucinations, we will probe the hidden states of the transformer language model (i.e., decoder) as it generates  $x$ . Note that these hidden states depend on  $u$  and on the prefix of  $x$  that has been generated so far, and they may be reconstructed from the string pair  $(u, x)$  via forced decoding. A single transformer layer actually consists of two sublayers (Vaswani et al., 2017): the first transforms a token’s embedding by adding a vector predicted by attention, and the second transforms it again by adding a vector predicted by a feed-forward network. We train a separate probe for each of the  $2L$  sublayers, where  $L$  is the number of layers.

**Linear Probe.** This probe is a linear classifier using a single hidden state vector (output by some sublayer) as input. Let  $x_i$  be the  $i^{\text{th}}$  response token and let  $\mathbf{h}_i \in \mathbb{R}^n$  be the corresponding LLM hidden state at a fixed layer and component. We write  $y_i \in \{0, 1\}$  to indicate whether  $x_i$  falls within a

minimum hallucination span. The linear probe predicts  $y_i$  by  $p(y_i = 1 \mid u, x_{\leq i}) = \sigma(\mathbf{w}^\top \mathbf{h}_i + b)$ , where  $\mathbf{w} \in \mathbb{R}^n$  are learned weights,  $b \in \mathbb{R}$  is a learned bias, and  $\sigma$  is the logistic function.

**Attention-Pooling Probe.** In order to directly consider the previous hidden states of the response as well, we also experiment with an attention-pooling probe over all hidden states so far,

$$p(y_i = 1 \mid u, x_{\leq i}) = \sigma(\mathbf{w}^\top \bar{\mathbf{h}}_i)$$

$$\bar{\mathbf{h}}_i = \sum_{j=1}^i \alpha_{i,j} \mathbf{h}_j, \quad \alpha_{i,j} = \frac{\exp(\mathbf{q}^\top \mathbf{h}_j)}{\sum_{k=1}^i \exp(\mathbf{q}^\top \mathbf{h}_k)}$$

where  $\mathbf{q} \in \mathbb{R}^n$  is a learned query vector.

**Ensemble Probe.** Our final approach is a logistic regression probe,  $p_{\text{ens}}(y_i \mid u, x_{\leq i}) = \sigma\left(\sum_{l,m} \beta_{l,m} \cdot p_{l,m}(y_i \mid u, x_{\leq i})\right)$ , that combines the predictions from  $2L$  separately trained probes indexed by layer  $l \in \{1, \dots, L\}$  and module  $m \in \{\text{feed-forward, attention}\}$ . The weights  $\beta_{l,m}$  are learned after training and freezing each  $p_{l,m}$ .

## 4.2 Training Objectives

**Token-level.** To train a probe to predict *token-level* hallucination labels  $y_i$ , we fit probe parameters to minimize the negative log-likelihood,  $\mathcal{L}(\theta) \propto \sum_{(u,x,y) \in \mathcal{D}} \sum_{i=1}^{|x|} -\log p(y_i \mid u, x_{\leq i}; \theta)$ , on the token-level annotations in our dataset  $\mathcal{D}$ .

**Response-level.** We also try training a classifier to predict whether  $x$  contains any hallucinations at all; denote this indicator as  $y = \bigvee_{i=1}^{|x|} y_i \in \{0, 1\}$ . For this, we simply use an attention-pooling probe over the entire response  $x$ . To fit the probe parameters,  $\theta$ , we again minimize negative log-likelihood,  $\mathcal{L}(\theta) \propto \sum_{(u,x,y) \in \mathcal{D}} -\log p(y \mid u, x_{\leq |x|}; \theta)$ .

## 4.3 Metrics

**Response-level Classification.** F1 is the harmonic mean of recall and precision. We denote the F1 score of a response-level classifier by **F1-R**.

**Span-level Classification.** For span-level classifiers, it is possible to evaluate the F1 score of their per-token predictions; however, this would give more weight to annotated or predicted spans that are longer in length. Here, we are more interested in *span-level* F1: specifically, whether we predicted all of the annotated hallucinations (recall), and whether all of the predicted spans

were annotated as actual hallucinations (precision). As requiring an exact match of spans is too severe—Appendix B shows how our human annotators often disagreed on the exact boundaries of a hallucinatory span—we modify span-level precision and recall metrics to give partial credit, as follows.

Define a span to be a set of tokens, where tokens from different responses are considered distinct. Let  $A$  be the set of all annotated spans in the reference corpus, and let  $P$  be the set of all predicted spans on the same set of examples. We compute the recall  $r$  as the average coverage of each reference span, i.e., the fraction of its tokens that are in some predicted span. Conversely, we compute the precision  $p$  as the average coverage of each predicted span. So  $r = \frac{1}{|A|} \sum_{s \in A} \frac{|s \cap (\bigcup_{\hat{s} \in P} \hat{s})|}{|s|}$  and  $p = \frac{1}{|P|} \sum_{\hat{s} \in P} \frac{|\hat{s} \cap (\bigcup_{s \in A} s)|}{|\hat{s}|}$  where  $\cup$ ,  $\cap$ , and  $|\cdot|$  denote set union, intersection, and cardinality respectively. We define **F1-S<sub>p</sub>** (standing for "span, partial credit") as the harmonic mean of  $p$  and  $r$ . Note that these quantities micro-average over spans, ignoring the boundaries between examples. Thus, in contrast to **F1-R**, examples with more spans have more influence on the final **F1-S<sub>p</sub>** metric achieved.

## 5 Experiments

Unless otherwise indicated, all evaluations are performed on organic data only, even if their probes were trained on synthetic data.

### 5.1 Probe Hyperparameters and Training

Single-layer hidden state probes, described in §4, are trained under float32 precision with unregularized log loss. Optimal batch sizes between 20-100 and Adam optimization (Kingma and Ba, 2014)—with learning rates between 0.1-0.001—and  $\beta_1, \beta_2 = (0.9, 0.999)$  were chosen for each task after a small grid search on hyperparameters. With 70/10/20% train, validation, and test splits, each classifier was trained using early stopping with 10-epoch patience. The ensemble classifier was trained with the same hyperparameters as the single-layer probes.

For CDM, the synthetic training dataset (from BUMP) is smaller than the organic training dataset, so the SYNTH results are at a disadvantage. For CF and E2E, however, we generated a synthetic training dataset that was matched in size.

When reporting results, *SL* denotes the best

single-layer classifier (chosen on the validation set),  $E$  denotes the ensembled classifier, and *SYNTH* indicates that the classifier was trained on synthetic data and tested on organic hallucinations.

## 5.2 Response-Level Baselines

We compare the performance of our probes against a set of contemporary methods used in faithfulness evaluation, measuring test set hallucination classification F1. Specifically: (1) the model uncertainty-based metric of length-normalized sequence log-probability **Seq-Logprob** (Guerreiro et al., 2023); (2) off-the-shelf synthetically-trained sentence-level claim classification **FactCC** (Kryscinski et al., 2020); (3) linguistic feature-based dependency arc entailment **DAE** (Goyal and Durrett, 2020); (4) QA-based metrics **FEQA** (Durmus et al., 2020); (5) **QuestEval** (Scialom et al., 2021); (6) the NLI-based metric **SummaC** (Laban et al., 2022); **ChatGPT** classifiers (Luo et al., 2023) gpt-3.5-turbo-0125 at zero temperature (7) without and (8) with basic chain-of-thought prompting (with prompts shown in Appendix C.1), as well as (9) the automatic claim-breaking and verification method **FActScore** (Min et al., 2023) with gpt-3.5-turbo-instruct.

Metrics (2)–(6) and (9) were not originally intended to evaluate individual responses, but rather a collection of responses for the sake of system comparisons. To adapt to our setting, however, we threshold each of these continuous metrics to classify the individual responses, setting the threshold to maximize validation set F1, following Laban et al. (2022). For sentence-level optimized baselines such as FactCC, response-level evaluation is performed by logical OR-ing sentence-level baseline predictions (obtained after thresholding the raw scores). In other words, any hallucination in any sentence leads to a "hallucinated" label. Note that, for our abstractive summarization tasks, methods such as FactCC (Kryscinski et al., 2020) are explicitly trained in the domain as they use CDM as the initial dataset for training. For certain baselines in CF and E2E, dashes are present in the results due to an inability to directly compare methods in these domains. For example, for DAE, one cannot obtain the dependency arcs from a set of restaurant attributes not expressed in natural language.

As a further reference baseline, we additionally report the performance achieved by randomly predicting a hallucination label with some probability

Response-Level Classification, F1-R			
Method	CDM	CF	E2E
OC	0.43	0.79	0.71
Seq-Logprob	0.64	0.81	0.72
FactCC	0.51	-	-
DAE	0.53	-	-
FEQA	0.44	-	-
QuestEval	0.61	-	-
SummaC <sub>ZS</sub>	0.68	-	-
SummaC <sub>Conv</sub>	0.67	-	-
ChatGPT	0.63	0.74	0.74
ChatGPT CoT	0.53	0.51	0.47
FActScore	0.61	0.80	0.72
Poolings <sub>SL</sub>	0.73	0.93	0.89
Pooling <sub>E</sub>	<b>0.75</b>	<b>0.94</b>	<b>0.90</b>
Poolings <sub>SL-SYNTH</sub>	0.65	0.80	0.68
Pooling <sub>E-SYNTH</sub>	0.66	0.83	0.71
Expert Human	0.55	0.89	0.79

Table 4: **Response-level** probe and baseline organic hallucination detection performance across tasks as measured via F1 scores (**F1-R**) achieved on the test set.

$p$  tuned to maximize F1 on the validation set, denoted here as Optimized Coin (**OC**).

## 5.3 Expert Human Judge

To compare our probe performance to a human performing the equivalent hallucination detection task, we had another professional annotator judge the test set. This annotator (denoted **Expert Human**) was distinct from the annotators who produced the gold labels. They were given access to the training data (consisting of all three sets of annotations plus the gold reconciled annotations) and performed a round of pilot annotations with feedback from the authors before annotating the test set.

## 5.4 Results

For performance comparisons, probe results are statistically significant ( $p < 0.05$ ) relative to respective baselines under exact paired-permutation testing (Zmigrod et al., 2022) unless otherwise noted.

**Response-level Results.** Response-level results are shown in Table 4. On all three tasks, the probes were able to surpass the Expert Human annotator on response-level classification, with the Pooling<sub>E</sub> probe outperforming the Expert Human by as much as 20 points absolute F1-R on the CDM dataset.

The CDM dataset is likely the most challenging for a human to judge as it requires checking the generated statements against the complete (and lengthy) news article. Both  $\text{Pooling}_{\text{SL}}$  and  $\text{Pooling}_{\text{E}}$  probes outperformed all baselines across all datasets. Ensembling generally adds at least one point in absolute F1-R above the best single layer. The probes trained on synthetic hallucinations generally do not perform as well as their organic equivalents, though they are still fairly competitive with the baselines. Seq-Logprob was surprisingly strong, beating both the Expert Human and all but one of the baselines on CDM. This suggests that while model confidence may not be the only feature related to hallucination detection, it is likely an important one.

**Span-Level Results.** These results appear in Table 5. The  $\text{Pooling}_{\text{E}}$  probes match or surpass the Expert Human performance on the CDM and E2E datasets while falling three points behind in absolute F1- $S_p$  on the Conv-FEVER dataset. Pooling information across time does help probe performance with the best single-layer pooling probe matching or beating the best single-layer linear probe across all datasets. Interestingly, what the linear probe lacks in time can be made up in depth— $\text{Linear}_{\text{E}}$  matches  $\text{Pooling}_{\text{SL}}$  on CDM and Conv-FEVER, and beats it by one point absolute F1- $S_p$  on E2E, though this wasn’t statistically significant.

## 6 Analysis

**Layers.** Evaluating the performance of trained single-layer attention-pooling probes (Figures 2 and 4), in model *middle layers* during decoding. Probe performance shows a consistent and gradual increase from the first layer after token embeddings up until layers 10 to 20. Hallucination information is reliably present at all layers  $\geq 5$ , although there are local optima around layers 5–10 and again at 40. Ensembling across many layers provides only a small (not stat. sig.) improvement over the best single layer (Table 4), so we do not have evidence that our probes at different layers detect different types or correlates of hallucination. In general, probe performance declines for subsequent layers after their initial peak layer across tasks, though CDM is an exception in that probe performance gradually increases again from layers 30 to 40, peaking there.

**Hidden State Type.** Slight variations in grounding saliency—the strength of the signal of hallucination as able to be detected by a probe, mea-

Span-level Classification, F1- $S_p$			
Method	CDM	CF	E2E
OC	0.31	0.75	0.13
$\text{Linear}_{\text{SL}}$	0.52	0.77	0.54
$\text{Linear}_{\text{E}}$	0.54	0.79	0.55
$\text{Pooling}_{\text{SL}}$	0.54	0.79	0.54
$\text{Pooling}_{\text{E}}$	<b>0.55</b>	<b>0.81</b>	<b>0.56</b>
Expert Human	0.40	0.84	0.56

Table 5: **Span-level** probe performance, as measured by span-level partial match scores (F1- $S_p$ ).

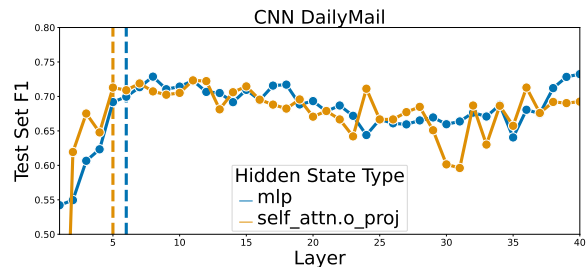


Figure 2: Grounding behavior saliency as measured by probe response-level F1 (F1-R) on the CDM test set. The  $x$ -axis corresponds to the layer of the probe. Plots across all tasks are shown in Appendix D and Figure 4. Vertical dashed lines denote the layers that respective probes first surpass 95% of their peak performance.

sured by probe hallucination classification performance—are also present among hidden states of different types during decoding. As shown in Figure 2, feed-forward probes reach peak performance at slightly deeper layers than their attention head out-projection probe counterparts and exceed them in single-layer hallucination detection. When training ensemble classifiers on only the hidden states of the same type at every layer, feed-forward hidden states possess a slightly greater & statistically significant saliency of model grounding behavior.

**Model Size.** Training probes in the same manner on the force-decoded hidden states of responses on our dataset using llama-2-7b and llama-2-70b, we find no statistically significant difference in hallucination detection across the hidden states from base models of varying sizes. These results show that we can still detect a hallucination even when it was originally generated from a different model, and that even the lower-dimensional hidden states of the 7B model still contain enough information to train a probe.



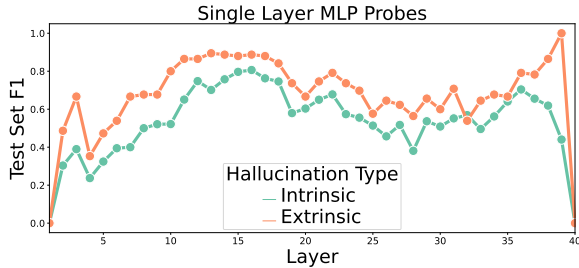


Figure 3: **Intrinsic** and **extrinsic** hallucination saliency across layers for synthetic CDM as measured by response-level MLP probe F1 (**F1-R**) on hallucination types; additional plots in Appendix D, Figure 5.

**Hallucination Type.** Next, we test for the saliency of different forms of hallucination behavior in model internal states. Here, with annotator-labeled test set responses according to whether they contained intrinsic or extrinsic hallucinations, we evaluate trained probe performance in correctly detecting the presence of hallucinations for instances of each hallucination type. Intrinsic hallucinations are rarer and may reflect a misunderstanding of the grounding data. We observe in Figure 3 that extrinsic hallucinations—which are  $1.8\times$  more common at the response level (as labeled by annotators) and presumably reflect a failure to find desired information in the grounding data at all—are easier to detect at every layer. Both kinds are generally most salient in probe middle layers, between layers 10 to 20. Overall, extrinsic hallucination remained more salient in hidden states than intrinsic hallucination.

**Synthetic Hallucinations.** As Table 6 shows, probes achieve very high F1 on the detection of synthetically created hallucinations across all tasks. However, probe performance is subject to a drop when trained and tested on different modalities, with probes generally performing better when trained and tested on the same hallucination modality. The most severe version of this is on CDM, which may be caused by differences between our organic annotation instructions and those of the BUMP dataset (Ma et al., 2023). Prior work has shown that synthetic data generation approaches specifically designed for factuality evaluation (Ma et al., 2023) do not align with actual errors made by generation models in their lexical error distributions (Goyal and Durrett, 2021). Here, our results show this to extend to their hidden state *signals* as well, which are used by the probe to detect unfaithfulness in the case of forced decoding on human-created content, relative to organic instances of

Task	Train	Test	F1-R
CDM	Organic	Organic	<b>0.75</b>
	Synthetic	Organic	0.66
	Organic	Synthetic	0.17
	Synthetic	Synthetic	<b>0.86</b>
CF	Organic	Organic	<b>0.94</b>
	Synthetic	Organic	0.83
	Organic	Synthetic	0.90
	Synthetic	Synthetic	<b>0.95</b>
E2E	Organic	Organic	<b>0.90</b>
	Synthetic	Organic	0.71
	Organic	Synthetic	0.94
	Synthetic	Synthetic	<b>0.99</b>

Table 6: Response-level Pooling<sub>E</sub> probe performance, as measured by **F1-R**, in detecting hallucinations on organic or synthetic ungrounded responses.

hallucination. This is also related to the idea of ecological validity (De Vries et al., 2020). Findings from an ecologically valid experiment should hold outside the context of that study. We do not find strong evidence that the detection of synthetic hallucinations generalizes outside that context.

**Lack of Task Transfer.** To test whether the probed features persist in similar hidden state locations across different tasks, we test our probe on each specific task when trained on the others. As shown in Table 8 in Appendix D, we find a lack of generalization in hallucination detection performance for our probes across different training and test tasks. This may only mean that each probe is overfitting to task-or-dataset-specific confounds. If so, training on a mixture of many tasks (more than the two we tried in Table 8) might result in a classifier that identifies the consistent hallucination signal and thus generalizes well to new tasks. However, it is also possible that no consistent hallucination signal exists.

## 7 Conclusion

We examined how an LLM’s internal representations reveal its hallucinatory behavior, via a case study of multiple in-context grounded generation tasks. In specific domains, we can accurately identify hallucinatory behavior through trained probes. Probe accuracy can depend on model layer, hidden state type, model size, hallucination type, and training-test mismatch. <sup>1</sup>*TL;DR: Yes, they do.*

## Limitations

Below, we highlight a few takeaways, broader implications, and limitations that naturally suggest directions for future work.

**Hallucinatory Span Detection.** In the context of use cases such as retrieval-augmented generation (RAG), span-level results have the potential to be extremely useful because they allow us to detect where hallucinations may fall. For example, hallucinatory spans could be (1) highlighted in the UI for RAG-based applications, (2) detected as soon as they are generated, triggering the decoder to backtrack and regenerate, or (3) a local hallucination detection could be used for further mitigation of hallucination in a reward-shaping setting for RLHF, in contrast to only having response-level signals available.

**Efficiency and Access.** Our work is motivated by the need for efficient LLM hallucination evaluation. Though computationally efficient, one of the key limitations of probing is in the need for labeled in-domain data for probe training. Our results in Table 8 suggest poor generalization to out-of-domain tasks, although perhaps this would be mitigated by larger and more diverse training sets. Furthermore, it perhaps goes without saying that probing requires access to the hidden states themselves. If LLMs continue to move behind closed-source APIs, such hidden state probes will not be possible for those without privileged access.

**Annotator Disagreements.** Labeling for faithfulness errors remains a challenging task for annotators (Ladhak, 2024). Despite asking annotators to label what they deem to be the *minimal spans* of hallucination in a response and providing a considerably detailed annotation schema (Appendix A), sizable variation still exists between annotators on judgments of their exact span boundaries. It remains an open question how to reduce or circumvent annotator disagreement in this context.

**Probe Design.** Even if span boundaries were standardized, they may not align perfectly with hallucination signals in LLM internal representations. For instance, while a generation may first contradict a given knowledge source at a certain token, the ultimate *signal* that it will hallucinate may already be salient in the tokens preceding it. Alternatively, the signal might be computed only later, as the decoder encodes previously generated

tokens and recognizes that it is currently pursuing an ungrounded train of thought. Thus, there may be probe architectures that could improve hallucination detection performance and mitigation. Nonlinear probes such as feed-forward networks are also worth considering: the hidden layer(s) of a feed-forward network could learn to detect different types of hallucinations. Furthermore, the lack of task transfer that we currently observe presents interesting future challenges in increasing the generalizability of our approach in search of a more consistent signal of hallucination, if one exists.

**Mitigation.** Finally, once we can *detect* ungrounded generation behavior, we can try to learn to *avoid* it. This has great practical importance as retrieval-oriented generation becomes popular across a variety of consumer-facing applications.

We have left this subject for future work. A simple method is to reject a response or prefix where hallucinations have been detected with high probability, and start sampling the response again (Guerreiro et al., 2023). Another approach would be to use the predicted hallucination probabilities to reweight or filter hypotheses during beam search. Finally, a detector can also be used as a proxy reward method for fine-tuning the language model, using a reward-based method such as PPO (Schulman et al., 2017). In addition to learning from response-level negative rewards incurred at the end of the utterance, PPO can benefit from early token-level negative rewards, which serve as a more informative signal for reinforcement learning (i.e., reward shaping) and thus can reduce sample complexity.

## Acknowledgments

We thank Val Ramirez, Patrick Xia, Harsh Jhamtani, Hao Fang, Tongfei Chen, Radhika Gaonkar, and Tatsunori Hashimoto for their helpful comments, thoughts, and discussions, as well as the data specialists who contributed and aided in the creation of this work.

## References

- Guillaume Alain and Yoshua Bengio. 2017. [Understanding intermediate layers using linear classifier probes](#). In *The Fifth International Conference on Learning Representations*.
- Amos Azaria and Tom Mitchell. 2023. [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.
- Jennifer A. Bishop, Sophia Ananiadou, and Qianqian Xie. 2024. [LongDocFACTScore: Evaluating the factuality of long document abstractive summarisation](#). In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 10777–10789, Torino, Italia. ELRA and ICCL.
- Harm De Vries, Dzmitry Bahdanau, and Christopher Manning. 2020. [Towards ecologically valid research on language user interfaces](#). *arXiv preprint arXiv:2007.14435*.
- Esin Durmus, He He, and Mona Diab. 2020. [FEQA: A question answering evaluation framework for faithfulness assessment in abstractive summarization](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5055–5070, Online. Association for Computational Linguistics.
- Ondřej Dušek, David M. Howcroft, and Verena Rieser. 2019. [Semantic noise matters for neural natural language generation](#). In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 421–426, Tokyo, Japan. Association for Computational Linguistics.
- Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, et al. 2021. [A mathematical framework for transformer circuits](#). *Transformer Circuits Thread*, 1.
- Tobias Falke, Leonardo F. R. Ribeiro, Prasetya Ajie Utama, Ido Dagan, and Iryna Gurevych. 2019. [Ranking generated summaries by correctness: An interesting but challenging application for natural language inference](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2214–2220, Florence, Italy. Association for Computational Linguistics.
- Angela Fan, Mike Lewis, and Yann Dauphin. 2018. [Hierarchical neural story generation](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898, Melbourne, Australia. Association for Computational Linguistics.
- Saadia Gabriel, Asli Celikyilmaz, Rahul Jha, Yejin Choi, and Jianfeng Gao. 2021. [GO FIGURE: A meta evaluation of factuality in summarization](#). In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 478–487, Online. Association for Computational Linguistics.
- Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2021. [Transformer feed-forward layers are key-value memories](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5484–5495, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Tanya Goyal and Greg Durrett. 2020. [Evaluating factuality in generation with dependency-level entailment](#). In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3592–3603, Online. Association for Computational Linguistics.
- Tanya Goyal and Greg Durrett. 2021. [Annotating and modeling fine-grained factuality in summarization](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1449–1462, Online. Association for Computational Linguistics.
- Nuno M. Guerreiro, Elena Voita, and André Martins. 2023. [Looking for a needle in a haystack: A comprehensive study of hallucinations in neural machine translation](#). In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1059–1075, Dubrovnik, Croatia. Association for Computational Linguistics.
- Andreas Hanselowski, Hao Zhang, Zile Li, Daniil Sorokin, Benjamin Schiller, Claudia Schulz, and Iryna Gurevych. 2018. [UKP-athene: Multi-sentence textual entailment for claim verification](#). In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 103–108, Brussels, Belgium. Association for Computational Linguistics.
- Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. [Teaching machines to read and comprehend](#). *Advances In Neural Information Processing Systems*, 28.
- Luyang Huang, Lingfei Wu, and Lu Wang. 2020. [Knowledge graph-augmented abstractive summarization with semantic-driven cloze reward](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5094–5107, Online. Association for Computational Linguistics.
- Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. [Survey of hallucination in natural language generation](#). *ACM Computing Surveys*, 55(12):1–38.
- 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.
- Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. 2022. [Language models \(mostly\) know what they know](#). *arXiv preprint arXiv:2207.05221*.
- Diederik P. Kingma and Jimmy Ba. 2014. [Adam: A method for stochastic optimization](#). In *International Conference on Learning Representations*.
- Wojciech Kryscinski, Nitish Shirish Keskar, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. [Neural text summarization: A critical evaluation](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 540–551, Hong Kong, China. Association for Computational Linguistics.
- Wojciech Kryscinski, Bryan McCann, Caiming Xiong, and Richard Socher. 2020. [Evaluating the factual consistency of abstractive text summarization](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9332–9346, Online. Association for Computational Linguistics.
- Philippe Laban, Tobias Schnabel, Paul N. Bennett, and Marti A. Hearst. 2022. [SummaC: Re-visiting NLI-based models for inconsistency detection in summarization](#). *Transactions of the Association for Computational Linguistics*, 10:163–177.
- Faisal Ladhak. 2024. [Faithfulness in Abstractive Summarization: Progress and Challenges](#). Columbia University.
- J. Richard Landis and Gary G. Koch. 1977. [The measurement of observer agreement for categorical data](#). *Biometrics*, pages 159–174.
- Katherine Lee, Orhan Firat, Ashish Agarwal, Clara Fanjjang, and David Sussillo. 2018. [Hallucinations in neural machine translation](#).
- Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. [Retrieval-augmented generation for knowledge-intensive NLP tasks](#). *Advances in Neural Information Processing Systems*, 33:9459–9474.
- Hui Lin and Vincent Ng. 2019. [Abstractive summarization: A survey of the state of the art](#). *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):9815–9822.
- Kevin Liu, Stephen Casper, Dylan Hadfield-Menell, and Jacob Andreas. 2023. [Cognitive dissonance: Why do language model outputs disagree with internal representations of truthfulness?](#) In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4791–4797, Singapore. Association for Computational Linguistics.
- Zheheng Luo, Qianqian Xie, and Sophia Ananiadou. 2023. [ChatGPT as a factual inconsistency evaluator for abstractive text summarization](#). *arXiv preprint arXiv:2303.15621*.
- Liang Ma, Shuyang Cao, Robert L Logan IV, Di Lu, Shihao Ran, Ke Zhang, Joel Tetreault, and Alejandro Jaimes. 2023. [BUMP: A benchmark of unfaithful minimal pairs for meta-evaluation of faithfulness metrics](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12788–12812, Toronto, Canada. Association for Computational Linguistics.
- Andreas Madsen, Siva Reddy, and Sarath Chandar. 2023. [Faithfulness measurable masked language models](#). *arXiv preprint arXiv:2310.07819*.
- Andrey Malinin and Mark Gales. 2021. [Uncertainty estimation in autoregressive structured prediction](#). In *International Conference on Learning Representations*.
- Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. [Locating and editing factual associations in GPT](#). *Advances in Neural Information Processing Systems*, 36.
- Kevin Meng, Arnab Sen Sharma, Alex J Andonian, Yonatan Belinkov, and David Bau. 2023. [Mass-editing memory in a transformer](#). In *The Eleventh International Conference on Learning Representations*.
- Sabrina J. Mielke, Arthur Szlam, Emily Dinan, and Y-Lan Boureau. 2022. [Reducing conversational agents’ overconfidence through linguistic calibration](#). *Transactions of the Association for Computational Linguistics*, 10:857–872.
- 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](#). In *EMNLP*.
- Anshuman Mishra, Dhruvesh Patel, Aparna Vijayakumar, Xiang Lorraine Li, Pavan Kapanipathi, and Kartik Talamadupula. 2021. [Looking beyond sentence-level natural language inference for question answering and text summarization](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1322–1336, Online. Association for Computational Linguistics.

- Giovanni Monea, Maxime Peyrard, Martin Josifoski, Vishrav Chaudhary, Jason Eisner, Emre Kıcıman, Hamid Palangi, Barun Patra, and Robert West. 2023. [A glitch in the Matrix? Locating and detecting language model grounding with Fakepedia](#). *arXiv preprint arXiv:2312.02073*.
- Noel Ngu, Nathaniel Lee, and Paulo Shakarian. 2024. [Diversity measures: Domain-independent proxies for failure in language model queries](#). In *2024 IEEE 18th International Conference on Semantic Computing (ICSC)*, pages 176–182. IEEE.
- Jekaterina Novikova, Ondřej Dušek, and Verena Rieser. 2017. [The E2E dataset: New challenges for end-to-end generation](#). In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 201–206, Saarbrücken, Germany. Association for Computational Linguistics.
- Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. [Language models are unsupervised multitask learners](#).
- 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*.
- Sashank Santhanam, Behnam Hedayatnia, Spandana Gella, Aishwarya Padmakumar, Seokhwan Kim, Yang Liu, and Dilek Hakkani-Tur. 2021. [Rome was built in 1776: A case study on factual correctness in knowledge-grounded response generation](#). *arXiv preprint arXiv:2110.05456*.
- John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. [Proximal policy optimization algorithms](#). *arXiv preprint arXiv:1707.06347*.
- Thomas Scialom, Paul-Alexis Dray, Sylvain Lamprier, Benjamin Piwowarski, Jacopo Staiano, Alex Wang, and Patrick Gallinari. 2021. [QuestEval: Summarization asks for fact-based evaluation](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6594–6604, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Sina Semnani, Violet Yao, Heidi Zhang, and Monica Lam. 2023. [WikiChat: Stopping the hallucination of large language model chatbots by few-shot grounding on Wikipedia](#). In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 2387–2413, Singapore. Association for Computational Linguistics.
- Aviv Slobodkin, Omer Goldman, Avi Caciularu, Ido Dagan, and Shauli Ravfogel. 2023. [The curious case of hallucinatory \(un\)answerability: Finding truths in the hidden states of over-confident large language models](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3607–3625, Singapore. Association for Computational Linguistics.
- Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. [Llama 2: Open foundation and fine-tuned chat models](#). *arXiv preprint arXiv:2307.09288*.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. [Attention is all you need](#). In *Advances in Neural Information Processing Systems 30*, pages 5998–6008.
- Alex Wang, Kyunghyun Cho, and Mike Lewis. 2020. [Asking and answering questions to evaluate the factual consistency of summaries](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5008–5020, Online. Association for Computational Linguistics.
- Rongwu Xu, Zehan Qi, Cunxiang Wang, Hongru Wang, Yue Zhang, and Wei Xu. 2024. [Knowledge conflicts for LLMs: A survey](#). *arXiv preprint arXiv:2403.08319*.
- Weijia Xu, Sweta Agrawal, Eleftheria Briakou, Marianna J. Martindale, and Marine Carpuat. 2023. [Understanding and detecting hallucinations in neural machine translation via model introspection](#). *Transactions of the Association for Computational Linguistics*, 11:546–564.
- Mert Yuksekgonul, Varun Chandrasekaran, Erik Jones, Suriya Gunasekar, Ranjita Naik, Hamid Palangi, Ece Kamar, and Besmira Nushi. 2023. [Attention satisfies: A constraint-satisfaction lens on factual errors of language models](#). In *The Twelfth International Conference on Learning Representations*.
- Ran Zmigrod, Tim Vieira, and Ryan Cotterell. 2022. [Exact paired-permutation testing for structured test statistics](#). In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4894–4902, Seattle, United States. Association for Computational Linguistics.

## Appendix

Here, we provide additional information about annotation guidelines, annotator disagreement examples, prompts used to generate organic responses to our tasks, ChatGPT baseline comparisons, and expanded results.

### A Annotation Guidelines

The purpose of this annotation task is two-fold: (1) identify if a generated response is a hallucination relative to a given knowledge source, and (2) if so, identify and mark the location(s) of the minimal spans of text that are ungrounded against the knowledge source. A response is grounded in a knowledge source if that source entails the content of the response. In other words, the response should (1) not contradict or (2) draw conclusions that are not explicitly given in the knowledge source.

#### A.1 CNN / DailyMail

**Provided Information.** (1) Original article, (2) summary of article.

**Task.** Identify the presence and location of hallucinations in the summary of the article, relative to the original article text. You are to only annotate—by copying the text in the summary and adding characters < and > in specific locations—around the minimal span(s) of the summary text relative to the original article; I.e., if changing an additional token wouldn't change whether or not the response was grounded or not, don't include it in the span you annotate. Note: You will be given summaries that contain a variety number of sentences. You are only to annotate the *first three sentences* in the summary, discarding the rest.

**Example.** (CNN) – The world's fastest man on a pair of skis just got faster. On Monday Italian Simone Origone broke his own speed skiing world record as he reached 252.4 kilometers per hour on the Chabrieres slopes in the French Alps – an achievement confirmed by organizers France Ski de Vitesse. With a 1,220 m slope that has a maximum gradient of 98% and an average of 52.5%, Chabrieres is not for the faint-hearted. Traveling at over 250 km/h is a privilege usually reserved for Formula One drivers, yet speed skier Origone was equipped with just an aerodynamic helmet to increase streamlining and a ski suit made from airtight latex to reduce wind resistance. Origone's one nod to safety was wearing a back protector

in case of a crash as he threw himself down the one kilometer track. Origone has been the fastest speed skier on the globe since April 2006, having set a then-new world record of 251.4 km/h at Les Arcs. Bastien Montes of France was Origone's closest challenger on Monday, but even the Frenchman's new personal best of 248.105 km/h was somewhat short of the Italian's 2006 world record, let alone his latest one. "Simone Origone is the greatest champion of all time, he is the only person to hold the record for two ski speeds in France – Les Arcs and Vars Chabrieres. This is a historic day," technical director of Speed Masters Philippe Billy told Vars.com. The 34-year-old Origone – a ski instructor, mountain guide and rescuer by day – only took up the discipline in 2003, having given up downhill skiing in 1999. "Now that I have twice won the world record, I can say that I have made history," Origone told Vars.com. "It is important for me and for speed skiing."

**Hypothetical Instance 1.** The world's fastest man on skis just got faster.

- Everything is grounded, no need to mark anything here.

**Annotation.** The world's fastest man on skis just got faster.

**Hypothetical Instance 2.** The world's fastest man on skis just got faster. He has now won the world record twice. He has excelled at Chabrieres. He has been the fastest speed skier on the globe since April 2006. His speed record reached more than 260 kilometers per hour.

- This summary has more than 3 sentences, so the first thing to do is to only look at the first three and discard any sentences beyond that in your annotation.
- Even though the last sentence is ungrounded relative to the source text, it is not within the first three sentences, so we can ignore it here.
- Annotate as such, copying the first three sentences into the annotation box. No need to highlight anything, as everything in the first three sentences are grounded:

**Annotation.** The world's fastest man on skis just got faster. He has now won the world record twice. He has excelled at Chabrieres.

**Hypothetical Instance 3.** Simone Origone, now in his twenties, just got faster. He has now won the world record more than twice.

- “In his twenties” contradicts with “The 34-year-old Origone” in the original article.
- “more than twice” contradicts with how he has now only won it exactly twice. As such, annotate:

**Annotation.** Simone Origone, now in his <twenties>, just got faster. He has now won the world record <more than> twice.

**Special Points of Note.** Below, we list a few points of special note to keep in mind during your annotations.

**Concatenation.** When finding yourself annotating something like “The city’s pollution levels have reached or <exceeded ><“very high”> levels for the last...,” do concatenate the two labeled spans; if there are no characters between them (for example, <exceeded ><“very high”>), concatenating the annotation bounds into (<exceeded “very high”>) is ideal.

**“Was”/“Is”.** If the article says something like “Woods insisted he is ready to win the famous Claret...” and the summary said something like “said he was ready to win the famous Claret...”, “was” here would not be considered to be a hallucination, as the summary is recounting what was said in the article, so the past tense here doesn’t contradict what was originally said in the article.

**When The Article Has “Contradicting” Facts.** Articles may sometimes “correct” themselves; i.e. the article may have something where it originally states a claim but then corrects it later in the article. For example, “The NWS – last May – initially forecast 13 to 20 “named storms,” including seven to 11 hurricanes. Then in August, it dialed down that initial forecast to predict 13 to 19 named storms, including six to nine hurricanes.”

In this case, if the summary only states “The National Weather Service forecast called for 13 to 20 named storms” (and doesn’t mention it was corrected later to 13 to 19), then this is a hallucination as it is corrected later in the article.

If the summary states “The National Weather Service forecast initially called for 13 to 20 named storms” (and doesn’t mention it was corrected later

to 13 to 19) then this is not a hallucination, as it quantifies the original prediction with “initially.”

If the summary states “The National Weather Service forecast called for 13 to 19 named storms,” then this is not a hallucination, as it is the final corrected fact.

**Identifying the Summary.** Sometimes the summarization system may produce things that are not a summary of the article. For example, “Woman dies while paragliding in Tenerife. Paragliding in Tenerife. Paragliding in Tenerife. Paragliding in Tenerife. Paragliding Tenerife Canary Islands.”

Here, only the first sentence is the summary, and you can discard the rest. This is because the summarization system tries to mimic how actual website summaries work, so they might also add things like topic tags, keywords, and things like that.

Another example is “InvisiBra, £48, Lavalia, lavalia.co.uk; Strapless Stick On Adhesive Bra, £12, Marks and Spencer; Natural Stick On Bra, £20, Debenhams.” These are all just keyword tags; no summary of the article is present here. Try to use your best judgment here, as it might not always be clear.

In the case where such things are generated that do not look to be part of a summary of the article in terms of format (not content), *discarding them and everything after them is the way to go*. For the former example, you would annotate only “Woman dies while paragliding in Tenerife.” for hallucinations.

## A.2 Conv-FEVER

**Provided Information.** (1) Conversational context, (2) Knowledge for next response, (3) Next response.

**Task.** Identify the presence and location of hallucinations in “Next response”, relative to “Knowledge for next response”. You are to only annotate—by copying the text in “Next Reply” and adding characters < and > in specific locations—around the minimal span(s) of an ungrounded response relative to the knowledge; i.e., if changing an additional token wouldn’t change whether or not the response was grounded or not, don’t include it in the span you annotate.

**Example. Conversational Context:** A: lets talk bowling a game where you throw a bowling ball towards a target. B: ok I love bowling I would

like to know more A: for one it is one of the major forms of throwing sports and I think a lot of people enjoy it B: yes me too I use to bowl all the time with my family what about the strike? A: well a strike is when you knock down all pins on the first round B: ok how can I get a turkey I think that may be hard to get though **Knowledge for Next Reply:** Three consecutive strikes are known as a "turkey".

**Hypothetical Instance 1. Next Reply:** A: yea four consecutive strikes is known as a turkey

- Here, a contradiction is present in the underlined portions: four consecutive strikes in the response goes against what is provided in the knowledge, which says three.
- To make this reply grounded, one would have to change “four” to “three”.
- As such, in your annotation, you will highlight the minimum needed span of text to make this reply grounded, with chevrons <>. Here, the minimum needed span of text to change is “four”, so you would annotate as such:

**Annotation.** yea <four> consecutive strikes is known as a turkey

**Hypothetical Instance 2. Next Reply:** A: yea it is pretty hard, four consecutive strikes is known as a turkey

- Here, the contradiction is still present in the previously underlined portion: four is a contradiction with three.
- However, in addition to this contradiction, you see an expression of an opinion in this response: that getting a turkey is pretty hard (underlined).
- You are to ignore the expression of personal opinions during your annotation of minimal spans for this task, as we won’t consider conversational chitchat—if complemented with grounded information elsewhere in the reply—to be a hallucination for this task. As such, you would annotate this as:

**Annotation.** yea it is pretty hard, <four> consecutive strikes is known as a turkey

**Hypothetical Instance 3. Next Reply:** A: yea it is pretty hard.

- Here, the response is purely an expression of an opinion and does not use any information in the provided knowledge.
- As such, you would mark this entire response to be a hallucination, annotating as:

**Annotation.** <yea it is pretty hard.>

**Hypothetical Instance 4. Next Reply:** A: Three strikes consecutively is a turkey.

- Here, the response is grounded. As such, you won’t mark anything in the response, annotating as such:

**Annotation.** Three strikes consecutively is a turkey.

**Hypothetical Instance 5. Next Reply:** A: four consecutive strikes is known as a turkey, also known as a strike.

- Two hallucinations are present in this response:
- The direct contradiction of four with three in the knowledge source, and
- The final part of the reply, saying that it’s also known as a strike, which is information not provided anywhere in the knowledge source. As such, you would mark this as:

**Annotation.** <four> consecutive strikes is known as a turkey, <also known as a strike>

### A.3 E2E

**Provided Information.** (1) Restaurant attributes, (2) Generated restaurant description.

**Task.** Identify the presence and location of hallucinations in the generated restaurant description—by copying the text in the summary and adding characters < and > in specific locations—around the minimal span(s) of the generated restaurant description relative to the provided attributes. i.e., if changing an additional token wouldn’t change whether the description was grounded or not, don’t include it in the span you annotate. Note that numerical ratings should be rendered as numerical ratings and text ratings should



be rendered as non-numerical expressions. For example, “customer rating” can be “low”, “medium” and “high”, but also “1”, “3”, and “5”. If the customer rating is “1”, for instance, and the generated description says “. . . has a low customer rating”, that would be considered a hallucination. The “priceRange” rating also has this attribute, in addition to “customer rating.” Finally, E2E specifies the “familyFriendly” attribute as akin to Child-friendly in gold annotations: if a restaurant is family-friendly, then it is also child-friendly.

**Hypothetical Instance 1. Restaurant Attributes:** name[The Wrestlers], priceRange[less than £20], customer rating[low], familyFriendly[no]. **Restaurant Description:** The Wrestlers is an adults only restaurant with a customer rating of 1 out of 5 and a price range of less than £20.

- While the familyFriendly[no] attribute says that the restaurant is not family friendly, this does not mean that the restaurant is adults only.
- - Having a low customer rating does not imply a 1 out of 5 customer rating. As such, annotate as:

**Annotation.** The Wrestlers is an <adults only> restaurant with a customer rating of <1 out of 5> and a price range of less than £20.

**Hypothetical Instance 2. Restaurant Attributes:** name[The Dumpling Tree], eatType[restaurant], food[Italian], priceRange[high]. **Restaurant Description:** The Dumpling Tree is an Italian restaurant with high prices.

- Everything is grounded, no need to mark anything.

**Annotation.** The Dumpling Tree is an Italian restaurant with high prices.

**Hypothetical Instance 3. Restaurant Attributes:** name[The Wrestlers], priceRange[less than £20], customer rating[low], familyFriendly[no]. **Restaurant Description:** Prices will never exceed £20 at The Wrestlers and they also don’t really rate well. They are not even family-friendly.

- Having a priceRange attribute of less than 20 does not mean that prices will never exceed twenty.

- As “will never exceed” is the minimal span you would need to change to make the description grounded relative to the provided attributes, annotate:

**Annotation.** Prices <will never exceed> £20 at The Wrestlers and they also don’t really rate well. They are not even family-friendly.

**Hypothetical Instance 4. Restaurant Attributes:** name[Midsummer House], food[French], customer rating[5 out of 5], near[Café Rouge]. **Restaurant Description:** A place for fine dining French food, Midsummer House is nearby Cafe Rouge.

- Nowhere in the attributes does it say that it’s “fine dining.” As such, annotate:

**Annotation.** A place for <fine dining> French food, Midsummer House is nearby Cafe Rouge.

**Hypothetical Instance 5. Restaurant Attributes:** name[Browns Cambridge], priceRange[cheap], priceRange[moderate], customer rating[1 out of 5]. **Restaurant Description:** The Brown’s chain of food restaurants is an 1 out of 5 customer rated restaurant, and have moderate and cheap pricing. They are childfree.

- The attribute does not mention that Browns Cambridge is a chain of food restaurants.
- Furthermore, it does not say that the restaurant is childfree. As such, annotate:

**Annotation.** The Brown’s <chain of food restaurants> is an 1 out of 5 customer rated restaurant, and have moderate and cheap pricing. <They are childfree.>

**Hypothetical Instance 6. Restaurant Attributes:** name[Browns Cambridge], priceRange[more than £30]. **Restaurant Description:** The Browns Cambridge is a restaurant with high prices.

- More than £30 does not necessarily mean “high prices.” As such, annotate:

**Annotation.** The Browns Cambridge is a restaurant with <high prices>.

---

Taxi driver dropped off a soldier on an unsafe road, and he was killed by a coach. **The taxi driver** was not at fault.

Taxi driver dropped off a soldier on an unsafe road, and he was killed by a coach. **The taxi driver was not at fault.**

---

Accounting is the **language of business, and it measures the results of an organization’s economic activities.**

**Accounting is the language of business, and it measures the results of an organization’s economic activities.**

---

Alimentum is a **high rated** family-unfriendly restaurant with less than £20 per person.

Alimentum is a **high** rated family-unfriendly restaurant with less than £20 per person.

---

Table 7: Annotator span-level annotation differences for hallucination, annotated in **bold**, in organic model responses for CNN / DailyMail (top), Conv-FEVER (middle), and E2E (bottom).

## B Annotator Disagreements

Examples of annotator disagreements in the exact spans of hallucination are shown in Table 7. Qualitatively, we find that annotators agreed *generally* on the specific items of hallucination; i.e. the subject that was hallucinated (i.e. the taxi driver), the description of what was mentioned (i.e. the description of accounting), and the attribute that was hallucinated (i.e. the specific rating of the restaurant). However, *where* exactly those hallucinated spans began and ended was a source of reasonable variation among annotators.

## C Prompts and Further Evaluation Details

**Organic Model Responses.** CNN / DailyMail generations take the form of a 0-shot prompt with [ARTICLE] TL;DR: [SUMMARY], whereafter the first 3 sentences generated in the summary are taken to be the summary of the article. For Conv-FEVER, we use a 2-shot in-context learning prompt in the form

```
## CONVERSATION HISTORY
[CONTEXT]
```

```
## KNOWLEDGE FOR NEXT RESPONSE
[KNOWLEDGE]
## RESPONSE THAT USES THE
INFORMATION IN KNOWLEDGE
[RESPONSE]
```

whereafter everything generated in the response preceding a newline character is taken as the final generation. For E2E, we use a 10-shot in-context learning prompt in the form of

```
Restaurant Attributes:
[RESTAURANT ATTRIBUTES]
Restaurant Description:
[RESTAURANT DESCRIPTION]
```

with the same newline constraints as Conv-FEVER.

**FActScore Evaluation.** As FActScore (Min et al., 2023) was originally designed for evaluating the faithfulness of long-form generated texts, we slightly modify its evaluation for our Data-to-Text task, E2E. Specifically, instead of evaluating on raw attribute sets (i.e. name[Clowns], eatType[pub], near[The Sorrento]), we transform these attributes into natural language sentences via simple regular expressions. In this example, these attributes would instead be transformed into:

```
The name of this restaurant is Clowns.
This restaurant is a pub. This restaurant
is near The Sorrento.
```

For our other tasks on abstractive summarization and knowledge-grounded dialogue generation, we keep evaluations unchanged with the original implementation of FActScore for our evaluations.

### C.1 ChatGPT Prompts

**ChatGPT Classic.** For the most basic zero-shot prompt classifier with no reasoning, we use the following prompt for CNN / DailyMail, making minimal changes relative to Luo et al. (2023):

```
Decide if the following summary
(SUMMARY) is consistent with the
corresponding article (ARTICLE).
Note that consistency means all
information in the summary
```

(SUMMARY) is supported by the article (ARTICLE).

Article (ARTICLE): [Article]

Summary (SUMMARY): [Summary]

Answer (yes or no):

For **Conv-FEVER**, our prompt is structured as follows:

Decide if the following sentence (SENTENCE) is consistent with the corresponding knowledge statement (KNOWLEDGE). Note that consistency means all information in the sentence (SENTENCE) is supported by the knowledge statement (KNOWLEDGE).

Knowledge Statement (KNOWLEDGE): [Knowledge]

Sentence (SENTENCE): [Response]

Answer (yes or no):

Finally, for **E2E**, our prompt is:

Decide if the following restaurant description (DESCRIPTION) is consistent with the provided restaurant attributes (ATTRIBUTES). Note that consistency means all information in the description is supported by the provided attributes.

Restaurant Attributes (ATTRIBUTES): [Attributes]

Restaurant Description (DESCRIPTION): [Response]

Answer (yes or no):

**ChatGPT CoT.** For zero-shot chain-of-thought reasoning prompts, we simply replace Answer (yes or no): in all prompts with Explain your reasoning step by step then answer (yes or no) the question:, following (Luo et al., 2023).

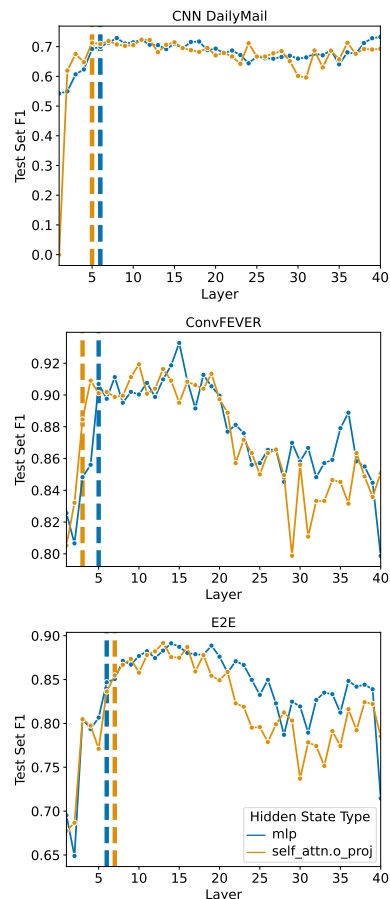


Figure 4: Grounding behavior saliency—as measured by probe response-level F1 on test set—of single layer probe classifiers trained on the **attention-head out projections** and **feed-forward states** of llama-2-13b, stratified across tasks and layers. Vertical dashed lines denote the layers that respective probes first surpass 95% of their peak performance.

## D Expanded Results and Discussions

**Grounding Behavior Saliency.** Full saliency results for all tasks and layers are shown in Figure 4; saliency as stratified across hallucination layers and types across both MLP and Attention probes is shown in Figure 5.

**Task Transfer.** Full results for task transfer using the response-level Pooling<sub>E</sub> probe are shown in Table 8.

**Differences from Specific Prior Work.** In the review process, reviewers raised uncertainties about the specific differences between our work and that of Azaria and Mitchell (2023) and Kadavath et al. (2022); we note our response here.

Azaria and Mitchell (2023) are interested in detecting the truth or falsehood of factual statements

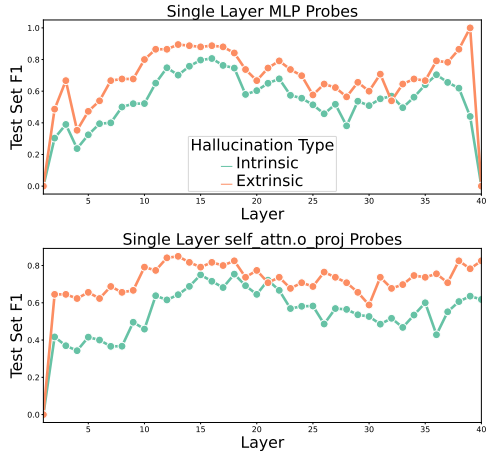


Figure 5: **Intrinsic** and **extrinsic** hallucination saliency across layers for synthetic abstractive summarization (BUMP, for CDM) as measured by response-level MLP and Attention probe F1 on hallucination types.

(e.g., “H2O is water, which is essential for humans”). The models they probe are presented with these statements without any grounding knowledge, so the probe can only measure if the model’s hidden states reveal some discrepancy between the encoded statement and the training data as encoded by the learned weights. This cannot work unless the learned weights really encode the truth, so one should only test facts that are well-known and unchanging or slowly changing in time, modality, etc.

In contrast, a RAG chatbot built on a company’s internal data will be discussing specialized and often ephemeral retrieved information that is not well represented in the world knowledge of an LLMs weights. Here, it is important to ensure that the model appropriately follows task instructions, and in particular that it does not hallucinate in the sense defined by §1 and Ji et al. (2023): its responses should be appropriately grounded in the retrieved data.

For emphasis we highlight three important distinctions. (1) Azaria and Mitchell (2023) examine the veracity of statements without supporting text, while we examine hallucination with respect to grounding text and task instructions. (2) They do not probe models on text that the models themselves generate, but text provided by a secondary model (ChatGPT) and additional human curation. This is an important distinction—we evaluate hallucination on both organically generated texts (and their hidden states) and those generated by a secondary model + human manual curation (synthetic grounding errors) and find that the probe perfor-

Test	Train	F1-R
CDM	CF	0.67
	E2E	0.65
	CF+E2E	0.68
	CDM	<b>0.70</b>
CF	CDM	0.84
	E2E	0.81
	CDM+E2E	0.84
	CF	<b>0.93</b>
E2E	CDM	0.06
	CF	0.71
	CDM+CF	0.71
	E2E	<b>0.86</b>

Table 8: Response-level Pooling<sub>E</sub> cross-task probe performance, as measured by **F1-R**. All experiments here use the same amount of training data (a random subset of the full training set). The “+” sign indicates an equal mixture of two training tasks.

mance generally degrades when probing across these two cases. Finally, (3) they do not study span or token-level detection, which is one of our primary contributions.

Kadavath et al. (2022) similarly probe an LLM in a closed book QA task by training an attention head to detect when the model does not know the answer to a trivia question in the TriviaQA dataset (Joshi et al., 2017). This is different than detecting hallucination in a grounded generation task, as it only tests recall of knowledge already stored in model weights. As for Azaria and Mitchell (2023), their generation task does not require the model to draw information from evidence in the prompt that may disagree with stored knowledge.