PALIMPSEST: BIDIRECTIONAL SCORING FOR SPAN-LEVEL HALLUCINATION DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models can produce "hallucinated" statements that sound fluent but are ungrounded in reality, motivating reliable detection. Even though support for a span of statement can depend on words that follow it, many evaluators score text left to right. Motivated by this shortcoming, we propose PALIMPSEST, a span-level detector that uses masked diffusion language models (MDLMs), which evaluates the factuality of a span of text using its context from both sides. We compare PALIMPSEST to autoregressive (AR) models with AUROC and find a) on BUMP (Ma et al., 2023) with human-annotated spans, 8B MDLM scorers match the strongest tested AR model, which has 3.9 times as many nominal parameters, and b) on LLM-AggreFact (Tang et al., 2024) they achieve comparable performance to the strongest tested AR model (0.85 vs. 0.89 AUROC) while using 3.9× fewer nominal parameters. On HalluEntity (Yeh et al., 2025), `LLaDA-8B-Base` outperforms the strongest tested AR scorer, a 31B checkpoint. Thus, PALIMPSEST can rival larger AR-based methods for span-level hallucination detection.

1 INTRODUCTION

Large language models (LLMs) have become remarkably capable and ubiquitous (Wei et al., 2022; Liang et al., 2025). They can scaffold a personal website, solve some of the hardest problems at the International Collegiate Programming Contest (ICPC) World Finals (Lin & Cheng, 2025; International Collegiate Programming Contest, 2025), and, according to a recent OpenAI announcement that has not yet been peer reviewed, help construct a finite-time blow-up solution to a forced version of the Navier–Stokes equations (OpenAI, 2026). Yet these models share an Achilles' heel: ***hallucination***, the generation of fluent but unsupported statements (Ji et al., 2023; Huang et al., 2025). In creative tasks, hallucination can be tolerable or even beneficial, since it may make the output more inventive (Sui et al., 2024). In high-stakes applications such as medicine (Pal et al., 2023), law (Dahl et al., 2024; Magesh et al., 2025), finance (Kang & Liu, 2023), and even news summarization (Maynez et al., 2020), however, a single hallucination can render the output unusable or even unsafe. Detecting hallucinations is therefore an urgent problem.

Studies of LLM hallucination attribute these errors, at least in part, to insufficient parametric knowledge (Kandpal et al., 2023; Mallen et al., 2023). To compensate, retrieval-augmented generation (RAG) (Lewis et al., 2020) supplies the LLM with task-relevant external knowledge on which to condition its generation. RAG is not a panacea, however: although it reduces the likelihood of hallucination (Shuster et al., 2021), hallucinations still occur (Niu et al., 2024; Magesh et al., 2025) and can cause harm in critical applications, which makes hallucination detection in the RAG setting essential. Moreover, rather than merely flagging whether a claim or sentence contains a hallucination, a detector should ideally *pinpoint the exact span of text that is hallucinated* (Mishra et al., 2024), so that a human verifier can check the detection and, if necessary, correct the error. This motivates the goal of this paper:

Goal: Develop a span-level hallucination detection method for retrieval-augmented LLMs.

Given a span of text, a natural way to evaluate it is to use a language model. Many such evaluators, however, are autoregressive (AR) (Liu et al., 2023; Fu et al., 2024): they score text from left to right, even though whether a span is supported can depend on the words that follow it. Masked diffusion

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

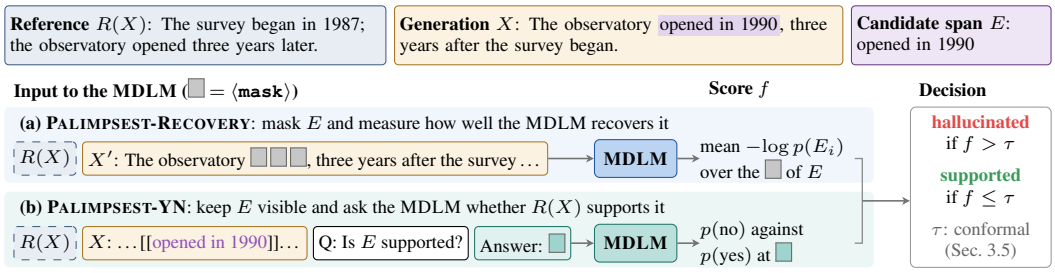


Figure 1: Overview of PALIMPSEST. Given a generation X, a reference $R(X)$ where available, and a candidate span E, (a) PALIMPSEST-RECOVERY scores how well the masked tokens are recovered from bidirectional context, while (b) PALIMPSEST-YN predicts whether E is supported via a masked yes/no query. Both produce a score f, where higher values indicate a greater likelihood of hallucination. The threshold τ is calibrated using split-conformal calibration (Section 3.5).

language models (MDLMs) (Sahoo et al., 2024; Nie et al., 2026), in contrast, condition on both sides of a masked position. This raises a natural question: can the bidirectional conditioning of MDLMs be turned into a better signal for judging whether a span is supported? Concretely, we make three contributions:

1. **A novel diffusion-based hallucination detector.** We propose PALIMPSEST (Section 3), a *novel* family of span-level hallucination detection methods based on MDLMs. Figure 1 provides a methodological overview of PALIMPSEST.

2. **Strong performance at a fraction of the cost.** We show that PALIMPSEST is not only competitive in terms of performance, but also more efficient compared to baselines that are significantly larger in scale via extensive empirical studies across 3 datasets (Section 5).

3. **Isolating the benefit of bidirectionality.** Through a careful controlled comparison between an MDLM model and the corresponding AR model, we isolate the effect of bidirectional conditioning, offering evidence that MDLMs are a strong foundation for hallucination detection (Section 5).

2 PRELIMINARIES ON MASKED DIFFUSION LANGUAGE MODELS

MDLMs (Sahoo et al., 2024; Lou et al., 2024; Nie et al., 2026) have recently emerged as an alternative to AR language modeling, due to their high token throughput (Inception Labs et al., 2025; Song et al., 2025; Wu et al., 2025). We briefly review how they work and differ from AR models below.

Let $X = (x_1, x_2, \ldots, x_n)$ denote a sequence of n tokens. A conventional AR model p_{AR} (Radford et al., 2019; Brown et al., 2020) factorizes the probability of x from left to right, so each token is conditioned only on the tokens before it:

$$p_{\text{AR}}(X) = \prod_{k=1}^{n} p_{\text{AR}}(x_k \mid x_1, \ldots, x_{k-1}).$$

As such, generation proceeds sequentially: a new token x_{n+1} is drawn from $p_{\text{AR}}(\cdot \mid x_1, \ldots, x_n)$.

MDLMs, on the other hand, are diffusion-based generative models (Austin et al., 2021; Shi et al., 2024). Like image diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020), they generate text by denoising; here, the noise consists of ⟨mask⟩ tokens, which can appear anywhere in the sequence. Let $\tilde{x}$ denote a partially masked copy of x in which the positions in a subset $M \subseteq \{1, \ldots, n\}$ are replaced by ⟨mask⟩. An MDLM p_{MDLM} predicts the token at each masked position $k \in M$ from all unmasked tokens of $\tilde{x}$, on *both* sides of k:

$$p_{\text{MDLM}}(x_k \mid \tilde{x}), \qquad k \in M.$$

Generation, or denoising, replaces masked positions with tokens predicted from these distributions. PALIMPSEST exploits this bidirectional conditioning to score candidate spans, as we describe next in Section 3.

3 METHOD: PALIMPSEST AND BASELINES

Compared with AR models, an MDLM predicts a token from both its preceding and following tokens (Section 2). This bidirectionality lets the MDLM attend to the entire context around the predicted token, as in masked language modeling (Devlin et al., 2019). Inspired by this, we treat hallucination as noise, and replace a span with $\langle\texttt{mask}\rangle$ tokens and let the MDLM "denoise" it from the surrounding context.

This intuition motivates our method, PALIMPSEST, named after a palimpsest, a writing surface on which earlier text has been erased to make room for new writing. Our first variant, PALIMPSEST-RECOVERY, builds directly on this idea. We also introduce PALIMPSEST-YN, a second MDLM-based variant that elicits a verbalized yes/no judgment. Below, we formalize the problem, describe both variants, and present the baselines and a statistical procedure for turning scores into decisions (conformal method). Figure 2 summarizes the four resulting scores, S1, S2, F1, and F2.

3.1 PROBLEM SETTING AND NOTATION

Let $X = (x_1, x_2, \ldots, x_n)$ denote a generation of n tokens produced by an LLM, and let $R(X)$ denote a reference text related to X. Let $E = (x_i, x_{i+1}, \ldots, x_j)$, where $1 \le i \le j \le n$, denote a contiguous subsequence (span) of X. Our goal is to determine whether E contains a hallucination. Concretely, we develop functions f such that

$$f(E, X, R(X)) \in \mathbb{R}, \quad \text{where higher } f \iff \text{higher hallucination likelihood.}$$

PALIMPSEST-RECOVERY and PALIMPSEST-YN are two implementations of f. Given the value of f, we apply a conformal method (Section 3.5) to select a threshold τ to flag hallucination:

$$f(E, X, R(X)) > \tau \iff \text{hallucination.}$$

3.2 PALIMPSEST-RECOVERY

Intuition. We replace the tokens of a span with $\langle\texttt{mask}\rangle$ tokens and measure how well the MDLM recovers the original tokens given the reference. If the MDLM recovers them well when conditioned on the reference, the span is less likely to contain a hallucination.

Formally, given a reference $R(X)$, a generation X, and a span E of X, we replace each token of E in X with a $\langle\texttt{mask}\rangle$ token. Consider the following example:

$R(X)$: *The survey began in 1987; the observatory opened three years later.*

X : *The observatory opened in 1990, three years after the survey began.*

E : *opened in 1990*

Replacing E with $\langle\texttt{mask}\rangle$ tokens[1] yields

X' : *The observatory* $\langle\texttt{mask}\rangle_1\langle\texttt{mask}\rangle_2\langle\texttt{mask}\rangle_3$*, three years after the survey began.*

We then define

$$f_{S1}(E, X, R(X)) = -\frac{1}{|E|}\sum_{i=1}^{|E|} \log p_{\text{MDLM}}(\langle\texttt{mask}\rangle_i = E_i \mid R(X), X'), \tag{S1}$$

where $|E|$ is the number of tokens in E and E_i is the i-th token of E. S1 scores all masked tokens in a single forward pass, so each token of E is predicted without seeing any other token of E. Alternatively, mirroring how an MDLM generates text by unmasking tokens progressively (Section 2), the tokens of E can be revealed one at a time. In that case, the conditional probability of each token depends on which other tokens of E are already visible; for example, the following two conditional probabilities differ:

$$p_{\text{MDLM}}(\langle\texttt{mask}\rangle_1 = \textit{opened} \mid R(X), \textit{The observatory } \langle\texttt{mask}\rangle_1 \textit{ in } \langle\texttt{mask}\rangle_3, \ldots)$$
$$\neq$$
$$p_{\text{MDLM}}(\langle\texttt{mask}\rangle_1 = \textit{opened} \mid R(X), \textit{The observatory } \langle\texttt{mask}\rangle_1\langle\texttt{mask}\rangle_2 \textit{ 1990}, \ldots)$$

[1] For simplicity, we assume that each word is a single token.

To account for this order dependence, let $\pi = (\pi_1, \pi_2, \ldots, \pi_{|E|})$ be a (random) ordering in which the token recovery probabilities are evaluated and let $X'(t, \pi)$ denote the sequence after the first t tokens in π have been revealed, so that $X'(0, \pi) = X'$. For example, if $\pi = (3, 1, 2)$, then $X'(2, \pi) = \ldots$ *opened* $\langle\texttt{mask}\rangle_2$ *1990* $\ldots$. The order-dependent score is then

$$f_{S_2}(E, X, R(X), \pi) = -\frac{1}{|E|} \sum_{i=1}^{|E|} \log p_{\text{MDLM}}(\langle\texttt{mask}\rangle_{\pi_i} = E_{\pi_i} \mid R(X), X'(i-1, \pi)). \tag{S2}$$

Let Π denote the set of all possible ordering of length $|E|$. Then, ideally, we would like to calculate

$$\frac{1}{|E|!} \sum_{\pi \in \Pi} f_{S_2}(E, X, R(X), \pi).$$

However, calculating all $|E|!$ is computationally hard. Therefore, we randomly sample four orderings and report the average of the four f_{S_2}.

3.3 PALIMPSEST-YN

Whereas PALIMPSEST-RECOVERY hides the span and measures how well it can be reconstructed, PALIMPSEST-YN keeps the span visible and asks the MDLM directly whether it is supported. This is analogous to verbalized scoring with AR models: we ask the MDLM to unmask a single $\langle\texttt{mask}\rangle$ token into either "yes" or "no." Formally, let $z_{\text{yes}} = p_{\text{MDLM}}(\texttt{<mask>} = \text{yes} \mid E, X, R(X))$ and $z_{\text{no}} = p_{\text{MDLM}}(\texttt{<mask>} = \text{no} \mid E, X, R(X))$. The score is then

$$f_F(E, X, R(X)) = \frac{\exp z_{\text{no}}}{\exp z_{\text{yes}} + \exp z_{\text{no}}}. \tag{F}$$

Because the MDLM attends to the context on both sides of the $\langle\texttt{mask}\rangle$ token, the answer need not be placed at the end of the prompt, and its position becomes a critical design choice. We therefore consider two placements. The first format, F1, places the $\langle\texttt{mask}\rangle$ token at the end of the prompt, whereas the second, F2, places it directly after the reference $R(X)$, before the question and the generation X. Figure 7 in the Appendix B.1 shows an example prompt in each format.

Figure 2 gives a quick visual summary of the four scores in PALIMPSEST.

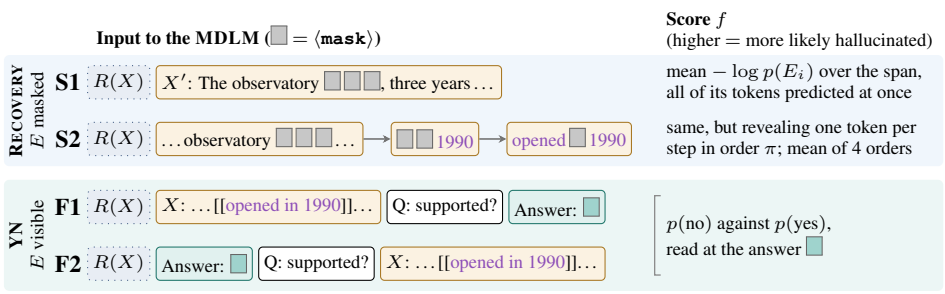


Figure 2: The four PALIMPSEST scores at a glance, on the example of Section 3.2 (E = *opened in 1990*). PALIMPSEST-RECOVERY masks E and measures how the MDLM recovers it, either all at once (S1) or one token at a time in a random order π (S2). PALIMPSEST-YN keeps E visible, marked with [[ ]], and reads a yes/no judgment at a masked answer slot placed after the question (F1) or directly after the reference (F2). Equations S1, S2, and F give the exact scores.

3.4 BASELINES

We compare PALIMPSEST with two types of baselines: (1) detectors built on various AR models; and (2) dedicated hallucination detectors. For AR models, we consider two variants.

The first variant, R1, gives the AR model the reference $R(X)$, the generation X, and the span E, together with a 5-point rubric for scoring how well the reference supports the span, in the style of LLM-as-a-judge evaluation. We defer the R1 prompt template to Appendix B.2.

The second variant, R2, mimics the recovery mechanism of PALIMPSEST-RECOVERY. We prompt the AR model with the reference $R(X)$ and the generation X, in which the span E is replaced with underscores, and ask the model to produce the missing text. The score is the negative log-likelihood of E given this prompt. We defer the R2 prompt template to Appendix B.3.

Following prior work that uses natural language inference (NLI) to detect factual inconsistencies, we also include two NLI models, `DeBERTa-v3-base` (He et al., 2023; Laurer et al., 2024) and `roberta-large-mnli` (Liu et al., 2019; Williams et al., 2018), and two trained fact-checkers, `MiniCheck-Flan-T5-Large` (Tang et al., 2024; Chung et al., 2024) and `AlignScore-large` (Zha et al., 2023).

Finally, as an AR log-likelihood-based baseline, we use Focus (Zhang et al., 2023), a reference-free uncertainty method that combines token surprisals and entropies with keyword gating, inverse-document-frequency weighting, and attention propagation; we run it with `Llama-3.1-8B` (Grattafiori et al., 2024).

3.5 CONFORMAL THRESHOLDING

Each method produces a real-valued score, which we threshold to decide whether a span is hallucinated. To this end, we use split-conformal calibration (Papadopoulos et al., 2002; Lei et al., 2018; Angelopoulos & Bates, 2023). Concretely, we hold out 20% of the training data as a calibration set. Let n_{cal} denote the number of non-hallucinated examples in it. Given a user-specified α, let

$$k_\alpha = \lceil (n_{\text{cal}} + 1)(1 - \alpha) \rceil, \qquad \tau_\alpha = \begin{cases} f^{\text{cal}}_{(k_\alpha)}, & k_\alpha \le n_{\text{cal}}, \\ +\infty, & k_\alpha > n_{\text{cal}}, \end{cases} \tag{1}$$

where $f^{\text{cal}}_{(1)} \le \cdots \le f^{\text{cal}}_{(n_{\text{cal}})}$ denote the sorted scores of the non-hallucinated calibration examples. We then flag a test span as hallucinated when its score satisfies $f > \tau_\alpha$. Because the threshold is calibrated only on non-hallucinated examples, it targets a false-positive rate (FPR) of α on non-hallucinated spans (Bates et al., 2023; Angelopoulos et al., 2024); accordingly, we report the true-positive rate (TPR) at the split-conformal threshold for each nominal FPR budget α (Section 4).

4 EXPERIMENTAL SETUP

Having defined PALIMPSEST-RECOVERY, PALIMPSEST-YN, and the baselines, we now describe how we evaluate them: Section 4.1 introduces the datasets and metrics, and Appendix A describes the fine-tuning setup. Our experiments address four research questions:

- **Q1**: Can the bidirectional nature of MDLMs lead to an improvement over AR models?
- **Q2**: How well does PALIMPSEST detect hallucinations?
- **Q3**: How much does span proposal matter?
- **Q4**: How does fine-tuning affect hallucination detection performance?

4.1 DATASETS AND METRICS

We evaluate PALIMPSEST and the baselines on three datasets:

1. **HalluEntity** (Yeh et al., 2025): a test split of 3,881 human-annotated spans in which hallucinations may occur. This dataset provides no reference, so we set $R(X)$ to the empty string, and each method must rely on the parametric knowledge of its underlying model.

2. **BUMP** (Ma et al., 2023): a test split of 353 human-annotated spans, each accompanied by a reference. Although human-annotated spans are available, we also obtain spans automatically on this dataset to study how much span proposal matters (**Q3**).

3. **LLM-AggreFact** (Tang et al., 2024): a test split of 29,320 generations without human-annotated spans but with reference. We treat the whole generation as the span, that is, $E = X$.

On all three datasets, we report the area under the receiver operating characteristic curve (AUROC) and the true-positive rate (TPR) at three user-specified false-positive rate (FPR) budgets $\alpha \in \{0.05, 0.1, 0.2\}$, with thresholds set by the split-conformal procedure of Section 3.5.

5 Results and Discussion

This section addresses the four research questions posed in Section 4 in turn.

5.1 Q1: Can the bidirectional nature of MDLMs lead to an improvement over AR models?

To compare an MDLM with its AR counterpart under closely matched conditions, we use `Dream-v0-Base-7B` (Ye et al., 2025) and `Qwen2.5-7B` (Qwen et al., 2025). Dream shares Qwen's architecture and is initialized from its weights, suggesting similar parametric knowledge. The two models also perform comparably on standard benchmarks such as MMLU (Hendrycks et al., 2021) and GSM8K (Cobbe et al., 2021), with Dream slightly behind (Ye et al., 2025). This comparison therefore helps us assess the contribution of bidirectional denoising to hallucination detection while minimizing differences in model architecture, initialization, and general capabilities.

Figure 3 illustrates that the best PALIMPSEST score outperforms the best AR-based score on two of the three datasets: HalluEntity and BUMP. On LLM-AggreFact, however, R1 performs best, which might be due to the whole X is used as the span. This is less precise comparing to the span used in other datasets (see 5.3 for detailed discussion on the preciseness of span). In summary, these results demonstrate some benefits from the bidirectional nature of MDLMs for span-level hallucination detection.

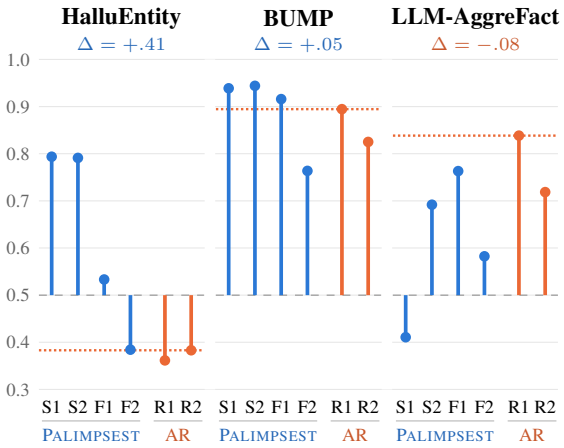


Figure 3: AUROC of PALIMPSEST vs. AR on a matched backbone (`Dream-v0-7B` vs. `Qwen2.5-7B`). The dotted line is the best AR score; Δ = best PALIMPSEST − best AR. Values in Table 3.

5.2 Q2: How well does PALIMPSEST detect hallucinations?

Having examined a matched pair, we broaden the comparison to AR models and other baselines, using two MDLM families: LLaDA (Nie et al., 2026) and Dream. For the AR scorers, we select recent open-weight models from four frontier developers: `Muse-Glimmer-30B` (Meta Superintelligence Labs, 2026), `Qwen3.8-27B` (Qwen Team, 2026), `gemma-4-31B-it` (Gemma Team, 2026), and `gpt-oss-20b` (OpenAI, 2025).

Table 1 presents the zero-shot results across all baselines. We omit NLI and checker results for HalluEntity because these models require a reference, which the dataset does not provide.

On HalluEntity, all four PALIMPSEST-RECOVERY configurations achieve AUROCs of 0.79–0.80, outperforming every baseline, including Focus (0.77) and the strongest AR scorer, R1 with `gemma-4-31B-it` (0.65). On BUMP, PALIMPSEST matches the best AR scorer (0.95 AUROC), while on LLM-AggreFact, it falls slightly behind (0.85 vs. 0.89), likely due to the fact that the span E in this dataset is not precise (see 5.3). Notably, PALIMPSEST achieves these results with 7B–8B models, compared with the 20B–31B AR models, suggesting ***greater parameter efficiency***. Figure 4 compares performance and the model size of each method and shows the efficiency of PALIMPSEST. The TPR results at fixed FPR budgets generally follow the same pattern, with one notable exception: on HalluEntity, R1 with `gemma-4-31B-it` achieves substantially higher TPRs at FPR 0.05 and 0.10 (0.36 and 0.44) than any PALIMPSEST configuration (at most 0.20 and 0.34). At FPR 0.20, however, PALIMPSEST achieves a higher TPR (0.59 vs. 0.45).

5.3 Q3: How much does span proposal matter?

Thus far, we have assumed that the candidate span E is given, either through human annotation (HalluEntity and BUMP) or as the entire generation (LLM-AggreFact). We now examine how the choice of candidate span identification method affects performance. We consider three conditions

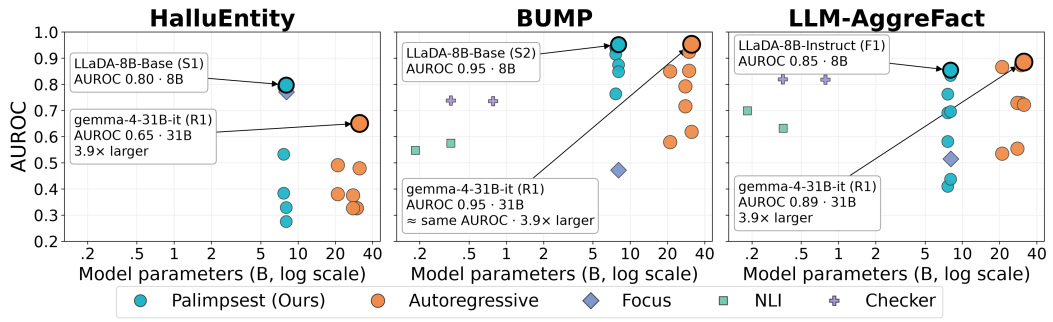


Figure 4: AUROC of each method versus its number of nominal parameters (log scale). The best PALIMPSEST configuration outperforms (HalluEntity), matches (BUMP), or trails (LLM-AggreFact) the best AR-based method, R1 with `gemma-4-31B-it`, which has 3.9 times as many parameters.

Table 1: Zero-shot results: AUROC (shaded blue above chance, red below) and TPR at the split-conformal threshold for nominal FPR budgets 0.05/0.10/0.20 (shaded gray by value). Each entry is followed by its standard deviation over a bootstrap that resamples source documents. Bold: best per column (ties at two decimals are all bold).

		HalluEntity				BUMP				LLM-AggreFact			
			TPR at FPR				TPR at FPR				TPR at FPR		
Score	Model	AUROC	.05	.10	.20	AUROC	.05	.10	.20	AUROC	.05	.10	.20
PALIMPSEST *(Ours)*													
S1	Dream-v0-Base-7B	.79±.01	.20±.02	.34±.03	**.59**±.03	.94±.01	.73±.03	.87±.02	.93±.02	.41±.01	.03±.00	.06±.00	.13±.01
	LLaDA-8B-Base	**.80**±.01	.17±.02	.34±.03	**.59**±.03	.94±.01	.73±.03	.84±.03	**.96**±.02	.44±.01	.03±.00	.06±.01	.13±.01
S2	Dream-v0-Base-7B	.79±.01	.15±.02	.31±.03	.57±.03	.94±.01	**.83**±.03	**.90**±.03	.95±.01	.69±.01	.15±.00	.25±.01	.41±.01
	LLaDA-8B-Base	**.80**±.01	.16±.02	.29±.03	**.59**±.03	**.95**±.01	.79±.03	**.90**±.02	**.96**±.02	.70±.01	.14±.01	.25±.01	.42±.01
F1	Dream-v0-Instruct-7B	.53±.03	.03±.01	.06±.02	.21±.04	.92±.02	.64±.03	.76±.04	.84±.03	.76±.01	.33±.01	.40±.01	.55±.01
	LLaDA-8B-Instruct	.28±.02	.00±.00	.01±.00	.02±.01	.88±.02	.53±.05	.63±.04	.71±.04	.85±.00	**.48**±.01	**.64**±.01	.77±.01
F2	Dream-v0-Instruct-7B	.38±.03	.01±.01	.03±.01	.13±.04	.76±.03	.37±.06	.50±.06	.52±.06	.58±.01	.05±.00	.07±.00	.18±.01
	LLaDA-8B-Instruct	.33±.02	.02±.01	.04±.01	.08±.02	.85±.02	.50±.04	.58±.04	.68±.05	.83±.00	.45±.01	.59±.01	.73±.01
Autoregressive													
R1	Muse-Glimmer-30B	.33±.02	.00±.00	.00±.00	.00±.00	.92±.01	.77±.03	.81±.03	.88±.02	.87±.00	.41±.01	.62±.01	.81±.01
	Qwen3.8-27B	.38±.03	.01±.01	.02±.01	.06±.02	.72±.03	.12±.03	.27±.04	.53±.06	.55±.01	.09±.00	.15±.01	.27±.01
	gemma-4-31B-it	.65±.05	**.36**±.08	**.44**±.08	.45±.08	**.95**±.01	.80±.02	.84±.02	.92±.02	**.89**±.00	.37±.01	.55±.01	**.86**±.01
	gpt-oss-20b	.38±.03	.04±.02	.07±.04	.12±.05	.85±.02	.62±.04	.67±.04	.77±.04	.87±.00	.47±.01	**.64**±.01	.77±.01
R2	Muse-Glimmer-30B	.33±.01	.01±.00	.02±.01	.04±.01	.85±.02	.33±.03	.52±.03	.73±.03	.73±.01	.21±.01	.32±.01	.48±.01
	Qwen3.8-27B	.33±.02	.03±.01	.07±.01	.11±.02	.79±.02	.47±.03	.53±.03	.62±.03	.73±.01	.20±.01	.32±.01	.48±.01
	gemma-4-31B-it	.48±.02	.04±.01	.09±.01	.16±.02	.62±.03	.18±.03	.33±.04	.51±.04	.72±.01	.23±.01	.33±.01	.49±.01
	gpt-oss-20b	.49±.03	.11±.02	.19±.02	.31±.03	.58±.02	.06±.02	.07±.02	.16±.03	.54±.01	.07±.00	.12±.01	.22±.01
Other baselines													
Focus	Llama-3.1-8B	.77±.01	.08±.01	.20±.02	.45±.03	.47±.02	.16±.03	.22±.04	.41±.04	.51±.01	.07±.00	.13±.01	.23±.01
NLI	DeBERTa-v3-base	–	–	–	–	.55±.02	.08±.05	.16±.06	.26±.07	.70±.01	.19±.01	.28±.01	.41±.01
	RoBERTa-L-MNLI	–	–	–	–	.57±.02	.02±.01	.16±.06	.19±.06	.63±.01	.15±.01	.23±.01	.37±.01
Checker	MiniCheck-FT5-L	–	–	–	–	.74±.02	.31±.03	.41±.03	.54±.03	.82±.00	.33±.01	.48±.01	.68±.01
	AlignScore-L	–	–	–	–	.74±.03	.28±.03	.45±.04	.63±.03	.82±.00	.32±.01	.47±.01	.66±.01

on BUMP: (1) *human-annotated spans*, where the scorer receives the gold span boundaries; (2) *sentence spans*, where `spaCy` (Honnibal et al., 2020) segments each generation into sentences, each treated as a candidate span; and (3) *proposed spans*, where `SmolLM3-3B` (Bakouch et al., 2025) proposes potentially hallucinated spans without access to the gold labels. The proposal prompt is provided in Appendix B.4.

Figure 5 compares performance under the three conditions. Replacing BUMP's human-annotated spans with the coarser `spaCy` sentence spans or `SmolLM3-3B` proposed span, which are less precise comparing to human annotated span, sharply reduces the AUROC of nearly all PALIMPSEST methods. The best AR scorer, which are about 3 times larger, are less affected to this change. This sensitivity to span boundaries motivates treating span proposal and span-level scoring as two distinct components of hallucination detection and we believe they are important future research directions.

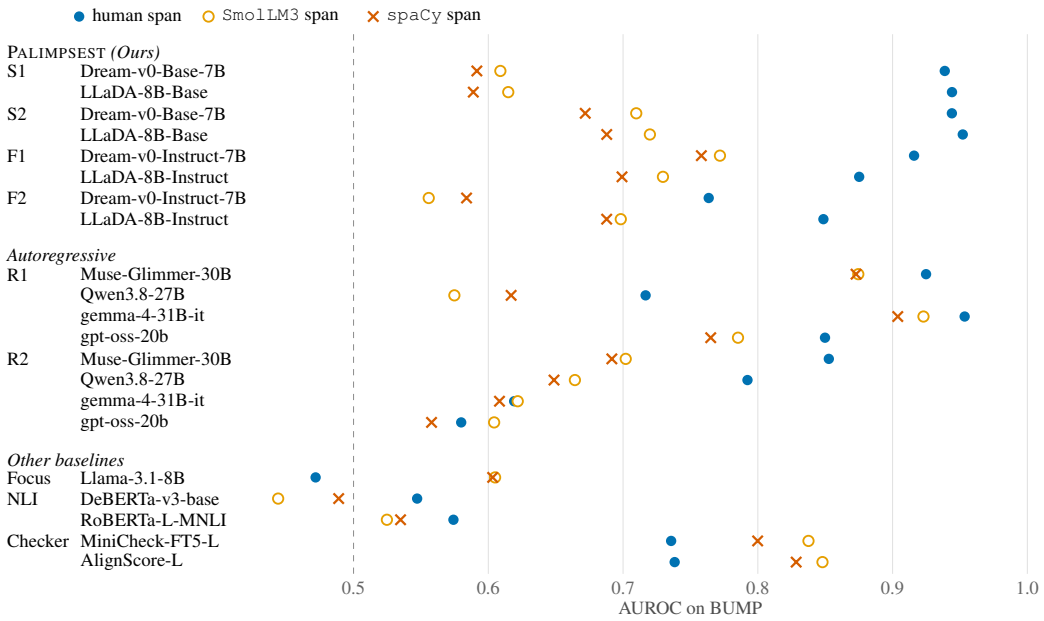


Figure 5: AUROC on BUMP as the candidate spans change from human-annotated spans to spans proposed by `SmolLM3-3B` and to `spaCy` sentence spans.

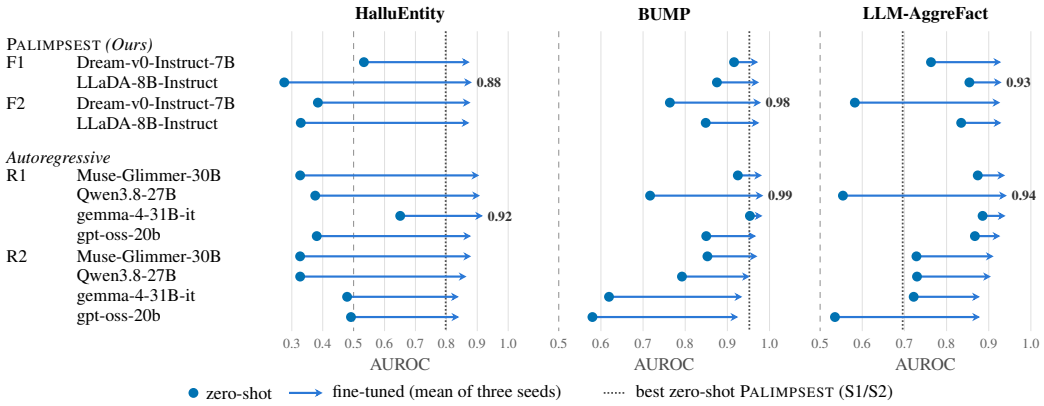


Figure 6: Effect of task-specific fine-tuning on AUROC. Dots mark zero-shot AUROC (Table 1); arrows point to fine-tuned AUROC (mean of three seeds; Table 5). The dotted line marks the best zero-shot PALIMPSEST score (S1/S2, no fine-tuning). The three panels use different x-axis ranges.

5.4 Q4: How does fine-tuning affect hallucination detection performance?

So far, we examine all of the PALIMPSEST and AR methods out of the box. Now, we examine how task-specific fine-tuning via LoRA (Hu et al., 2022) (see Appendix A for fine-tuning details) changes these comparisons. Figure 6 reports results on the same datasets as Table 1 after fine-tuning. Fine-tuning raises the AUROC of every PALIMPSEST-YN and AR configuration on all three datasets. Notebaly, after fine-tuning, PALIMPSEST-YN with 7B–8B checkpoints approaches the best AR scorers, which use 27B–31B checkpoints. The best PALIMPSEST-YN configurations also match or exceed R1 with `gpt-oss-20b`. This again ***shows the efficiency of* PALIMPSEST**. The smaller size of MDLMs not only enables faster inference, but also faster fine-tuning. Taken together, fine-tuning narrows the gap between PALIMPSEST and AR detectors that are roughly four times larger, making PALIMPSEST an attractive choice under limited compute budgets.

6 Related Work

Fine-grained hallucination detection and evaluation. Hallucination detection operates at the response (Manakul et al., 2023), claim (Min et al., 2023; Hu et al., 2024), entity (Yeh et al., 2025), token, or span level, each with a distinct prediction target (Huang et al., 2025). Early work labels hallucinated tokens in conditional generation or without a reference (Liu et al., 2022). The three benchmarks we use cover complementary settings: BUMP (Ma et al., 2023) supplies faithful–unfaithful minimal pairs, HalluEntity (Yeh et al., 2025) tests reference-free entity factuality, and LLM-AggreFact (Tang et al., 2024) aggregates grounded factuality datasets.

Detectors also differ in whether they consult external evidence. Uncertainty-based detectors rely on signals from a generating model, such as surprisal, entropy, and attention. Focus combines these signals with keyword selection and inverse-document-frequency weighting (Zhang et al., 2023). It is a useful reference-free comparator, but its uncertainty signals do not directly test whether external evidence supports a fluent statement. Reference-conditioned systems instead check text against evidence: NLI-based detectors apply entailment models to source–output pairs. MiniCheck uses a compact encoder–decoder model trained on synthetic claim–document pairs (Tang et al., 2024), AlignScore learns a unified alignment function (Zha et al., 2023). We use Focus, MiniCheck, and AlignScore as baselines (Section 3.4).

Diffusion language models as evaluators rather than generators. Like masked language models, masked diffusion objectives yield bidirectional mask predictors (Austin et al., 2021; Sahoo et al., 2024; Shi et al., 2024). LLaDA and Dream, the two MDLM families we evaluate, instantiate large language diffusion models (Nie et al., 2026; Ye et al., 2025). Existing hallucination studies on diffusion models primarily analyze the *evaluated generator*'s own decoding trace, temporal dynamics, hidden evidence, reveal-time uncertainty, or generation errors (Chang et al., 2026; Hemmat et al., 2026; Zhao et al., 2026; Weng et al., 2026; Guo & Tan, 2026). Our setting differs: an MDLM externally scores an *already generated* claim or span, which need not have been produced by that MDLM. Accordingly, we study the interface between candidate selection and factuality scoring rather than claiming that diffusion generation itself is more factual.

Calibration and factuality. Conformal prediction converts a real-valued score into a decision rule under an exchangeability assumption (Vovk et al., 2005; Angelopoulos & Bates, 2023; Angelopoulos et al., 2024). In factuality applications, conformal methods have been used to elicit factual contents from LLM's generation (Mohri & Hashimoto, 2024), and recent studies examine informativeness, robustness, and adaptive reweighting under shift (Chen et al., 2026; Hu et al., 2026). These approaches are complementary to ours, since PALIMPSEST supplies the underlying score while conformal procedures provide the statistical guarantees on top of it.

7 Conclusion, Limitations, and Future Work

We presented PALIMPSEST, a novel family of span-level hallucination detection methods built on MDLMs. PALIMPSEST-RECOVERY masks a candidate span and scores how well an MDLM recovers it, whereas PALIMPSEST-YN keeps the span visible and reads a verbalized support judgment from a masked answer position. Despite using roughly a quarter of the parameters of the strongest tested AR models (7B–8B vs. 27B–31B), MDLM scorers remain competitive. In zero-shot evaluation, they match the strongest AR model on BUMP (with human-annotated spans), outperform it on HalluEntity with PALIMPSEST-RECOVERY, and trail it by only 0.04 AUROC on LLM-AggreFact (0.85 vs. 0.89). After fine-tuning, the best PALIMPSEST-YN configuration comes within 0.04, 0.01, and 0.01 AUROC of the best AR scorer on HalluEntity, BUMP, and LLM-AggreFact, respectively. These results suggest that MDLMs offer a parameter-efficient alternative to scaling up AR detectors, reaching comparable detection quality at a fraction of the model size.

A limitation that PALIMPSEST has, as the sentence-span experiments show, is its sensitivity to span preciseness. So its practical performance depends on how candidate spans are obtained. This limitation motivates a main direction for future work: developing span proposers that produce precise candidate spans, treating span proposal and span-level scoring as distinct components.

AI USE STATEMENT

In this work, we used generative AI tools for implementing method (Focus by Zhang et al. (2023)). We have *not* used generative AI tools for,

- proposing or refining hypotheses,
- designing or providing feedback on research methodology or experiments,
- cleaning and reformatting dataset,
- supporting qualitative and thematic data analysis,
- and interpreting results.

Moreover, the following list of tasks are not applicable to this work:

- generating synthetic data sets,
- helping develop theoretical models or conceptual frameworks,
- formulating mathematical claims,
- providing critical ingredients for proving mathematical claims,
- assisting in the writing of proofs,
- and assisting with translation

Additionally, we used generative AI tools for

- creating or modifying scientific figures or images,
- suggesting experimental parameters,
- creating or editing software code,
- searching for information,
- editing a research paper to improve readability,
- identifying relevant literature,
- formatting references,
- and proposing a title or keywords for a research paper.

We have reviewed all AI-assisted work. The initial codebase was written by the authors and LLMs were used to extending and generalizing the initial codebase to different datasets, models, and computing environment. LLMs are also used to generate scripts for running the experiments on GPU clusters. LLM-generated code was verified and tested for correctness by the authors. The draft was written by the authors. LLMs were used while editing to obtain suggestions to correct grammar errors and improve the writing. The authors conducted literature review independently as well as used ChatGPT's websearch feature to look for additional relevant works. Author lists and content of related works in the references are checked manually by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

- The datasets used are public research benchmarks and were used as released.
- The models were used as released.
- Section 3 explains the methods we used in the paper.
- Appendix A describes how we used LoRA to do the fine-tuning.
- Appendix B lists the prompts we used.

REFERENCES

Anastasios N. Angelopoulos and Stephen Bates. Conformal prediction: A gentle introduction. *Foundations and Trends in Machine Learning*, 16(4):494–591, 2023. doi: 10.1561/2200000101. URL https://doi.org/10.1561/2200000101.

Anastasios N. Angelopoulos, Rina Foygel Barber, and Stephen Bates. Theoretical foundations of conformal prediction, 2024. URL https://arxiv.org/abs/2411.11824.

Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces. In *Advances in Neural Information Processing Systems*, volume 34, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/958c530554f78bcd8e97125b70e6973d-Abstract.html.

Elie Bakouch, Loubna Ben Allal, Anton Lozhkov, Nouamane Tazi, Lewis Tunstall, Carlos Miguel Patiño, Edward Beeching, Aymeric Roucher, Aksel Joonas Reedi, Quentin Gallouédec, Kashif Rasul, Nathan Habib, Clémentine Fourrier, Hynek Kydlicek, Guilherme Penedo, Hugo Larcher, Mathieu Morlon, Vaibhav Srivastav, Joshua Lochner, Xuan-Son Nguyen, Colin Raffel, Leandro von Werra, and Thomas Wolf. SmolLM3: smol, multilingual, long-context reasoner. https://huggingface.co/blog/smollm3, 2025.

Stephen Bates, Emmanuel Candès, Lihua Lei, Yaniv Romano, and Matteo Sesia. Testing for outliers with conformal p-values. *The Annals of Statistics*, 51(1):149–178, 2023. doi: 10.1214/22-AOS2244. URL https://doi.org/10.1214/22-AOS2244.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, 2020. URL https://papers.nips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html.

Shenxu Chang, Junchi Yu, Weixing Wang, Yongqiang Chen, Jialin Yu, Philip Torr, and Jindong Gu. TRACEDET: HALLUCINATION DETECTION FROM THE DECODING TRACE OF DIFFUSION LARGE LANGUAGE MODELS. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=4puxTouUSV.

Yi Chen, Daiwei Chen, Sukrut Madhav Chikodikar, Caitlyn Heqi Yin, and Ramya Korlakai Vinayak. Is conformal factuality for RAG-based LLMs robust? novel metrics and systematic insights, 2026. URL https://arxiv.org/abs/2603.16817.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53, 2024. URL https://jmlr.org/papers/v25/23-0870.html.

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

Matthew Dahl, Varun Magesh, Mirac Suzgun, and Daniel E. Ho. Large legal fictions: Profiling legal hallucinations in large language models. *Journal of Legal Analysis*, 16(1):64–93, 2024. doi: 10.1093/jla/laae003. URL https://doi.org/10.1093/jla/laae003.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423/.

Jinlan Fu, See-Kiong Ng, Zhengbao Jiang, and Pengfei Liu. GPTScore: Evaluate as you desire. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 6556–6576, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.365. URL https://aclanthology.org/2024.naacl-long.365/.

Gemma Team. Gemma 4 technical report, 2026. URL https://arxiv.org/abs/2607.02770.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, et al. The Llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Zhengnan Guo and Fei Tan. Lost in diffusion: Uncovering hallucination patterns and failure modes in diffusion large language models. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 17805–17818, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-395-1. doi: 10.18653/v1/2026.findings-acl.882. URL https://aclanthology.org/2026.findings-acl.882/.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. DeBERTaV3: Improving DeBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=sE7-XhLxHA.

Arshia Hemmat, Philip Torr, Yongqiang Chen, and Junchi Yu. TDGNet: Hallucination detection in diffusion language models via temporal dynamic graphs, 2026. URL https://arxiv.org/abs/2602.08048.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021. URL https://arxiv.org/abs/2009.03300.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/hash/4c5bcfec8584af0d967f1ab10179ca4b-Abstract.html.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. spaCy: Industrial-strength natural language processing in python, 2020.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Xiangkun Hu, Dongyu Ru, Lin Qiu, Qipeng Guo, Tianhang Zhang, Yang Xu, Yun Luo, Pengfei Liu, Yue Zhang, and Zheng Zhang. Knowledge-centric hallucination detection. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 6953–6975, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.395. URL https://aclanthology.org/2024.emnlp-main.395/.

Zirui Hu, Zheng Zhang, Yingjie Wang, Leszek Rutkowski, and Dacheng Tao. Cofact: Conformal factuality guarantees for language models under covariate shift. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=eiBp7rsc3K.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55, 2025. doi: 10.1145/3703155. URL https://doi.org/10.1145/3703155.

Inception Labs, Samar Khanna, Siddhant Kharbanda, Shufan Li, Harshit Varma, Eric Wang, Sawyer Birnbaum, Ziyang Luo, Yanis Miraoui, Akash Palrecha, Stefano Ermon, Aditya Grover, and Volodymyr Kuleshov. Mercury: Ultra-fast language models based on diffusion, 2025. URL https://arxiv.org/abs/2506.17298.

International Collegiate Programming Contest. OpenAI at the 2025 ICPC world finals. https://worldfinals.icpc.global/2025/openai.html, 2025. ICPC World Finals.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38, 2023. doi: 10.1145/3571730. URL https://doi.org/10.1145/3571730.

Nikhil Kandpal, Haikang Deng, Adam Roberts, Eric Wallace, and Colin Raffel. Large language models struggle to learn long-tail knowledge. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 15696–15707, Honolulu, Hawaii, USA, 23–29 Jul 2023. PMLR. URL https://proceedings.mlr.press/v202/kandpal23a.html.

Haoqiang Kang and Xiao-Yang Liu. Deficiency of large language models in finance: An empirical examination of hallucination, 2023. URL https://arxiv.org/abs/2311.15548.

Moritz Laurer, Wouter Van Atteveldt, Andreu Casas, and Kasper Welbers. Less annotating, more classifying: Addressing the data scarcity issue of supervised machine learning with deep transfer learning and BERT-NLI. *Political Analysis*, 32(1):84–100, 2024.

Jing Lei, Max G'Sell, Alessandro Rinaldo, Ryan J. Tibshirani, and Larry Wasserman. Distribution-free predictive inference for regression. *Journal of the American Statistical Association*, 113 (523):1094–1111, 2018. doi: 10.1080/01621459.2017.1307116. URL https://doi.org/10.1080/01621459.2017.1307116.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 9459–9474, 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html.

Weixin Liang, Yaohui Zhang, Mihai Codreanu, Jiayu Wang, Hancheng Cao, and James Zou. The widespread adoption of large language model-assisted writing across society. *Patterns*, 6(12), 2025.

Hanzhao Lin and Heng-Tze Cheng. Gemini achieves gold-medal level at the International Collegiate Programming Contest world finals. https://deepmind.google/blog/gemini-achieves-gold-medal-level-at-the-international-collegiate-programming-contest-world-finals/, September 2025. Google DeepMind Blog.

Tianyu Liu, Yizhe Zhang, Chris Brockett, Yi Mao, Zhifang Sui, Weizhu Chen, and Bill Dolan. A token-level reference-free hallucination detection benchmark for free-form text generation. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 6723–6737, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.464. URL https://aclanthology.org/2022.acl-long.464/.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. G-eval: NLG evaluation using GPT-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 2511–2522, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.153. URL https://aclanthology.org/2023.emnlp-main.153/.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Aaron Lou, Chenlin Meng, and Stefano Ermon. Discrete diffusion modeling by estimating the ratios of the data distribution. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, Vienna, Austria, 2024. PMLR. URL https://proceedings.mlr.press/v235/lou24a.html.

Liang Ma, Shuyang Cao, Robert L Logan IV, Di Lu, Shihao Ran, Ke Zhang, Joel Tetreault, and Alejandro Jaimes. 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)*, pp. 12788–12812, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.716. URL https://aclanthology.org/2023.acl-long.716/.

Varun Magesh, Faiz Surani, Matthew Dahl, Mirac Suzgun, Christopher D. Manning, and Daniel E. Ho. Hallucination-free? Assessing the reliability of leading AI legal research tools. *Journal of Empirical Legal Studies*, 22(2):216–242, 2025. doi: 10.1111/jels.12413. URL https://doi.org/10.1111/jels.12413.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9802–9822, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.546. URL https://aclanthology.org/2023.acl-long.546/.

Potsawee Manakul, Adian Liusie, and Mark Gales. SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 9004–9017, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.557. URL https://aclanthology.org/2023.emnlp-main.557/.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 1906–1919, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.173. URL https://aclanthology.org/2020.acl-main.173/.

Meta Superintelligence Labs. Introducing Muse Glimmer: An open agentic model that runs on your device. https://research.meta.ai/blog/introducing-muse-glimmer-open-agentic-model, August 2026. Blog post, 10 August 2026. Model weights: https://huggingface.co/meta-models/Muse-Glimmer-30B.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 12076–12100, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.741. URL https://aclanthology.org/2023.emnlp-main.741/.

Abhika Mishra, Akari Asai, Vidhisha Balachandran, Yizhong Wang, Graham Neubig, Yulia Tsvetkov, and Hannaneh Hajishirzi. Fine-grained hallucination detection and editing for language models. In *First Conference on Language Modeling*, 2024. URL https://arxiv.org/abs/2401.06855.

Christopher Mohri and Tatsunori Hashimoto. Language models with conformal factuality guarantees. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 36029–36047. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/mohri24a.html.

Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Large language diffusion models. *Advances in Neural Information Processing Systems*, 38:50608–50646, 2026.

Cheng Niu, Yuanhao Wu, Juno Zhu, Siliang Xu, KaShun Shum, Randy Zhong, Juntong Song, and Tong Zhang. RAGTruth: A hallucination corpus for developing trustworthy retrieval-augmented language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10862–10878, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.585. URL https://aclanthology.org/2024.acl-long.585/.

OpenAI. gpt-oss-120b & gpt-oss-20b model card, 2025. URL https://arxiv.org/abs/2508.10925.

OpenAI. Finite time blowup for Navier–Stokes. https://cdn.openai.com/pdf/32d9f210-8b73-45e0-91bc-82a30aef8a9a/navier-stokes.pdf, September 2026. Released 8 September 2026. Lean 4 formalization: https://github.com/openai/NavierStokesAndEuler.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. Med-HALT: Medical domain hallucination test for large language models. In *Proceedings of the 27th Conference on Computational Natural Language Learning (CoNLL)*, pp. 314–334, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.conll-1.21. URL https://aclanthology.org/2023.conll-1.21/.

Harris Papadopoulos, Kostas Proedrou, Volodya Vovk, and Alex Gammerman. Inductive confidence machines for regression. In Tapio Elomaa, Heikki Mannila, and Hannu Toivonen (eds.), *Machine Learning: ECML 2002*, Lecture Notes in Computer Science, pp. 345–356, Berlin, Heidelberg, 2002. Springer. doi: 10.1007/3-540-36755-1_29. URL https://doi.org/10.1007/3-540-36755-1_29.

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. Qwen2.5 technical report, 2025. URL https://arxiv.org/abs/2412.15115.

Qwen Team. Qwen3.8-Max: A new bar for coding and cowork. https://qwen.ai/blog?id=qwen3.8, August 2026. Official citation of the Qwen/Qwen3.8-27B model card, https://huggingface.co/Qwen/Qwen3.8-27B.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. https://d4mucfpksywv.cloudfront.net/better-language-models/language_models_are_unsupervised_multitask_learners.pdf, 2019. OpenAI.

Subham Sekhar Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 130136–130184. Curran Associates, Inc., 2024. doi: 10.52202/079017-

4135. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/eb0b13cc515724ab8015bc978fdde0ad-Paper-Conference.pdf.

Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis K. Titsias. Simplified and generalized masked diffusion for discrete data. In *Advances in Neural Information Processing Systems*, volume 37, pp. 103131–103167, 2024. doi: 10.52202/079017-3277. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/bad233b9849f019aead5e5cc60cef70f-Abstract-Conference.html.

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. Retrieval augmentation reduces hallucination in conversation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pp. 3784–3803, Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-emnlp.320. URL https://aclanthology.org/2021.findings-emnlp.320/.

Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, Lille, France, 2015. URL https://proceedings.mlr.press/v37/sohl-dickstein15.html.

Yuxuan Song, Zheng Zhang, Cheng Luo, Pengyang Gao, Fan Xia, Hao Luo, Zheng Li, Yuehang Yang, Hongli Yu, Xingwei Qu, Yuwei Fu, Jing Su, Ge Zhang, Wenhao Huang, Mingxuan Wang, Lin Yan, Xiaoying Jia, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Yonghui Wu, and Hao Zhou. Seed diffusion: A large-scale diffusion language model with high-speed inference, 2025. URL https://arxiv.org/abs/2508.02193.

Peiqi Sui, Eamon Duede, Sophie Wu, and Richard Jean So. Confabulation: The surprising value of large language model hallucinations. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14274–14284, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.770. URL https://aclanthology.org/2024.acl-long.770/.

Liyan Tang, Philippe Laban, and Greg Durrett. MiniCheck: Efficient fact-checking of LLMs on grounding documents. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 8818–8847, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.499. URL https://aclanthology.org/2024.emnlp-main.499/.

Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. *Algorithmic Learning in a Random World*. Springer, New York, 2005. ISBN 0387001522. doi: 10.1007/b106715. URL https://doi.org/10.1007/b106715.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. Emergent abilities of large language models. *Transactions on Machine Learning Research*, 2022. ISSN 2835-8856. URL https://openreview.net/forum?id=yzkSU5zdwD. Survey Certification.

Pengcheng Weng, Yanyu Qian, Yue Tan, and Yixin Liu. TRE: Training-free hallucination detection for diffusion language models, 2026. URL https://arxiv.org/abs/2607.22661.

Adina Williams, Nikita Nangia, and Samuel R. Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 1112–1122, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1101. URL https://aclanthology.org/N18-1101/.

Chengyue Wu, Hao Zhang, Shuchen Xue, Zhijian Liu, Shizhe Diao, Ligeng Zhu, Ping Luo, Song Han, and Enze Xie. Fast-dLLM: Training-free acceleration of diffusion LLM by enabling KV cache and parallel decoding, 2025. URL https://arxiv.org/abs/2505.22618.

Jiacheng Ye, Zhihui Xie, Lin Zheng, Jiahui Gao, Zirui Wu, Xin Jiang, Zhenguo Li, and Lingpeng Kong. Dream 7b: Diffusion large language models, 2025. URL https://arxiv.org/abs/2508.15487.

Min-Hsuan Yeh, Max Kamachee, Seongheon Park, and Yixuan Li. HalluEntity: Benchmarking and understanding entity-level hallucination detection. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL https://openreview.net/forum?id=494k7e9R5D.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. AlignScore: Evaluating factual consistency with a unified alignment function. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 11328–11348, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.634. URL https://aclanthology.org/2023.acl-long.634/.

Tianhang Zhang, Lin Qiu, Qipeng Guo, Cheng Deng, Yue Zhang, Zheng Zhang, Chenghu Zhou, Xinbing Wang, and Luoyi Fu. Enhancing uncertainty-based hallucination detection with stronger focus. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 915–932, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.58. URL https://aclanthology.org/2023.emnlp-main.58/.

Guoshenghui Zhao, Tan Yu, and Weijie Zhao. HIVE: Hidden-evidence verification for hallucination detection in diffusion large language models, 2026. URL https://arxiv.org/abs/2604.26139.

A FINE-TUNING

We fine-tune the models used for F1, F2, R1, and R2. We use low-rank adaptation (LoRA) (Hu et al., 2022) with rank 16, scaling factor 32, and dropout 0.05, and optimize with AdamW (Loshchilov & Hutter, 2019) at a learning rate of 10^{-4} for 300 steps. We use a batch size of 1 with 16 gradient-accumulation steps, `bfloat16` precision, a 2,048-token limit, class-balanced sampling, and three random seeds. F1 and F2 are trained with cross-entropy over the normalized yes/no distribution, and R1 with cross-entropy over the five rubric labels. R2 instead uses a pairwise hinge loss between supported and hallucinated spans from the same reference group.

B PROMPT TEMPLATES

This appendix lists the prompt templates for the two PALIMPSEST-YN, F1 and F2 (Section 3), as well as the two AR baselines, R1 and R2 (Section 3.4), and for the `SmolLM3-3B` span proposer used in Q3 (Section 5). Colored fields in braces are filled in for each input: $\{R(X)\}$ is the reference, $\{X\}$ the generation, and $\{E\}$ the candidate span. The fine-tuned R1 and R2 scorers (Appendix A) are trained and evaluated with the same templates as their zero-shot counterparts. When a prompt exceeds the 2,048-token budget, we truncate only the reference, keeping its beginning, so that the generation, the span, and the instructions are never shortened.

B.1 F1 AND F2 PROMPT TEMPLATE

Figure 7 shows two examples of filling the prompt templates for F1 and F2. The only difference between F1 and F2 is the location for the answer `<mask>`.

F1

Reference:
Paris is the capital of France. Its population was 2.1 million in 2023.

Statement: Paris, the capital of France, had [[3.5 million]] residents in 2023.

Question: is the text marked with [[ ]] supported by the reference? Answer yes or no.

Answer: ⟨`mask`⟩

F2

Reference:
Paris is the capital of France. Its population was 2.1 million in 2023.

Answer: ⟨`mask`⟩

Question: is the text marked with [[ ]] supported by the reference? Answer yes or no.

Statement: Paris, the capital of France, had [[3.5 million]] residents in 2023.

Figure 7: The two PALIMPSEST-YN prompt formats, with the same content in different orders. In format F1, the ⟨`mask`⟩ token comes last; in format F2, it appears before the question and the statement.

B.2 R1 PROMPT TEMPLATE

The R1 template is shown below. We place it in the model's chat template as a single user turn, followed by the generation prompt. Rather than sampling a rating, we read the probability of each label $d \in \{1, \dots, 5\}$ at the answer position, summing over the tokenizations of the label with and without a leading space, and renormalize over the five labels to obtain $\tilde{p}(d)$. Mapping the labels to $v(1), \dots, v(5) = -1, -\frac{1}{2}, 0, \frac{1}{2}, 1$, we use the negated expected rating as the score,

$$f = -\sum_{d=1}^{5} \tilde{p}(d)\, v(d) \in [-1, 1],$$

so that, as for every other method, higher values indicate a greater likelihood of hallucination.

R1, with reference

```
You are checking whether a span of text is supported by a reference
document.

REFERENCE:
{R(X)}

TEXT:
{X}

SPAN: {E}

Rate how well the REFERENCE supports the SPAN:
5 = fully supported by the reference
4 = mostly supported
3 = the reference neither supports nor contradicts it
2 = mostly unsupported by the reference
1 = contradicted by the reference

Answer with a single digit from 1 to 5.
Answer:
```

HalluEntity provides no reference, and support is undefined without one, so on this dataset R1 uses the variant below, which asks instead whether the span is factually correct. On LLM-AggreFact, where the candidate span is the whole generation, $\{X\}$ and $\{E\}$ are identical.

R1, without reference (HalluEntity)

```
You are checking whether a span of text states something true.

TEXT:
{X}

SPAN: {E}

Rate how likely the SPAN is to be factually correct:
5 = certainly correct
4 = probably correct
3 = unclear
2 = probably incorrect
1 = certainly incorrect

Answer with a single digit from 1 to 5.
Answer:
```

B.3 R2 Prompt Template

The R2 template is shown below. The field $\{X$ with E replaced by ___$\}$ is the generation in which the span is replaced by a single blank of three underscores, whatever the length of E. We append the tokens of E, preceded by a space, to the prompt and score them by teacher forcing in a single forward pass:

$$f = -\frac{1}{|E|}\sum_{i=1}^{|E|}\log p_{\mathrm{AR}}\big(E_i \mid \text{prompt}, E_{<i}\big),$$

which, like S1, is normalized by the span length. For the running example of Section 3.2, the `TEXT` block reads "`The observatory ___, three years after the survey began.`" and the scored continuation is "`opened in 1990`". On HalluEntity, the `REFERENCE` block is omitted.

R2

```
REFERENCE:
{R(X)}

TEXT:
{X with E replaced by ___}

The missing text is:
```

B.4 Span Proposer Prompt

In the proposed-span condition of Q3, `SmolLM3-3B` receives the reference and the generation, never the annotations, and returns a JSON list of strings copied from the generation. The prompt consists of two few-shot exchanges followed by the query, all formatted with the model's chat template with its reasoning mode disabled. The few-shot examples are synthetic and share no content with the evaluation data. They demonstrate three behaviors: quoting from the generation rather than from the reference, returning the shortest span that carries the error, and returning an empty list when everything is supported.

Span proposer: few-shot exchanges

User:

```
REFERENCE (evidence only -- never quote from this):
The Kestrel bridge opened in 1954. It spans 320 metres and carries two
lanes of traffic.

RESPONSE (quote only from this):
The Kestrel bridge, opened in 1961, spans 320 metres and carries four
lanes.

JSON:
```

Assistant:

```
["1961", "four lanes"]
```

User:

```
REFERENCE (evidence only -- never quote from this):
Dr Awan joined the institute in 2011 and leads the materials group.

RESPONSE (quote only from this):
Dr Awan leads the materials group.

JSON:
```

Assistant:

```
[]
```

Span proposer: query (final user turn)

User:

```
You are checking a RESPONSE against a REFERENCE for unsupported content.

REFERENCE (evidence only -- never quote from this):
{R(X)}

RESPONSE (quote only from this):
{X}

List the spans of the RESPONSE above that are NOT supported by the
REFERENCE -- invented facts, wrong numbers or names, or claims the
reference does not establish.

Rules:
```

```
- Every span you return MUST be copied from the RESPONSE, not from the
REFERENCE.
- Copy it EXACTLY as it appears in the RESPONSE, character for character,
including spacing.
- Quote the shortest span that carries the problem, not the whole sentence.
- If everything is supported, return an empty list.

The RESPONSE, again, is:
{X}

Return only a JSON list of strings copied from that RESPONSE.

JSON:
```

We decode greedily, with at most 192 new tokens, and parse the first JSON list of strings in the output. Each returned string is mapped back to a character range of the generation by exact substring matching, with a fallback that ignores whitespace, because BUMP summaries are tokenized with spaces before punctuation that the model often drops when quoting. Strings that cannot be located are discarded rather than matched approximately, as are strings that overlap an earlier proposal. A proposed span is labeled as hallucinated if it overlaps a human-annotated hallucinated span by at least one character.

On the 353 BUMP test summaries, the proposer returned 911 spans, averaging 13.0 words; every returned string was located in its summary, and only one summary yielded no parsable output. Only 150 of the 911 spans (16.5%) are labeled as hallucinated, compared with 180 of the 353 human-annotated spans, so most proposed spans are supported text that the proposer nonetheless found suspicious.

C Agreement Between S1 and S2

To examine how much the two scores differ, we selected 100 examples from each of the datasets we used and computed the Spearman rank correlation between S1 and S2 as a function of the span length $|E|$ (Table 2). The two scores agree almost perfectly on short spans, and their correlation decreases monotonically as $|E|$ grows, from 1.00 for single-token spans to 0.33 for spans of 33 or more tokens, suggesting that the choice between S1 and S2 matters mainly for longer spans.

Table 2: Spearman rank correlation between the PALIMPSEST-RECOVERY scores S1 and S2 as a function of the masked span length $|E|$.

$\|E\|$ (tokens)	Spearman ρ
1	1.00
2	0.99
3–4	0.96
5–8	0.63
9–16	0.56
17–32	0.47
33+	0.33

D Numerical Results

Tables 3, 4, and 5 list the AUROCs plotted in Figures 3, 5, and 6, respectively, together with the corresponding TPRs, in the format of Table 1.

Table 3: Zero-shot AUROC (plotted in Figure 3) and TPR at the split-conformal threshold for nominal FPR budgets 0.05/0.10/0.20 for the matched backbone pair, `Dream-v0-7B` (PALIMPSEST) and `Qwen2.5-7B` (AR). The PALIMPSEST values repeat Table 1. Shading and standard deviations as in Table 1; bold marks the best value per column within this table.

		HalluEntity				BUMP				LLM-AggreFact			
			TPR at FPR				TPR at FPR				TPR at FPR		
Score	Model	AUROC	.05	.10	.20	AUROC	.05	.10	.20	AUROC	.05	.10	.20
PALIMPSEST *(Ours)*													
S1	Dream-v0-Base-7B	**.79**±.01	**.20**±.02	**.34**±.03	**.59**±.03	**.94**±.01	.73±.03	.87±.02	.93±.02	.41±.01	.03±.00	.06±.00	.13±.01
S2	Dream-v0-Base-7B	**.79**±.01	.15±.02	.31±.03	.57±.03	**.94**±.01	**.83**±.03	**.90**±.03	**.95**±.01	.69±.01	.15±.00	.25±.01	.41±.01
F1	Dream-v0-Instruct-7B	.53±.03	.03±.01	.06±.02	.21±.04	.92±.02	.64±.03	.76±.04	.84±.03	.76±.01	.33±.01	.40±.01	.55±.01
F2	Dream-v0-Instruct-7B	.38±.03	.01±.01	.03±.01	.13±.04	.76±.03	.37±.06	.50±.06	.52±.06	.58±.01	.05±.00	.07±.00	.18±.01
Autoregressive													
R1	Qwen2.5-7B	.36±.03	.00±.00	.00±.00	.02±.01	.89±.01	.66±.04	.71±.03	.77±.03	**.84**±.00	**.45**±.01	**.61**±.01	**.73**±.01
R2	Qwen2.5-7B	.38±.02	.03±.01	.04±.01	.08±.01	.82±.03	.31±.03	.49±.04	.72±.04	.72±.01	.18±.01	.30±.01	.45±.01

Table 4: Zero-shot AUROC (plotted in Figure 5) and TPR at the split-conformal threshold for nominal FPR budgets 0.05/0.10/0.20 on BUMP with human-annotated spans, spans proposed by `SmolLM3-3B`, and `spaCy` sentence spans. The human-span columns repeat the BUMP columns of Table 1. Shading, standard deviations, and bold as in Table 1.

		Human span				`SmolLM3` span				`spaCy` span			
			TPR at FPR				TPR at FPR				TPR at FPR		
Score	Model	AUROC	.05	.10	.20	AUROC	.05	.10	.20	AUROC	.05	.10	.20
PALIMPSEST *(Ours)*													
S1	Dream-v0-Base-7B	.94±.01	.73±.03	.87±.02	.93±.02	.61±.03	.22±.05	.30±.05	.53±.06	.59±.03	.20±.04	.30±.05	.42±.07
	LLaDA-8B-Base	.94±.01	.73±.03	.84±.03	**.96**±.02	.61±.03	.26±.05	.36±.05	.43±.06	.59±.02	.14±.03	.29±.05	.42±.06
S2	Dream-v0-Base-7B	.94±.01	**.83**±.03	**.90**±.03	.95±.01	.71±.03	.35±.06	.39±.07	.65±.07	.67±.03	.18±.04	.38±.06	.59±.07
	LLaDA-8B-Base	**.95**±.01	.79±.03	**.90**±.02	**.96**±.02	.72±.03	.39±.07	.44±.07	.59±.07	.69±.03	.20±.04	.37±.06	.56±.07
F1	Dream-v0-Instruct-7B	.92±.02	.64±.03	.76±.04	.84±.03	.77±.03	.34±.05	.52±.05	.64±.05	.76±.02	.34±.04	.51±.04	.67±.04
	LLaDA-8B-Instruct	.88±.02	.53±.05	.63±.04	.71±.04	.73±.03	.32±.06	.41±.06	.54±.06	.70±.02	.33±.05	.40±.05	.51±.05
F2	Dream-v0-Instruct-7B	.76±.03	.37±.06	.50±.06	.52±.06	.56±.04	.15±.05	.24±.06	.31±.07	.58±.02	.15±.05	.26±.06	.33±.07
	LLaDA-8B-Instruct	.85±.02	.50±.04	.58±.04	.68±.05	.70±.03	.25±.05	.39±.06	.53±.06	.69±.02	.24±.04	.34±.05	.48±.05
Autoregressive													
R1	Muse-Glimmer-30B	.92±.01	.77±.03	.81±.03	.88±.02	.87±.02	**.69**±.04	.75±.04	.83±.03	.87±.02	.59±.04	.69±.03	.79±.03
	Qwen3.8-27B	.72±.03	.12±.03	.27±.04	.53±.06	.57±.05	.05±.02	.08±.03	.19±.06	.62±.03	.07±.02	.14±.04	.31±.06
	gemma-4-31B-it	**.95**±.01	.80±.02	.84±.02	.92±.02	**.92**±.01	.63±.04	**.81**±.03	**.86**±.03	**.90**±.01	**.61**±.04	**.76**±.03	**.82**±.03
	gpt-oss-20b	.85±.02	.62±.04	.67±.04	.77±.04	.79±.03	.51±.06	.59±.05	.69±.05	.77±.02	.40±.04	.50±.04	.64±.04
R2	Muse-Glimmer-30B	.85±.02	.33±.03	.52±.03	.73±.03	.70±.04	.35±.06	.49±.07	.56±.06	.69±.03	.18±.04	.41±.06	.59±.06
	Qwen3.8-27B	.79±.02	.47±.03	.53±.03	.62±.03	.66±.03	.19±.05	.29±.06	.53±.08	.65±.04	.20±.05	.32±.06	.51±.07
	gemma-4-31B-it	.62±.03	.18±.03	.33±.04	.51±.04	.62±.04	.05±.03	.09±.04	.31±.06	.61±.03	.06±.03	.13±.04	.24±.06
	gpt-oss-20b	.58±.02	.06±.02	.07±.02	.16±.03	.60±.03	.05±.02	.21±.05	.28±.07	.56±.02	.07±.03	.20±.04	.28±.06
Other baselines													
Focus	Llama-3.1-8B	.47±.02	.16±.03	.22±.04	.41±.04	.61±.04	.10±.04	.21±.05	.45±.07	.60±.03	.09±.04	.20±.05	.42±.07
NLI	DeBERTa-v3-base	.55±.02	.08±.05	.16±.06	.26±.07	.44±.03	.20±.07	.22±.08	.25±.08	.49±.02	.14±.06	.16±.06	.31±.07
	RoBERTa-L-MNLI	.57±.02	.02±.01	.16±.06	.19±.06	.52±.04	.01±.01	.13±.06	.21±.07	.53±.02	.02±.01	.14±.06	.20±.06
Checker	MiniCheck-FT5-L	.74±.02	.31±.03	.41±.03	.54±.03	.84±.03	.19±.03	.47±.04	.69±.04	.80±.04	.27±.03	.52±.03	.69±.03
	AlignScore-L	.74±.03	.28±.03	.45±.04	.63±.03	.85±.03	.39±.06	.53±.05	**.86**±.04	.83±.03	.46±.05	.51±.05	.75±.04

Table 5: AUROC (plotted in Figure 6) and TPR at the split-conformal threshold for nominal FPR budgets 0.05/0.10/0.20 after fine-tuning, averaged over three seeds. Each entry is followed by its standard deviation across the three seeds, not the bootstrap standard deviation of Table 1. The zero-shot counterparts are in Table 1. Shading and bold as in Table 1.

		HalluEntity				BUMP				LLM-AggreFact			
			TPR at FPR				TPR at FPR				TPR at FPR		
Score	Model	AUROC	.05	.10	.20	AUROC	.05	.10	.20	AUROC	.05	.10	.20
PALIMPSEST *(Ours)*													
F1	Dream-v0-Instruct-7B	.88±.00	.41±.02	.60±.02	.80±.01	.97±.00	.91±.01	.92±.01	.96±.01	.93±.00	.66±.03	.77±.01	.88±.01
	LLaDA-8B-Instruct	.88±.00	.42±.03	.61±.02	.83±.03	.98±.00	.84±.05	.91±.01	.96±.01	.93±.00	.66±.02	.78±.01	.88±.00
F2	Dream-v0-Instruct-7B	.88±.01	.41±.02	.59±.02	.82±.02	.98±.00	.91±.01	.94±.01	.96±.01	.93±.00	.65±.02	.77±.02	.88±.01
	LLaDA-8B-Instruct	.87±.01	.40±.03	.58±.02	.81±.03	.98±.00	.90±.01	.93±.02	.95±.00	.93±.00	.66±.01	.77±.00	.88±.00
Autoregressive													
R1	Muse-Glimmer-30B	.91±.01	.57±.01	.70±.02	.88±.01	.98±.00	**.94**±.01	**.96**±.01	**.97**±.01	**.94**±.00	**.71**±.02	**.81**±.00	.90±.01
	Qwen3.8-27B	.91±.00	.59±.02	.73±.00	**.89**±.01	**.99**±.00	**.94**±.01	.95±.00	.96±.00	**.94**±.00	**.71**±.00	**.81**±.01	**.91**±.00
	gemma-4-31B-it	**.92**±.01	**.61**±.06	**.76**±.05	**.89**±.01	.98±.00	.93±.01	**.96**±.01	**.97**±.00	**.94**±.00	.69±.03	.80±.01	**.91**±.00
	gpt-oss-20b	.88±.01	.42±.04	.60±.05	.81±.03	.97±.00	.69±.12	.84±.02	.94±.01	.93±.00	.64±.02	.76±.01	.87±.01
R2	Muse-Glimmer-30B	.88±.00	.41±.02	.64±.02	.84±.01	.97±.00	.88±.04	.94±.01	**.97**±.01	.91±.01	.55±.02	.70±.03	.86±.02
	Qwen3.8-27B	.87±.01	.36±.03	.57±.03	.79±.03	.95±.00	.77±.05	.85±.04	.95±.01	.91±.01	.52±.02	.67±.02	.85±.01
	gemma-4-31B-it	.84±.00	.27±.02	.44±.03	.74±.01	.94±.02	.71±.09	.80±.02	.91±.02	.88±.00	.46±.02	.60±.01	.78±.02
	gpt-oss-20b	.84±.01	.28±.03	.47±.03	.75±.04	.93±.00	.69±.03	.79±.05	.91±.01	.88±.00	.48±.02	.62±.01	.80±.01