FACTCHECKMATE: Preemptively Detecting and Mitigating Hallucinations in LMs

Deema Alnuhait[1*], **Neeraja Kirtane**[1*], **Muhammad Khalifa**[2], **Hao Peng**[1]
[1]University of Illinois Urbana-Champaign, [2]University of Michigan
{deemaa2,kirtane3,haopeng}@illinois.edu
khalifam@umich.edu

Abstract

Language models (LMs) hallucinate. We inquire: Can we detect and mitigate hallucinations *before* they happen? This work answers this research question in the positive, by showing that the internal representations of LMs provide rich signals that can be used for this purpose. We introduce FACTCHECKMATE, which *preemptively* detects hallucinations by learning a classifier that predicts whether the LM *will* hallucinate, based on the model's hidden states produced over the inputs, *before* decoding begins. If a hallucination is detected, FACTCHECKMATE then intervenes by adjusting the LM's hidden states such that the model will produce more factual outputs. FACTCHECKMATE provides fresh insights that the inner workings of LMs can be revealed by their hidden states. Practically, both its detection and mitigation models are lightweight, adding little inference overhead; FACTCHECKMATE proves a more efficient approach for mitigating hallucinations compared to many post-hoc alternatives. We evaluate FACTCHECKMATE over LMs of different scales and model families (including Llama, Mistral, Qwen and Gemma), across a variety of QA datasets from different domains. Our results demonstrate the effectiveness of FACTCHECKMATE, achieving over 70% *preemptive* detection accuracy. On average, outputs generated by LMs with intervention are 34.4% more factual compared to those without.

1 Introduction

Language models (LMs) hallucinate, a phenomenon where they produce nonfactual or even misleading outputs that often appear plausible (Ji et al., 2023a; Bang et al., 2023; Xu et al., 2024; Zhang et al., 2023; Li et al., 2024a; Huang et al., 2023; Ye et al., 2023). Extensive efforts have been devoted to mitigating their hallucination issues (Min et al., 2023; Manakul et al., 2023b; Rawte et al., 2023; Zhou et al., 2021). These approaches are mostly *reactive*, addressing hallucinations *after* they occur, and often require resampling new outputs (Li et al., 2023; Manakul et al., 2023a), substantially increasing the inference overhead. In addition, they often treat the LM as a black box, while relying on external LMs for detecting hallucinations, missing the opportunity to gain deeper insights into the internal workings of these models.

Recent findings by Azaria and Mitchell (2023) and Burns et al. (2022) show that the LMs' representations can provide useful information about the factuality of their outputs. Marks and Tegmark (2023) observe that LMs' hidden states generated over factual and non-factual statements are linearly separable. However, these studies have a relatively narrow focus, primarily addressing hallucination detection in a *reactive* manner, and a more thorough investigation is needed.

The key hypothesis of this paper is that, the LMs' hidden states reveals valuable information about their internal working mechanisms, and provide signals that can be used to predict whether it will hallucinate *before* decoding. We propose FACTCHECKMATE to answer the following research question (RQ): *Can we* ***preemptively*** *predict and mitigate hallucinations with LMs' internal representations?* FACTCHECKMATE learns a classifier that, taking the models' hidden states over the inputs, predicts whether the model *is about to* hallucinate. If a hallucination is detected, FACTCHECKMATE intervenes, by adjusting the LM's hidden states with a learned intervention model, and steering them towards producing more factual outputs (Figure 1).

Our controlled experiments answer the RQ in the positive. We evaluate FACTCHECKMATE across four QA datasets from different domains: NQ-open (Wikipedia; Lee et al., 2019), MMLU (STEM exams; Hendrycks et al., 2020), MedMCQA (medical; Pal et al., 2022), and GSM8K (Math; Cobbe

*Equal contribution.

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 12413–12428
November 4-9, 2025 ©2025 Association for Computational Linguistics

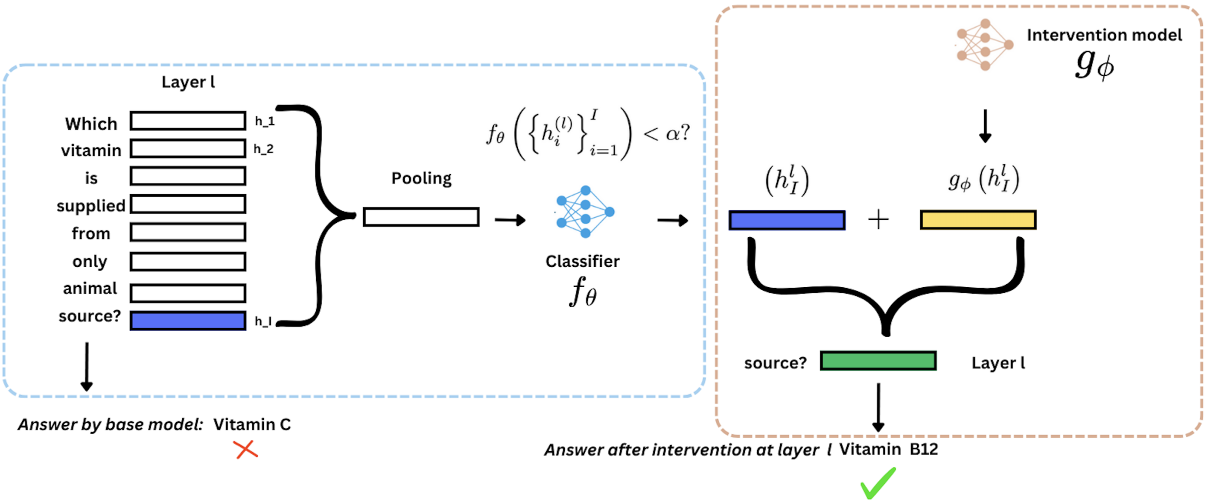


Figure 1: FACTCHECKMATE P *at inference time*. A demonstration of how preemptive detection and subsequent mitigation work. As shown, at a layer l, the hidden states of only the prefix I are aggregated and passed to the classifier f_θ (highlighted in the light blue box §3). Once hallucination is detected with classification probability $< \alpha$, $\mathbf{g}_\phi$ (highlighted in the light brown box §4) intervenes and adjusts the last token $\mathbf{h}_I^{(l)}$. This leads to a more factual output than before.

et al., 2021). Using the LMs' hidden states over the input questions, FACTCHECKMATE can successfully predict whether or not the LMs will hallucinate over 70% of the time, significantly outperforming a 50% random baseline (§3). We observe consistent trends across base and fine-tuned LMs of different scales and families, including Llama2 (7B and 13B; Touvron et al., 2023a), Llama3 (8B and 70B) and Llama3.1 (8B and 70B) (Dubey et al., 2024) Mistral-7B (Jiang et al., 2023), Gemma-7B (Team et al., 2024), and Qwen2.5 (7B and 32B) Qwen et al., 2025.

We further conduct cross-model and cross-dataset evaluations to assess the generalizability of FACTCHECKMATE. In the cross-model training, we train on the hidden states of multiple models. Whereas, in the cross-dataset, we train the classifier on multiple domains. The cross-model classifier achieves on average 71% accuracy in preemptive hallucination detection, demonstrating its robustness across different models. Similarly, the cross-dataset classifier attains an average of 65% accuracy. Furthermore, FACTCHECKMATE's intervention model can effectively improve the LMs' factuality. Using GPT-4o as a judge, which shows high agreement with human evaluations in our experiments, we find that on average, outputs generated by LMs with intervention are 34.4% more factual than those produced without intervention (§4). We also calculate the inference time overhead introduced by FACTCHECKMATE, incurring minimal average overhead of a 1.2% increase in decoding time, showing minimal impact on inference efficiency (§5.1). FACTCHECKMATE reveals surprising insights into existing LMs, and can potentially lead to more profound understanding of their internal working. All code, data, and checkpoints for reproducing our findings will be released.

2 Related Work and Motivation

Hallucination Detection. We investigate hallucinations in language models that generate responses based solely on their parametric knowledge, similar to Azaria and Mitchell (2023). This contrasts with in-context generation scenarios where external knowledge sources are explicitly incorporated within the prompt. We focus on addressing factuality hallucinations, an important type of hallucinations as argued in Huang et al. (2023).

Existing research primarily focuses on post-processing methods applied after the inference process is completed and often utilizing external knowledge for verification (Manakul et al., 2023a; Li et al., 2023; Chern et al., 2023). For instance, CRITIC (Gou et al., 2024) validates model outputs through tool interactions, and FACTSCORE (Min et al., 2023) breaks down generated content into atomic facts, assessing their accuracy by comparing them against reliable sources.

A recent line of research leverages the internal mechanics of LMs to detect hallucinations

(Burns et al., 2024; Azaria and Mitchell, 2023; Marks and Tegmark, 2023). Meng et al. (2022) locates where factual associations are stored in GPT models. These studies have spurred further research into using LMs' internal representations in hallucination detection (Chen et al., 2024a; CH-Wang et al., 2024). For instance, MIND (Su et al., 2024) generates training data in unsupervised approach for training hidden states based hallucination detectors. Duan et al. (2024) conducts an experimental examination of the hidden states of LLMs when processing factual versus nonfactual responses. FACTCHECKMATE demonstrates the effectiveness of preemptive hallucination detection, identifying warning signals several tokens **before** the hallucinations actually occur, via the language model's hidden states.

Hallucination Mitigation. For hallucination mitigation at inference time, existing works have explored self-correction and automated feedback approaches, where the language model is prompted to fix its generation flaws, with or without leveraging feedback from the model itself or some external knowledge source (Pan et al., 2023; Dhuliawala et al., 2023; Ji et al., 2023b). A recent approach involves utilizing activation engineering (Subramani et al., 2022; Duan et al., 2024; Zhang et al., 2024). FACTCHECKMATE builds on these findings and explores activation engineering techniques to preemptively intervene and mitigate hallucinations during inference time. It is also related to inference-time approaches that utilize a scoring function to steer the LM toward desired behavior (Dathathri et al., 2020; Khalifa et al., 2023).

Hidden States as Predictive Signals of Hallucination. Previous works primarily use hidden states as indicators of factuality after generation (Chuang et al., 2024; Zhang et al., 2024; Li et al., 2024b; Orgad et al., 2024). FACTCHECKMATE instead asks: Can hidden states reveal early signals of hallucination before tokens are generated? If so, this would suggest that factuality cues are embedded in the model's internal mechanisms earlier than previously assumed. FACTCHECKMATE demonstrates, for the first time, that hidden states offer early signals correlated with hallucination. Its finding reveals that factuality cues are embedded within the model's internal mechanisms well before the output is generated. This fresh insights allows FACTCHECKMATE to use the model's hidden states to anticipate when it is likely to hallucinate, rather than waiting for errors to surface in generated tokens. This reduces the need for expensive post-hoc corrections and provides insight into how factual knowledge is internally encoded and accessed. Our results (§3) suggest that hallucinations are not merely failures of token-level prediction but often emerge from systematic patterns in how models encode factual information during inference (Zou et al., 2023).

3 Preemptive Hallucination Detection

This section focuses on FACTCHECKMATE's preemptive hallucination classifier (§3.1) and experimental results (§3.2).

3.1 Preemptive Hallucination Detection with a Lightweight Classifier over Hidden States

Classifier. FACTCHECKMATE learns a binary classifier f_θ to preemptively detect hallucinations. Parameterized by a learned two-layer $\mathrm{ReLU\text{-}MLP}$ followed by a sigmoid function, f_θ takes as input the LM's hidden states and outputs the probability that the LM *will* hallucinate. More specifically, let $\{\boldsymbol{h}_i^{(l)}\}_{i=1}^{I}$ be a sequence of I hidden states that the LM produces over the input question with I tokens. A d-dimensional vector $\boldsymbol{h}_i^{(l)}$ denotes the output of the feedforward network (FFN) of the l-th transformer layer, at the i-th token.[1].

The classifier f_θ takes as input the pooled values over $\{\boldsymbol{h}_i^{(l)}\}_{i=1}^{I}$ and produces a scalar between 0 and 1 indicating the probability that the LM will hallucinate in its response to the input: $f_\theta(\{\boldsymbol{h}_i^{(l)}\}_{i=1}^{I}) =$

$$\sigma\left(\mathrm{ReLU\text{-}MLP}\left(\mathcal{A}\left(\{\boldsymbol{h}_i^{(l)}\}_{i=1}^{I}\right)\right)\right)$$

where $\mathcal{A}$ represents the pooling function, which can be the mean, max, or selecting the last token. l is empirically determined based on validation performance, and can vary by the LMs and datasets. In general, l tends to be the middle to last layers. More details about the best empirical layer for each LM can be found in Appendix C.1.

We train a separate classifier tailored to each LM. We consider LMs from different families of different scales, including Llama2 (7B and 13B; Touvron et al., 2023a), Llama3 (8B and 70B) and 3.1-8B (8B and 70B; Dubey et al., 2024), Mistral-7B (Jiang et al., 2023), Gemma-7B (Team et al.,

[1]We utilize the outputs of the FFNs, following previous work by (Marks and Tegmark, 2023; Azaria and Mitchell, 2023), as the FFN module is commonly regarded as a knowledge memory (Hernandez et al., 2024)

2024), and Qwen2.5 (7B and 32B; Qwen et al., 2025) for base and fine-tuned versions.

Data collection and training. In order to train f_θ, we need to collect paired data consisting of the LMs' hidden states and binary labels indicating whether it will hallucinate. We construct the training data on four QA datasets from different domains: NQ-open (Wikipedia; Lee et al., 2019), MMLU (STEM; Hendrycks et al., 2020), MedMCQA (medical entrance exam; Pal et al., 2022), and GSM8K (Math; Cobbe et al., 2021). NQ-open is a QA dataset and contains question and answer pairs. MMLU and MedMCQA are multiple choice datasets, pairing each question with multiple options. We convert MMLU and MedMCQA into a QA dataset by pairing each input question with the gold answer. GSM8K consists of grade school math problems, where each problem takes between 2 and 8 steps, we convert GSM8K into a QA dataset by pairing each problem with the final answer.

To collect the training data for LM M, we prompt M with few-shot demonstrations followed by a question, and then collect its hidden states over the inputs and the outputs. M's output answers are checked against gold ones with the exact match (EM), following standard practice (Gao et al., 2023). If the model's output is wrong, its associated hidden states are labeled nonfactual and vice versa, as shown in Figure 2. After producing hidden state and label pairs, we subsample the data to obtain balanced training data containing roughly the same amount of positive (factual) and negative (nonfactual) pairs. In order to compare across different LMs, we create a shared test split across all LMs. Each LM have different training/validation splits. Table 6 in Appendix A summarizes the statistics of the datasets. f_θ is trained with a cross-entropy loss on the input's hidden state and label pairs. Early stopping based on the validation accuracy is used. Other training details are explained in C.1.

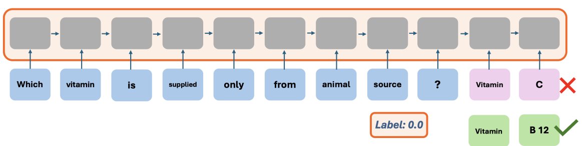


Figure 2: Example of the data collection process. We capture the hidden states over the input and output then label them based on EM with the gold outputs. (§3.1).

We choose to focus on short-answer QA tasks because they allow for unambiguous evaluation using exact match (EM) and controlled experiments. Besides, the short answers allow us to identify exactly where hallucinations begin, as our approach involves preemptively analyzing hidden states before hallucinations occur. More concretely, for a non-factual output, we can use the first token in the wrong answer as the starting point of the hallucination.

3.2 Experiments

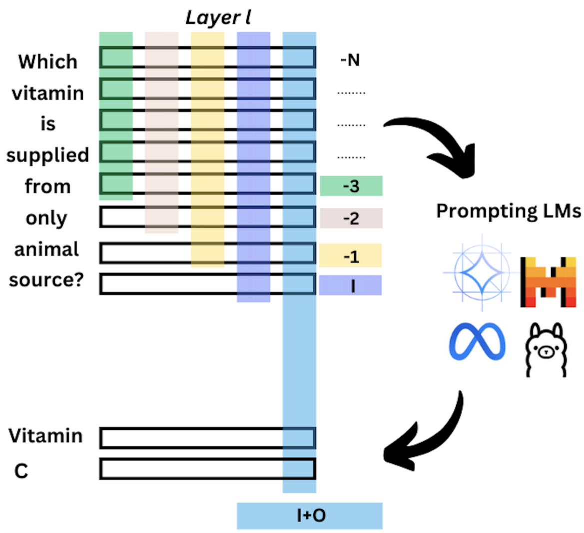


Figure 3: An illustration of different settings used in the experiment. **(Input + Output)** are the hidden states of both the input and output. The subsequent hidden states in the experiment are taken by using only the input or by dropping the last n tokens from the prefix.

Setting. We evaluate f_θ with the settings below:

- **I:** our *preemptive* classifier. It takes the LMs' hidden states produced over the **input questions only**.
- **I-n** takes it even further, restricting FactCheckmate's access to only prefixes of the input questions that exclude the last n tokens. See Figure 3 for illustrative diagrams.
- **I+O** is a reactive setting, and is not to be compared to FactCheckmate because it leverages additional information from a concatenation of both the input questions and the models' outputs and is expected to perform better. Rather, it serves as an approximation of the ceiling performance.

Results. Table 1 shows the hallucination detection test accuracy. We see that across all sizes (7B to 70B), various families (Llama2, Llama3, Mistral, Gemma, Qwen), various types (base, chat, instruct) **I** achieves competitive performance, sometimes even comparable to **I+O** (e.g., only a 1% gap with Llama2-13b on MMLU) This means that the hidden states are capable of predicting the hallucination just by looking at the question, i.e even before the model outputs any incorrect answer.

This confirms that LMs' hidden states provide useful signal for predicting their hallucinations *preemptively*. In some cases, using a prefix of **I-n** underperforms **I**, while for others their performance is comparable. These results suggest that f_θ can often predict whether the LM is likely to hallucinate before it even finishes processing the input questions.

Cross-Model and Cross-Dataset Generalization. Results are shown in Table 2. For the cross-model setting, f_θ is trained on the input hidden states of Llama2-7B, Llama-3.1-8B, and Mistral-7B. These LMs were chosen to ensure consistent embedding sizes. Similar to the previous experiments Table 1 , the accuracy of using only the input hidden states (**I**) remains comparable to using both input and output hidden states (**I+O**) across all three test sets. However, it struggles with model transferability, showing limited effectiveness when applied to unseen models (Table 13 in Appendix G)

For cross-dataset generalization, we train f_θ on a combined dataset spanning NQ-open, MMLU, and MedMCQA. As seen in Table 3, the classifier maintains consistent accuracy for **I** across all three datasets. However, it performs reasonably well in generalizing to out-of-domain datasets.(Table 12 in Appendix G). In summary, training on diverse datasets and models provides an appealing and practical way for FACTCHECKMATE to generalize to various tasks and architectures.

4 FACTCHECKMATE Preemptive Hallucination Mitigation

This section focuses on using FACTCHECKMATE to preemptively mitigate hallucinations, including its intervention model (§4.1) and experiments (§4.2).

4.1 The Intervention Model

When f_θ detects that LM M is about to hallucinate, FACTCHECKMATE relies on an **intervention model** g_ϕ to mitigate hallucinations preemptively. Conditioning on $\boldsymbol{h}_I^{(l)}$, g_ϕ generates a d-dimensional vector and adds it to $\boldsymbol{h}_I^{(l)}$, *before* the LM decodes.

$$\widetilde{\boldsymbol{h}}_I^{(l)} = \boldsymbol{h}_I^{(l)} + \boldsymbol{g}_\phi\left(\boldsymbol{h}_I^{(l)}\right) \tag{1}$$

$\widetilde{\boldsymbol{h}}_I^{(l)}$ is then used in place of $\boldsymbol{h}_I^{(l)}$ for onward LM decoding. In inference, the intervention is applied at the last hidden state of the input $\boldsymbol{h}_I^{(l)}$, as it aligns with the natural progression of decoding and targets the point where hallucinations are most likely to arise.

Intuitively, g_ϕ is supposed to steer the LM's hidden state towards a "target hidden state" $\boldsymbol{h}^{*(l)}$, which is more likely to lead to a factual output.

We explore a deterministic and a stochastic g_ϕ:

- **The deterministic** g_ϕ is a three-layer ReLU-MLP. It trains by minimizing the mean squared error (MSE) between the adjusted hidden state $\widetilde{\boldsymbol{h}}_I^{(l)}$ and the target one $\boldsymbol{h}^{*(l)}$.
- **The stochastic** g_ϕ treats the adjustment vector as a random variable of multivariate Gaussian. It applies a reparameterization trick: $\boldsymbol{g}_\phi(\boldsymbol{h}_I^{(l)}) = \boldsymbol{\mu}(\boldsymbol{h}_I^{(l)}) + \boldsymbol{\epsilon} \odot \boldsymbol{\sigma}(\boldsymbol{h}_I^{(l)})$ for training. Two three-layer ReLU-MLPs are used to for $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$, with the first two layer shared. Its training objective remains the same MSE loss. One benefit of the stochastic g_ϕ is allowing for sampling the adjustment vectors during inference, which we explore in the experiments.

Data collection and training. g_ϕ is trained on pairs of $\boldsymbol{h}_N^{(l)}$, last token over the concatenation of the input and the output of N sequence length, and $\boldsymbol{h}^{*(l)}$. When the LM answers the question correctly, no further modification is needed and $\boldsymbol{h}^{*(l)} = \boldsymbol{h}_N^{(l)}$. However, when the model answers the question incorrectly, we set the $\boldsymbol{h}^{*(l)}$ to the LM's final hidden state *over the input prompt followed by the gold answer*. We construct the training data on two QA datasets: NQ-open (Wikipedia; Lee et al., 2019), and MedMCQA (medical entrance exam; Pal et al., 2022). Other training details are explained in C.1.

4.2 Experiments

Setting. Our preliminary experiments indicate that Exact Match (EM) fails to capture the nuanced improvements introduced by interventions. EM's binary nature overlooks partial corrections, which are common in our setting. For instance, if the gold answer is *May 2024*, the base model outputs *2025*, and the intervened model outputs *May*, EM considers both as equally incorrect. However, the

	NQ					MMLU					MedMCQA					GSM8K				
		Preemptive (ours)					Preemptive (ours)					Preemptive (ours)					Preemptive (ours)			
LM	**I+O**	**I**	**-1**	**-2**	**-3**	**I+O**	**I**	**-1**	**-2**	**-3**	**I+O**	**I**	**-1**	**-2**	**-3**	**I+O**	**I**	**-1**	**-2**	**-3**
Llama2-7B	72.8	71.8	71.1	68.1	65.7	91.7	91.9	91.7	91.7	91.7	77.0	72.9	72.9	72.9	74.5	65.8	66.0	66.0	63.5	63.5
Llama2-13B	74.4	72.0	70.6	71.4	69.7	94.0	93.0	84.1	92.7	85.7	76.0	78.3	78.6	78.3	74.2	68.4	69.1	68.4	66.8	63.8
Llama3-8B	74.9	70.2	68.5	66.8	66.8	93.8	94.0	87.5	87.1	77.3	77.1	76.3	74.3	71.2	67.3	71.3	72.5	72.9	71.3	66.2
Llama3.1-8B	74.3	73.1	70.9	68.9	68.1	94.5	92.3	86.3	80.0	78.0	78.4	76.2	74.9	73.6	69.4	72.3	69.1	61.2	60.2	60.6
Mistral-7B	73.3	72.5	71.4	71.1	70.3	93.2	90.2	83.0	82.5	82.8	77.9	75.4	75.2	73.9	72.8	69.4	70.0	70.0	71.8	71.8
Gemma-7B	80.2	74.5	74.4	74.2	73.9	92.2	96.9	91.3	81.5	89.6	77.0	77.5	74.7	75.2	75.2	70.9	67.4	67.0	67.4	67.8
Qwen2.5-7B	76.1	74.5	72.7	71.2	69.2	94.3	94.0	71.3	85.4	74.4	78.9	76.6	76.9	75.7	74.9	67.0	67.2	67.2	67.2	67.0
Llama2-7B-chat	76.2	74.5	67.5	69.4	69.2	94.8	90.9	79.1	83.3	83.0	81.1	79.3	79.3	79.0	79.3	72.3	73.6	72.2	72.9	72.2
Llama2-13B-chat	74.8	72.4	70.7	70.9	68.5	93.8	93.9	80.1	78.6	92.1	81.3	73.3	70.8	70.0	71.9	72.3	71.9	71.9	71.9	71.9
Llama3-8B-Instruct	81.5	78.6	77.2	76.4	75.0	93.8	95.6	87.4	85.0	79.5	81.4	77.3	72.2	71.1	68.8	74.7	74.3	74.3	74.3	74.3
Llama3.1-8B-Instruct	83.3	74.5	71.3	70.9	66.7	93.1	91.8	86.4	85.4	80.1	81.7	78.8	76.5	71.7	70.1	76.2	78.4	78.0	78.0	78.4
Llama3-70B-Instruct	81.0	77.1	73.3	69.6	65.9	87.6	79.6	76.5	76.4	73.4	74.7	67.6	64.5	63.6	61.0	82.7	78.8	72.5	71.3	69.4
Mistral-7B-Instruct	75.1	74.2	70.9	66.9	65.8	94.3	93.9	89.7	70.7	90.8	76.2	75.9	75.6	76.1	75.3	71.5	72.9	72.9	72.9	72.5
Gemma-7B-it	83.2	75.4	73.5	71.5	68.7	90.2	94.0	78.9	78.8	84.3	77.0	76.2	76.2	75.0	75.2	74.7	75.5	61.1	64.6	70.2
Qwen2.5-7B-Instruct	74.4	72.6	70.2	69.9	66.6	92.5	94.5	85.9	87.8	86.2	82.0	77.5	75.7	73.9	72.3	76.4	80.4	74.8	67.4	67.8

Table 1: Hallucination detection test accuracy. **I+O** indicates a "reactive" baseline that classifies the LMs' hidden states produced over both input questions and output answers, while **I** preemptively classifies hallucinations based on the hidden states over only the inputs. $-n$ indicates that the classifier only sees a prefix of the input excluding the last n tokens.

Model	**I + O**	**I**	**I - 1**	**I - 2**	**I - 3**
Llama 2 7B	78.3	69.2	62.8	61.6	62.7
Llama 3.1 8B	78.6	68.6	64.2	61.7	62.0
Mistral V0.3 7B	80.4	76.4	76.0	75.8	75.7

Table 2: Test accuracy results of different models on various test datasets. The models were trained on Llama-2-7B, Llama-3.1-8B, and Mistral-V0.3-7B on the NQ-open dataset.

Test Data	**I + O**	**I**	**I - 1**	**I - 2**	**I - 3**
NQ-Open	79.2	72.9	68.0	68.1	68.9
MMLU	82.1	69.1	69.4	68.9	67.3
MedMCQA	75.4	59.3	54.2	54.6	54.6

Table 3: Test accuracy results of Llama2-7b-hf after training a multi-data classifier on different test datasets. The model was trained on MMLU, NQ-Open, and MedMCQA.

intervened output is clearly closer to the gold answer. This limitation makes EM uninformative for evaluating interventions that move outputs toward greater factual accuracy, even if they don't perfectly align with the gold answer. Hence, following recent works (Raju et al., 2024; Chen et al., 2024b), we employ GPT-4o (OpenAI et al., 2024) as the evaluator to assess for factuality. See Appendix B for the full prompt. Human evaluation performed by the authors indicate that there is a substantial agreement between GPT-4o and human judgement, with a Cohen's Kappa of 0.6 (substantial agreement), justifying our choice of using GPT-4o as an automatic evaluation metric. For the stochastic $\mathbf{g}_\phi$, we sample 1, 10, 20, and 30 different $\boldsymbol{\epsilon}$, and apply the interventions; we then use f_θ to select the intervened hidden state that leads to the highest probability by f_θ, which is then used for onward decoding.[2] We apply the adjustment only to the first decoding step, modifying $\boldsymbol{h}_I^{(l)}$ to $\widetilde{\boldsymbol{h}}_I^{(l)}$ when the classifier's confidence α is less than or equal to 0.3.

Results. Figure 4 summarizes the performance of FACTCHECKMATE's intervention performance on the NQ-open dataset, including both the deterministic and stochastic variants, with greedy decoding. The intervened LMs consistently outperform the base LMs, with a higher proportion of wins favoring the adjusted outputs. The deterministic intervention consistently achieves a win rate of at least 60% in all cases, while without interventions (Base), the LMs show significantly lower performance, with wins as low as 34%. On average, the winning rate of LMs with intervention across all intervention models is 34.4% higher than that of the base LMs. A similar trend is observed on the MedMCQA dataset; results are provided in E.

[2]A higher probability by f_θ indicates the hidden state is more likely to lead to a factual output.

LM	Inference Time Increase (Δ)
Llama-2-7B	0.94%
Llama-3-8B	1.71%
Llama-3.1-8B	0.96%

Table 4: LMs inference time overheads over three runs per LM. The average relative increase in inference time is approximately 1.2%, showing minimal impact on inference performance §5.1).

	Preemptive				
LM	I+O	I	-1	-2	-3
Llama-2-7B	63.9	52.3	55.3	54.9	55.6

Table 5: Results for the word embedding layer of Llama-2-7b on MedMCQA dataset. (§5.2). The table shows classification accuracy of approximately 50%, indicating no influence of the question difficulty or type on the preemptive hallucination results shown in Table 1.

The results demonstrate that both deterministic and stochastic intervention models improve the factuality of LM's outputs. These finding suggest that, we can mitigate the hallucination even before it shows up in the generation of the LM. Additional qualitative examples are presented in Appendix F.

We note that direct comparison with decoding-based methods such as (Chuang et al., 2024) may not be directly comparable, as they assess factuality after tokens have been generated, whereas FACTCHECKMATE predicts hallucination risk before generation begins. Similarly, existing mitigation methods typically assess factuality and apply corrections on the generated output at different stages, including the hidden states (Zhang et al., 2024; Li et al., 2024b), while FACTCHECKMATE operates purely on input-conditioned hidden states. Due to this fundamental difference, we focus on preemptive detection rather than methods that leverage the full generated output for mitigation. Our goal is to explore the feasibility and potential benefits of early intervention, which may help reduce hallucinations before they occur.

5 Additional Experiments

In the following section, we first evaluate the inference time overhead (§5.1). Next, we investigate the role of word embedding layers (§5.2). Other experiments are detailed in (§H)

5.1 FACTCHECKMATE's Time Overhead

Both f_θ and g_ϕ are lightweight and should incur minimal inference overhead. We confirm this across three models: Llama-2-7B, Llama-3-8B, and Llama-3.1-8B. For each model, the average inference time was measured both with and without FACTCHECKMATE over three runs, each processing 400 few-shot prompts. The results are summarized in Table 4. FACTCHECKMATE introduces a negligible overhead. We see that the result is consistent over models. This negligible overhead is a promising factor for scaling the experiments or integrating it into the existing LMs' pipelines.

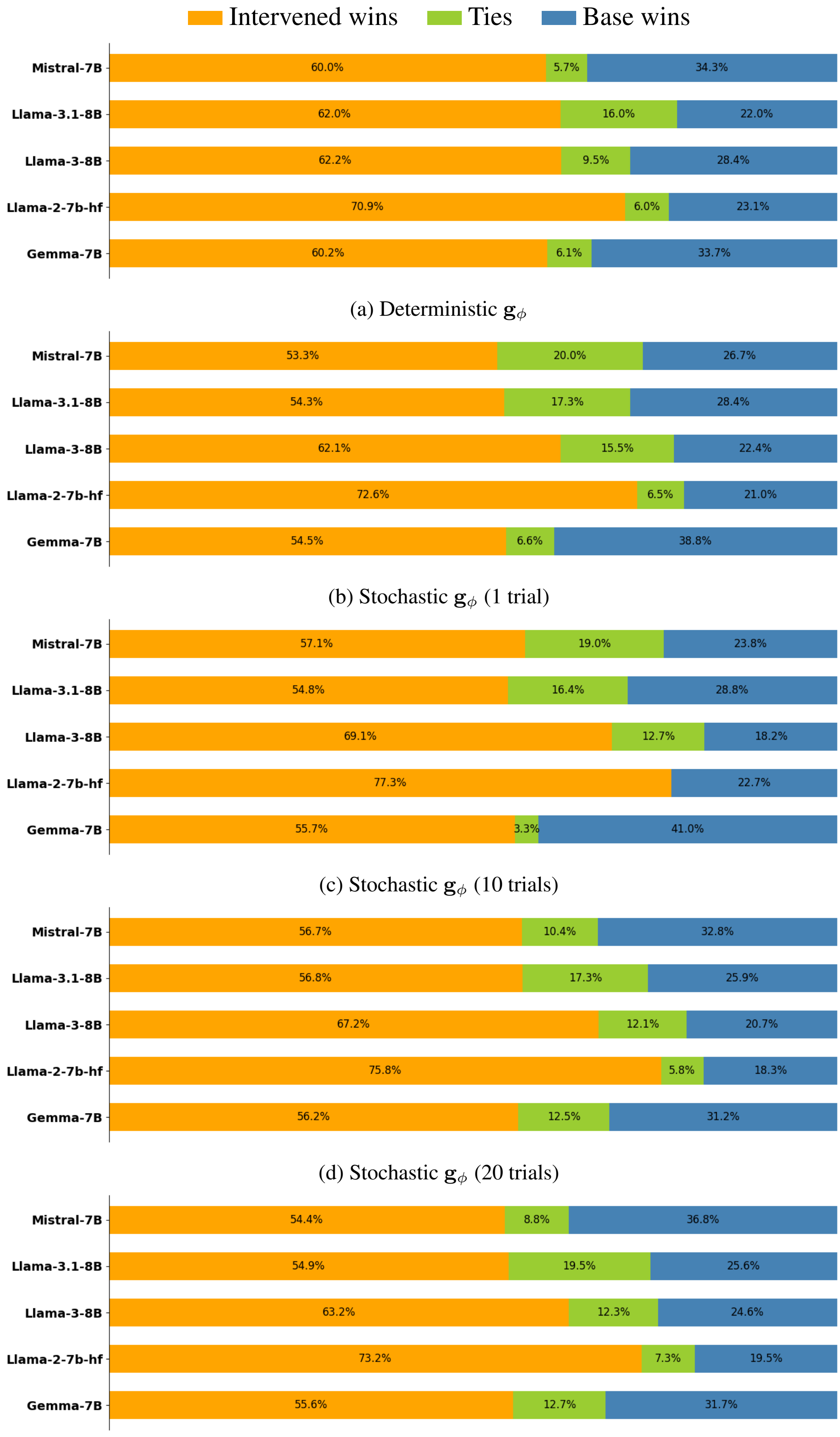
5.2 f_θ Classifies the Hidden States Rather than the Questions

One possible explanation for f_θ's strong preemptive hallucination detection is that it might be classifying the input questions rather than the LMs' hidden states, since intuitively, more difficult questions could lead to a higher chances of hallucinations by the LMs. However, our results indicate that it is the LMs' hidden states, rather than the questions themselves, that drive the success of f_θ.

Table 5 summarizes the test accuracies for an f_θ trained and tested on the word embedding layer of Llama-2-7B, before any contextualization by the LM. Across the board, the accuracies are close to 50% random guess. This confirms that the model is not skewed towards favoring a certain type of question over another while doing the classification. The difficulty of the question is hence, not a contributing factor to the accuracy calculated by classifying the hidden states.

6 Conclusion

In conclusion, FACTCHECKMATE demonstrates that the LMs' hidden states encode rich information that can be used to predict hallucination preemptively, even before they appear in the generated output. Leveraging this insight, we develop a preemptive detection and intervention mechanism that steers the LM's generation towards more factual outputs, once the hallucination is likely to occur. We achieve a preemptive hallucination detection accuracy of more than 70%, and an average of 34.4% more factual output by LMs supported by FACTCHECKMATE, compared to the base LMs. We empirically prove the significant potential of utilizing the internal working of LMs, through learning lightweight models for hallucination detection and mitigation, introducing a negligible average



(a) Deterministic $\mathbf{g}_\phi$

(b) Stochastic $\mathbf{g}_\phi$ (1 trial)

(c) Stochastic $\mathbf{g}_\phi$ (10 trials)

(d) Stochastic $\mathbf{g}_\phi$ (20 trials)

(e) Stochastic $\mathbf{g}_\phi$ (30 trials)

Figure 4: Comparison of FACTCHECKMATE's intervention models. The stochastic model resamples ϵ for 1, 10, 20, and 30 times, f_θ used to select the intervened hidden state that leads to the highest probability by f_θ. Green color indicates tie, orange for the intervened LM wins, and blue for the base LM wins. (§4.2).

overhead of 1.2% increase in the inference time.

7 Limitations and Future work

While our current experiments focus on QA tasks due to their structured evaluation, the core idea behind FACTCHECKMATE leveraging hidden states for preemptive hallucination detection and mitigation could extend to other generation tasks. For tasks like summarization or dialogue systems, we anticipate similar patterns in hidden states when the model is about to generate hallucinated content. However, pinpointing where the hallucination starts to analyze preceding hidden states is challenging and presents a more nuanced problem. Therefore, alternative assessment metrics, such as faithfulness scoring or costly human annotation, would be needed instead of exact match evaluation.

Additionally, we have only looked at the hidden states as an internal component for classification to predict the factuality of a sentence. Exploring other LM's internal components presents a potential direction for future work. This pipeline also would not work for Black-box LMs.

8 Ethics Statement

In developing the FACTCHECKMATE framework, we committed to advancing the ethical use and dependability of large language models (LLMs). We recognize that as LLMs increasingly permeate various aspects of life, ensuring their reliability and truthfulness in generating content is paramount. Thus, our research focused on preemptively detecting factual inaccuracies, aiming to mitigate potential misinformation spread and reduce the propagation of biases present in training data. Furthermore, we meticulously avoided using any data that could potentially compromise individual privacy or confidentiality and ensured our data handling procedures comply with relevant data protection regulations.

Acknowledgments

This research used the Delta advanced computing and data resource which is supported by the National Science Foundation (award OAC 2005572) and the State of Illinois. Delta is a joint effort of the University of Illinois Urbana-Champaign and its National Center for Supercomputing Applications. Also, this research was supported in part by the Illinois Computes project which is supported by the University of Illinois Urbana-Champaign and the University of Illinois System. We thank SambaNova for providing access to Llama models.

References

Amos Azaria and Tom Mitchell. 2023. The internal state of an llm knows when its lying. *arXiv preprint arXiv:2304.13734*.

Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, Quyet V. Do, Yan Xu, and Pascale Fung. 2023. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity. *Preprint*, arXiv:2302.04023.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. 2022. Discovering latent knowledge in language models without supervision. *arXiv preprint arXiv:2212.03827*.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. 2024. Discovering latent knowledge in language models without supervision. *Preprint*, arXiv:2212.03827.

Sky CH-Wang, Benjamin Van Durme, Jason Eisner, and Chris Kedzie. 2024. Do androids know they're only dreaming of electric sheep? *Preprint*, arXiv:2312.17249.

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

Zhaorun Chen, Yichao Du, Zichen Wen, Yiyang Zhou, Chenhang Cui, Zhenzhen Weng, Haoqin Tu, Chaoqi Wang, Zhengwei Tong, Qinglan Huang, et al. 2024b. Mj-bench: Is your multimodal reward model really a good judge for text-to-image generation? *arXiv preprint arXiv:2407.04842*.

I-Chun Chern, Steffi Chern, Shiqi Chen, Weizhe Yuan, Kehua Feng, Chunting Zhou, Junxian He, Graham Neubig, and Pengfei Liu. 2023. Factool: Factuality detection in generative ai – a tool augmented framework for multi-task and multi-domain scenarios. *Preprint*, arXiv:2307.13528.

Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James Glass, and Pengcheng He. 2024. Dola: Decoding by contrasting layers improves factuality in large language models. *Preprint*, arXiv:2309.03883.

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

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. 2020. Plug and play language models: A simple approach to controlled text generation. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Shehzaad Dhuliawala, Mojtaba Komeili, Jing Xu, Roberta Raileanu, Xian Li, Asli Celikyilmaz, and Jason Weston. 2023. Chain-of-verification reduces hallucination in large language models. *Preprint*, arXiv:2309.11495.

Hanyu Duan, Yi Yang, and Kar Yan Tam. 2024. Do llms know about hallucination? an empirical investigation of llm's hidden states. *Preprint*, arXiv:2402.09733.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. 2023. A framework for few-shot language model evaluation.

Zhibin Gou, Zhihong Shao, Yeyun Gong, Yelong Shen, Yujiu Yang, Nan Duan, and Weizhu Chen. 2024. Critic: Large language models can self-correct with tool-interactive critiquing. *Preprint*, arXiv:2305.11738.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Evan Hernandez, Belinda Z. Li, and Jacob Andreas. 2024. Inspecting and editing knowledge representations in language models. *Preprint*, arXiv:2304.00740.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2023. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *Preprint*, arXiv:2311.05232.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023a. Survey of hallucination in natural language generation. *ACM Comput. Surv.*, 55(12).

Ziwei Ji, Tiezheng Yu, Yan Xu, Nayeon Lee, Etsuko Ishii, and Pascale Fung. 2023b. Towards mitigating hallucination in large language models via self-reflection. *Preprint*, arXiv:2310.06271.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Muhammad Khalifa, Lajanugen Logeswaran, Moontae Lee, Honglak Lee, and Lu Wang. 2023. Grace: Discriminator-guided chain-of-thought reasoning. *arXiv preprint arXiv:2305.14934*.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6086–6096, Florence, Italy. Association for Computational Linguistics.

Junyi Li, Jie Chen, Ruiyang Ren, Xiaoxue Cheng, Wayne Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2024a. The dawn after the dark: An empirical study on factuality hallucination in large language models. *arXiv preprint arXiv:2401.03205*.

Junyi Li, Xiaoxue Cheng, Wayne Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2023. Halueval: A large-scale hallucination evaluation benchmark for large language models. *Preprint*, arXiv:2305.11747.

Kenneth Li, Oam Patel, Fernanda ViÃl'gas, Hanspeter Pfister, and Martin Wattenberg. 2024b. Inference-time intervention: Eliciting truthful answers from a language model. *Preprint*, arXiv:2306.03341.

Potsawee Manakul, Adian Liusie, and Mark J. F. Gales. 2023a. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. *Preprint*, arXiv:2303.08896.

Potsawee Manakul, Adian Liusie, and Mark JF Gales. 2023b. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. *arXiv preprint arXiv:2303.08896*.

Samuel Marks and Max Tegmark. 2023. The geometry of truth: Emergent linear structure in large language model representations of true/false datasets. *arXiv preprint arXiv:2310.06824*.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in gpt. *Advances in Neural Information Processing Systems*, 35:17359–17372.

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. *Preprint*, arXiv:2305.14251.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany

Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, SimÃşn Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, ÅĄukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, ÅĄukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David MÃl'ly, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe CerÃşn Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Hadas Orgad, Michael Toker, Zorik Gekhman, Roi Reichart, Idan Szpektor, Hadas Kotek, and Yonatan Belinkov. 2024. Llms know more than they show: On the intrinsic representation of llm hallucinations. *Preprint*, arXiv:2410.02707.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. 2022. Medmcqa: A large-scale multi-subject multi-choice dataset for medical domain question answering. In *Conference on health, inference, and learning*, pages 248–260. PMLR.

Liangming Pan, Michael Saxon, Wenda Xu, Deepak Nathani, Xinyi Wang, and William Yang Wang. 2023. Automatically correcting large language models: Surveying the landscape of diverse self-correction strategies. *Preprint*, arXiv:2308.03188.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Ravi Raju, Swayambhoo Jain, Bo Li, Jonathan Li, and Urmish Thakkar. 2024. Constructing domain-specific evaluation sets for llm-as-a-judge. *arXiv preprint arXiv:2408.08808*.

Vipula Rawte, Amit Sheth, and Amitava Das. 2023. A survey of hallucination in large foundation models. *Preprint*, arXiv:2309.05922.

Weihang Su, Changyue Wang, Qingyao Ai, Yiran HU, Zhijing Wu, Yujia Zhou, and Yiqun Liu. 2024. Unsupervised real-time hallucination detection based on the internal states of large language models. *Preprint*, arXiv:2403.06448.

Nishant Subramani, Nivedita Suresh, and Matthew E. Peters. 2022. Extracting latent steering vectors from pretrained language models. *Preprint*, arXiv:2205.05124.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. 2024. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, TimothÃl'e Lacroix, Baptiste RoziÃÍre, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023a. Llama: Open and efficient foundation language models. *Preprint*, arXiv:2302.13971.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023b. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Ziwei Xu, Sanjay Jain, and Mohan Kankanhalli. 2024. Hallucination is inevitable: An innate limitation of large language models. *arXiv preprint arXiv:2401.11817*.

Hongbin Ye, Tong Liu, Aijia Zhang, Wei Hua, and Weiqiang Jia. 2023. Cognitive mirage: A review of hallucinations in large language models. *Preprint*, arXiv:2309.06794.

Shaolei Zhang, Tian Yu, and Yang Feng. 2024. Truthx: Alleviating hallucinations by editing large language models in truthful space. *Preprint*, arXiv:2402.17811.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, et al. 2023. Siren's song in the ai ocean: a survey on hallucination in large language models. *arXiv preprint arXiv:2309.01219*.

Chunting Zhou, Graham Neubig, Jiatao Gu, Mona Diab, Paco Guzman, Luke Zettlemoyer, and Marjan Ghazvininejad. 2021. Detecting hallucinated content in conditional neural sequence generation. *Preprint*, arXiv:2011.02593.

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

A Dataset size

B Factual Assessment Prompt

To assess factual accuracy, we use GPT-4o (OpenAI et al., 2024) as the evaluator. To reduce stochasticity in the prompting process, we set the temperature to 1×10^{-14} and top_p to 1×10^{-17}. The prompt used for evaluation is as follows:

> **System:** *You are an expert evaluator with an access to Google Search. Your task is to evaluate two responses to a question for factual accuracy. For this task, 'Factual accuracy' refers to the correctness and relevance of the information, aligned with facts accepted or verified as recent as 2021. Ignore stylistic differences, length, opinions, or phrasing unless they change the factual meaning. Supported by your Google Search results, decide which response, if any, is correct. Answer 'first' if the first response is the only correct response, 'second' if the second response is the only correct response, 'both' if both responses are correct, or 'neither' if neither response is correct or if the information provided is ambiguous or insufficient for making a decision, You should favor the response that shows uncertainty if the other response is incorrect. Then, in a new line, briefly explain the reason.*
> **User**: *Question:* who played first game in world cup 2018? *First Response:* Russia vs Saudi Arabia *Second Response:* Brazil vs Germany.

C Experiments for classification

C.1 Hidden Representation Classification Analysis

Given the datasets and models described above, for every layer in a model we train a corresponding classifier on hidden states of that respective layer. We use three modes for aggregating the hidden states before passing them to the classifier: mean pooling, max pooling and taking the last token in the hidden states. Figure 5b illustrates the accuracy of hallucination detection of the classifiers for the entire sequence, using the mean token representation for aggregation. As shown, the accuracy across all evaluated models mostly exceeds 0.75, indicating a robust capability to identify hallucinations. This high level of performance underscores the efficacy of the hidden state representations in distinguishing factual accuracies within generated content. As seen in the figure, we see that the accuracy peaks for the middle layers. The best per-

Dataset	**Total Size**	**Train (70%)**	**Validation (15%)**	**Test (15%)**
NQ-Open (Lee et al., 2019)	6,666	4,666	1,000	1,000
MMLU (Hendrycks et al., 2020)	1,844	1,292	276	276
MedMCQA (Pal et al., 2022)	2,000	1,400	300	300
GSM8K (Cobbe et al., 2021)	1,040	728	156	156

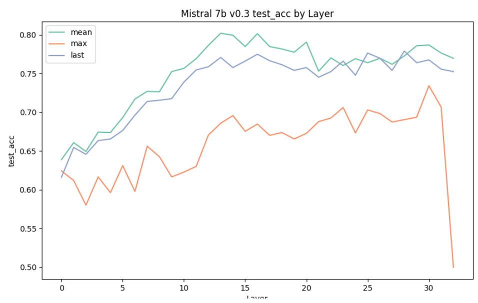
Table 6: Dataset splits and sizes for training the hallucination classifier f_θ over the LMs' hidden states (§3.1).

forming layer per model and dataset is shown in Table 7.

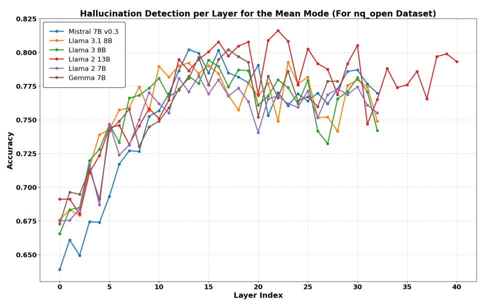
Therefore, for all models we calculate the test accuracy across all layers and all modes of aggregation. Quantitative results are shown in the first column of Table 1. Given the three modes of aggregation, we see that mean pooling gives the best results in most cases. Figure 5a shows the test accuracy per layer and mode.

Classifier Setup: The classifier f_θ is two-layer MLP with ReLU activations and BCELoss, trained using an Adam optimizer with a learning rate of 10^{-4} with a dropout rate of 0.1. We train all classifiers for 50 epochs and apply early stopping based on the validation accuracy.

Intervention Setup: The intervention model $\mathbf{g}_\phi$ is a three-layer RELU MLP. It is trained using an Adam optimizer. MSE loss is used between the altered hidden state and original hidden state. We train it for 100 epochs.



(a) Test Accuracy by Layer for all modes for the Mistral-7b



(b) Accuracies for entire sentence across models and layers.

LM	**NQ**	**MMLU**	**MedMCQA**
Llama-2-7b-hf	14	16	14
Llama-2-13b-hf	22	15	14
Llama-3-8B	15	17	11
Llama3.1-8B	23	14	15
Mistral-7B	13	-	12
Gemma-7B	17	17	18
Llama2-7B-chat	14	13	14
Llama2-13B-chat	19	14	16
Llama3-8B-Instruct	15	15	12
Llama-3.1-8B-Instruct	15	13	14
Llama3-70B-Instruct	74	35	75
Mistral-7B-Instruct	18	15	18
Gemma-7B-it	16	18	18

Table 7: Best Performing layer per model and dataset

		Preemptive			
LM	**I+O**	**I**	**-1**	**-2**	**-3**
Llama3-70B	78.0	77.2	76.8	76.8	76.5
Qwen2.5-32B	79.8	76.3	75.9	75.5	75.9
Llama3.1-70B-Instruct	85.3	81.0	77.6	70.9	77.2

Table 8: Results for the gsm8k dataset on the following models

		Preemptive			
LM	**I+O**	**I**	**-1**	**-2**	**-3**
Qwen2.5-32B	92.7	90.8	86.9	87.3	74.4
Qwen2.5-32B-Instruct	92.7	92.1	83.7	86.2	83.8

Table 9: Results for the mmlu dataset on the following models

C.2 Additional Experiments

D Impact on Nominal Questions

A key consideration is whether the proposed intervention method impacts nominal, non-hallucinatory questions, particularly given the classifier's false positive rate (FPR). To address this, we conducted a detailed analysis of non-hallucinatory responses generated by the Llama 3.1-8B model on the MedMCQA dataset after intervention.

The classifier achieved robust performance metrics: Precision: 0.987, Recall: 0.571, F1-Score: 0.723, and Accuracy: 0.623. Among the 1,362 responses analyzed, 9 cases were identified as false

Table 10: Confusion Matrix for Llama 3.1-8B Model After Intervention

	Actual	
	Positive	**Negative**
Predicted Positive	672 (TP)	9 (FP)
Predicted Negative	505 (FN)	176 (TN)

positives, where the intervention occurred despite the absence of hallucination. Importantly, only 2 out of these 9 false positives led to a degradation in the factual quality of the generated output. This finding highlights the minimal disruptive impact of our intervention on non-hallucinatory responses.

Furthermore, interventions correctly enhanced factual outputs in 672 true positive cases, demonstrating the classifier's ability to effectively improve factuality while minimizing unnecessary disruptions. This cautious approach ensures that the vast majority of factual responses remain unaffected, while significant improvements are achieved for hallucinatory responses.

These results alleviate concerns about the classifier's FPR by showing that the proposed method maintains high reliability, minimally impacting nominal questions while effectively enhancing the factuality of hallucinatory outputs.

E MedMCQA Dataset Intervention Results

Results. Figure 6 presents the results of the deterministic $\mathbf{g}_\phi$ on the MedMCQA dataset. The trends are consistent with those observed for NQ-open: the intervened LMs consistently outperform the base LMs. These results further highlight the effectiveness of FACTCHECKMATE's intervention model in mitigating hallucinations and improving factuality across several domains.

F Qualitative Examples

G Generalization results

H Ablation Study

H.1 Preemptive Hallucination Detection across various Aggregation Methods

We explore three modes for aggregating the hidden states : mean pooling, max pooling, and taking the last token. We see that the mean pooling shows the best accuracy as shown in Fig 5a. To test how different modes of aggregation work for the preemptive experiments, we compare all the three modes. This is done across the same layer for a the same model. As shown in Table 14, we see that the accuracy of the entire sentence (**I+O**) is similar for last token and mean pooling. However, the drop in the subsequent accuracies is the maximum when last token is used. The maximum accuracy for **I** is when mean pooling is used. Therefore, we use mean pooling as our mode of aggregation in all our experiments.

H.2 Ablation Study: Classifier-Based Sampling without Intervention

Sample-FACTCHECKMATE-CLS refers to a sampling-based decoding approach that leverages the hallucination classifier f_θ component of FACTCHECKMATE to select the sample with the highest classifier accuracy, without applying intervention. To evaluate its effectiveness, we compare Sample-FACTCHECKMATE-CLS against the native sampling of the non-intervened LM, using the same random seed for fair comparison, with top-100. As shown in Figure 7, while Sample-FACTCHECKMATE-CLS improves performance in certain cases, it underperforms in others. This highlights the necessity of intervention to effectively mitigate hallucinations and achieve consistent factual improvements.

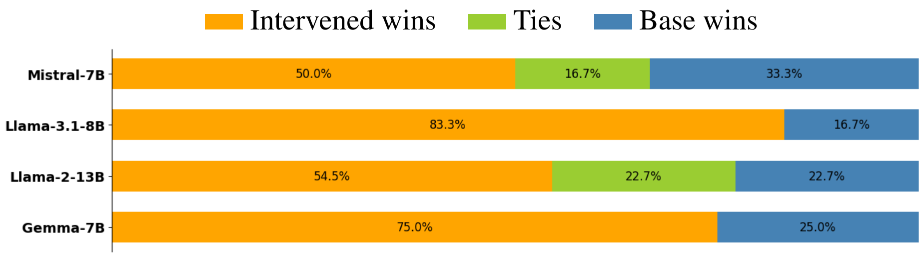
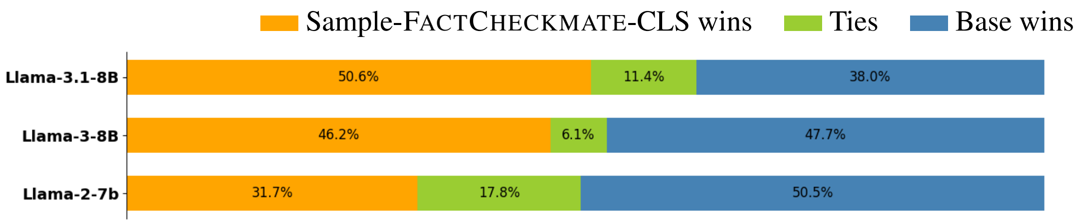


Figure 6: Comparison of FACTCHECKMATE's deterministic g_ϕ intervention model on the MedMCQA dataset, under greedy decoding. Green color indicated tie, orange for the intervened LM, and blue for the base LM. Similar to the NQ results, the intervened LMs achieve significantly higher win rates compared to the base LMs. (§E)



(a) Sample-FACTCHECKMATE-CLS

Figure 7: The figure shows the winning rate of the Sample-FACTCHECKMATE-CLS LM (Orange), the base LM (Blue), and the ties (Green) (§H.2).

Question	**Base Answer**	**Answer after Intervention**	**Explanation**
When was the last time the death penalty was used in PA?	June	(1999)	The last execution in Pennsylvania was in 1999. The second response is correct.
How many Black Eyed Pea restaurants are there?	Over one hundred	0	The Black Eyed Pea restaurant chain has significantly reduced its number of locations, and as of recent years, it has closed all of its locations. Therefore, the second response is correct.
When does Jess come back on New Girl?	Season six	Season five	Jess returns to New Girl in season five after a brief absence.
Who plays Angela's father on Boy Meets World?	William Daniels	(Unknown)	The character of Angela's father on Boy Meets World is played by Julius Carry, not William Daniels. William Daniels played Mr. Feeny on the show. The second response is correct in indicating uncertainty as it does not provide incorrect information.

Table 11: Qualitative Examples. Comparison of base and intervened answers with their explanations from GPT4o justifying its choice.

Model used	Training Data	Test Data	I + O	I	I - 1	I - 2	I - 3
Llama2-7b-hf	MMLU + NQ-Open	NQ-Open	75.6	70.8	67.0	67.4	68.6
		MMLU	80.0	72.3	70.7	68.7	68.9
		MedMCQA	63.2	58.8	58.7	59.2	58.5

Table 12: Test accuracy results of Llama2-7b-hf on different test datasets and out-of-domain data.

Trained on	Tested on	I + O	I	I - 1	I - 2	I - 3
Llama 2 7B	Llama 3.1 8B	48.4	48.4	48.4	48.4	48.4
	Mistral V0.3 7B	51.5	51.5	51.5	51.5	51.5
Llama 3.1 8B	Llama 2 7B	47.7	48.3	48.3	48.4	48.3
	Mistral V0.3 7B	51.5	51.5	51.5	51.5	51.5
Mistral V0.3 7B	Llama 2 7B	48.8	48.4	48.4	48.4	48.4
	Llama 3.1 8B	48.5	48.4	48.4	48.4	48.4

Table 13: Performance results of different models on various test datasets.

	Mean Pooling					Last Token					Max Pooling				
		Preemptive					Preemptive					Preemptive			
LM	I+O	I	-1	-2	-3	I+O	I	-1	-2	-3	I+O	I	-1	-2	-3
Llama3-8B	79.4	**75.9**	73.4	72.2	71.6	81.7	71.0	63.3	51.8	51.3	73.1	70.5	69.9	68.8	68.9

Table 14: Comparison of hallucination classification across different aggregation modes for the same layer and LM. We show the results for the Llama3-8B on layer 15. We see that the difference between **I+O** and **I** is the least when the mean is the mode of aggregation.