FaithLens: Detecting and Explaining Faithfulness Hallucination

Anonymous ACL submission

Abstract

Recognizing whether outputs from large language models (LLMs) contain faithfulness hallucination is crucial for real-world applications, e.g., retrieval-augmented generation and summarization. In this paper, we introduce **FaithLens**, a cost-efficient and effective faithfulness hallucination detection model that can jointly provide binary predictions and corresponding explanations to improve trustworthiness. To achieve this, we first synthesize training data with explanations via advanced LLMs and apply a well-defined data filtering strategy to ensure label correctness, explanation quality, and data diversity. Subsequently, we fine-tune the model on these well-curated training data as a cold start and further optimize it with rule-based reinforcement learning, using rewards for both prediction correctness and explanation quality. Results on 12 diverse tasks show that the 8B-parameter FaithLens outperforms advanced models such as GPT-4.1 and o3. Also, FaithLens can produce high-quality explanations, delivering a distinctive balance of trustworthiness, efficiency, and effectiveness.

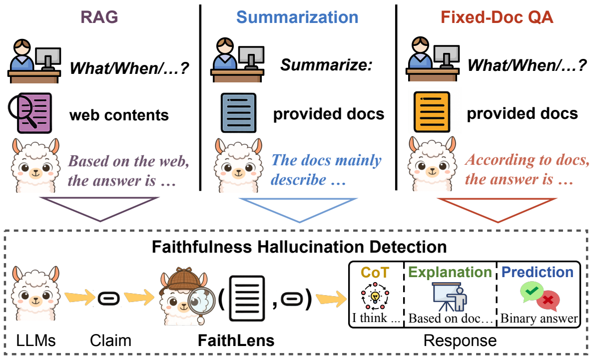

Figure 1: **The illustration of our FaithLens**. Given a document *doc* and a claim c, FaithLens can jointly determine whether the claim is faithful or hallucinated and provide the corresponding explanations for its decision, applicable across various tasks.

1 Introduction

Recent progress in large language models (LLMs) has revolutionized text generation (OpenAI, 2025). In practice, LLMs are widely used to generate coherent responses based on the provided contextual information, e.g., retrieval-augmented generation (RAG) (Wang et al., 2025). However, LLMs are prone to generating hallucinated claims that are inconsistent or irrelevant to the given context, i.e., faithfulness hallucinations (Bi et al., 2025; Si et al., 2025c). Therefore, detecting such hallucinations is critical for providing responsible LLM services.

To identify faithfulness hallucinations in LLM-generated outputs, recent works utilize the strong generalization abilities of LLMs and formulate it as a binary classification task (Wang et al., 2024). The first line of research leverages designed prompts to query advanced LLMs like GPT-4o (OpenAI, 2023) to check if generated outputs contain hallucinated claims (Liu et al., 2023c; Lei et al., 2023; Dhuliawala et al., 2024; Muhammed et al., 2025), e.g., SelfCheckGPT (Manakul et al., 2023). However, these methods are inefficient for real-world deployment because they rely on large and advanced models to achieve reliable detection performance.

Thus, many studies have focused on developing cost-efficient and specialized classifiers to detect hallucinations (Zha et al., 2023; Seo et al., 2025). For example, MiniCheck (Tang et al., 2024a) uses synthetic data generation techniques to train a 7B-parameter model, achieving performance comparable to GPT-4o. However, developing a detection model for real-world users still faces three key challenges. Specifically, **(1) Lack of Explainability**: Current methods typically treat faithfulness hallucination detection as a binary classification task, acting as a black box that only returns the final prediction without corresponding explanation (Tang et al., 2024a). This makes it difficult for users to localize errors and understand why tested claims are hallucinated, which limits the trustworthiness of detection models. **(2) Inconsistent Generalization across Tasks**: Previous methods are primarily designed for detecting task-specific hallucination (George and Stuhlmueller, 2023), e.g., summarization (Wan et al., 2024), and then fail to transfer across different tasks effectively. Even the models designed for general-purpose scenarios (Tang et al., 2024a; Lei et al., 2025; Seo et al., 2025) still perform unevenly on different tasks because each task may have unique hallucination patterns. For example, summarization hallucinations typically manifest as subtly distorted content from the context (Li and Yu, 2025), whereas RAG hallucinations often ignore the retrieved context and involve conflicting claims (Xu et al., 2024). **(3) Lack of High-Quality Data**: Annotating training data for hallucination detection is costly and often results in low inter-annotator agreement (Seo et al., 2025). Consequently, recent works propose to utilize synthetic data to train the model (Tang et al., 2024a; Lei et al., 2025). However, these methods often lack well-defined data quality control strategies. This may result in a low-quality training set, such as ignoring data diversity and retaining too many simple instances, ultimately limiting the model's abilities in complex detection scenarios.

In this paper, we introduce a cost-efficient and effective model **FaithLens** for faithfulness hallucination detection. As shown in Figure 1, FaithLens not only predicts whether a claim is hallucinated, but also produces the corresponding explanation for users to localize errors and understand why certain claims are considered hallucinations. To this end, we begin by leveraging open-source datasets and querying an advanced model to synthesize samples with explanations. Next, to ensure data quality and the effectiveness of the trained model across diverse scenarios, we design a targeted data filtering pipeline that jointly ensures label correctness, the synthesized explanation quality, and data diversity. After using this well-curated dataset for supervised fine-tuning (SFT) as a cold start, we further strengthen the model through a rule-based reinforcement learning (RL) stage. Specifically, we introduce a prediction correctness reward to improve detection performance and an explanation quality reward to enhance the informativeness and clarity of generated explanations. The correctness reward is computed directly from the model prediction, ensuring that the training signal explicitly reinforces accurate hallucination detection. Meanwhile, our proposed explanation quality reward thoroughly assesses a generated explanation by checking if it can help a novice-level model (e.g., untuned Llama-3.1-8B-Inst (Grattafiori et al., 2024) model) correctly predict the corresponding label. If the generated explanation enables a novice-level model to generate the correct prediction, it indicates that the explanation is sufficiently coherent and informative to convey the relevant evidence. By utilizing these two rewards together with a format reward, our model can achieve a unique combination of trustworthiness and effectiveness.

We evaluate the effectiveness of our proposed FaithLens on 12 diverse faithfulness hallucination detection tasks from LLM-AggreFact (Tang et al., 2024a) and HoVer (Jiang et al., 2020). Experiments show that our 8B-parameter FaithLens achieves state-of-the-art performance, even surpassing advanced LLMs such as GPT-4.1 and o3 (Jaech et al., 2024) with much lower cost. Also, FaithLens can offer high-quality explanations that are informative and coherent, providing users with a clear understanding of why a claim is considered hallucinated.

2 Task Formulation

Given the grounding document doc and the LLM-generated claim c, we consider c to be faithful to doc if a generic reader would affirm the statement "According to the given doc, c is true". Conversely, c is considered hallucinated if it contradicts, misinterprets, or cannot be verified using doc.

Previous works (Laban et al., 2022; Zha et al., 2023; Tang et al., 2024a; Lei et al., 2025; Seo et al., 2025) formulate such hallucination detection as a binary classification task. The goal is to train model $\mathcal{M}$ to estimate the conditional probability:

$$P_{\mathcal{M}}(y \mid doc, c), \tag{1}$$

where $y \in \{0, 1\}$ denotes whether the provided claim c is faithful (1) or hallucinated (0) with respect to the given document doc.

In this work, we extend the standard binary classification formulation to not only predict whether a claim c is faithful or hallucinated, but also provide a corresponding explanation e that justifies the prediction from our model $\hat{\mathcal{M}}$. Formally,

$$P_{\hat{\mathcal{M}}}(e, y \mid doc, c), \tag{2}$$

where $y \in \{0, 1\}$ is the prediction and e is a textual explanation that support the prediction. This formulation allows the model to provide explainable outputs that are informative to users, improving trustworthiness in hallucination detection.

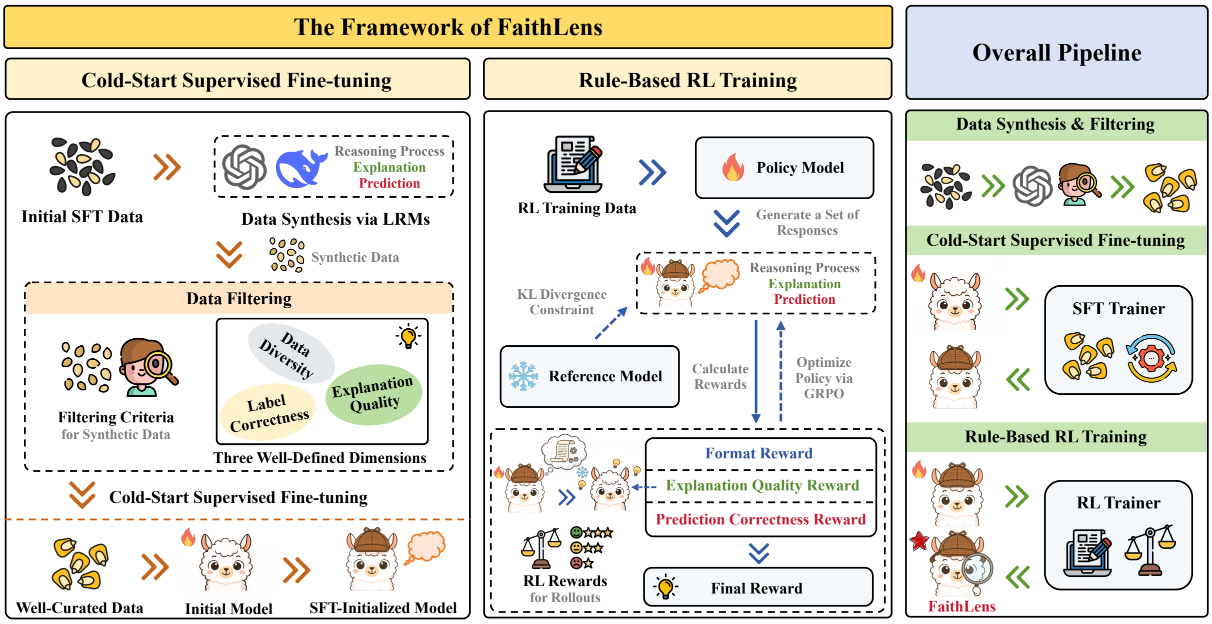

Figure 2: **The Overall Process of Training FaithLens**, including (1) **Cold-Start SFT**: We first synthesize high-quality data with explanations used for the SFT stage. (2) **Rule-Based RL Training**: We further refine the model using a rule-based RL approach with the designed rewards for both prediction correctness and explanation quality.

3 Methodology

In this paper, we build a cost-efficient and effective hallucination detection model **FaithLens** that can jointly determine whether the claim is faithful or hallucinated, and provide corresponding explanations to improve trustworthiness. As shown in Figure 2, we include two key stages to train FaithLens without human efforts: (1) A training data synthesis pipeline that first generates data with explanations, then uses a well-defined data filtering strategy to ensure data quality, and finally starts the SFT stage as a cold start (§ 3.1); (2) A rule-based RL stage to further optimize model effectiveness and trustworthiness, using rewards from both prediction correctness and explanation quality (§ 3.2).

3.1 Cold-Start Supervised Fine-tuning

To equip the model with the ability to detect hallucinations and generate corresponding explanations, we start by training the model via the SFT stage.

3.1.1 Data Synthesis

Previous works (Tang et al., 2024a; Lei et al., 2025) formulate such hallucination detection as a binary classification task and cannot provide corresponding explanations. Thus, existing training datasets only provide prediction labels without corresponding explanations. To bridge this gap, we first leverage the advanced large reasoning models (LRMs), e.g., DeepSeek-V3.2-Think (DeepSeek-AI et al., 2025), to synthesize data with explanations. We provide the LLM with the prompt that consists of the given document doc and a claim c from open-source training datasets (Lei et al., 2025), allowing it to autoregressively provide its chain-of-thought (CoT) $\hat{cot}$, an explanation $\hat{e}$, and its own predicted label $\hat{y}$. By doing so, we can obtain a synthesized sample $\hat{s}$ used for the cold-start SFT stage.

3.1.2 Data Filtering

However, even if we apply well-designed prompts, the synthesized data without quality control could still be noisy or useless. Thus, we propose a well-defined strategy to avoid low-quality samples without human effort. We consider three dimensions to ensure data quality, including (1) label correctness, (2) explanation quality, and (3) data diversity.

Label Correctness. For each synthesized sample, we first compare the predicted label $\hat{y}$ from the LLM with the ground-truth label y_{gt} provided in the original dataset. If the two labels are inconsistent, we directly discard the sample $\hat{s}$ along with the generated CoT $\hat{cot}$ and explanation $\hat{e}$. Formally,

$$F_{label}(\hat{s}) = \mathbb{I}\Big\{\hat{y} = y_{\text{gt}}\Big\}, \quad (3)$$

where $\mathbb{I}$ is the indicator function for filtering low-quality data that do not match the target. If the label from the LLM is incorrect, the related CoT and explanation may appear coherent, but they are internally aligned with an incorrect prediction. Including these samples would cause the model to learn incorrect patterns, which would reduce its detection effectiveness and explanation quality.

Explanation Quality. After ensuring the label correctness, we further focus on the explanation quality to prevent low-value or misleading explanations from the training data. We evaluate the quality of explanations by testing whether they can help the model $\mathcal{M}$ used for training (e.g., Llama-3.1-8B-Instruct) to make correct predictions. Specifically, we first measure the model's perplexity for the ground-truth label using only the document doc, claim c, and the synthetic CoT $\hat{cot}$:

$$\mathrm{PPL}_{\text{w/o. exp}} = \mathrm{PPL}_{\mathcal{M}}(y_{\text{gt}} \mid doc, c, \hat{cot}), \quad (4)$$

which indicates the model's confidence in generating the correct label. We then include the synthesized explanation $\hat{e}$ as the input and compute the model's perplexity again, i.e.,

$$\mathrm{PPL}_{\text{w. exp}} = \mathrm{PPL}_{\mathcal{M}}(y_{\text{gt}} \mid doc, c, \hat{cot}, \hat{e}), \quad (5)$$

which reflects the model's confidence in generating the correct label based on the tested explanation. We retain only the samples $\hat{s}$ with explanations that lower model perplexity on correct labels:

$$F_{exp}(\hat{s}) = \mathbb{I}\{\mathrm{PPL}_{\text{w. exp}} < \mathrm{PPL}_{\text{w/o. exp}}\}, \quad (6)$$

where $\mathbb{I}$ is the indicator function for filtering data with low-quality explanations. This indicates that the explanation makes the model more confident in the correct answer, showing that the explanation is both informative and high-quality. In this way, our method is able to filter out low-quality explanations, ultimately ensuring that the cold-started model can provide high-quality explanations.

Data Diversity. Although filtering for label correctness and explanation quality improves the reliability of individual samples, it may also lead to distribution bias, where the retained data focus on specific tasks and hallucination patterns, ultimately limiting the model's cross-task generalization. For instance, the filtering for label correctness can retain too many easy samples, reducing the model's abilities in complex hallucination scenarios.

Thus, we consider the diversity of the given document doc and claim c, since faithfulness hallucinations arise from their semantic relationship. We adopt a clustering-based approach to preserve data diversity, which can identify semantically close document-claim pairs (doc, c) and form clusters for different types of data. For each (doc, c) pair from the sample $\hat{s}$, we first use a sentence embedding model to map it to a dense vector. We utilize the obtained embeddings to employ the K-Medoids algorithm (Park and Jun, 2009) and cosine similarity to get different clusters and their corresponding medoids, i.e., the most centrally located samples in the clusters. Then, we use the K medoids to construct a probe set $\mathcal{S}_p = \{\hat{s}'_1, ..., \hat{s}'_K\}$, then utilize this set to evaluate whether a tested sample $\hat{s}$ can help diverse samples within probe set $\mathcal{S}_p$ towards correct labels. Specifically, we first infer each probe sample into the model $\mathcal{M}$ and compute the perplexity of the ground-truth labels:

$$\mathrm{PPL}(\hat{s'_i}) = \mathrm{PPL}_{\mathcal{M}}(\hat{y}'_i \mid doc'_i, c'_i, \hat{cot}'_i, \hat{e}'_i), \quad (7)$$

where $\hat{s}'_i = (doc'_i, c'_i, \hat{cot}'_i, \hat{e}'_i, \hat{y}'_i)$ denotes the i-th sample in probe set $\mathcal{S}_p$. Next, we incorporate the candidate sample $\hat{s}$ as an in-context demonstration and recompute the perplexity:

$$\mathrm{PPL}(\hat{s'_i} \mid \hat{s}) = \mathrm{PPL}_{\mathcal{M}}(\hat{y}'_i \mid \hat{s}, doc'_i, c'_i, \hat{cot}'_i, \hat{e}'_i), \quad (8)$$

where a decrease in perplexity indicates that $\hat{s}$ provides complementary information that helps the model better predict the correct label for sample $\hat{s}'_i$. Finally, we count the number of probe samples whose perplexity decreases and retain $\hat{s}$ if it improves a sufficient portion of the probe set:

$$F_{div}(\hat{s}) = \mathbb{I}\left\{\left|\{\hat{s}'_i \in \mathcal{S}_p \mid \mathrm{PPL}(\hat{s}'_i \mid \hat{s}) < \mathrm{PPL}(\hat{s}'_i)\}\right| \geqslant \frac{K}{2}\right\}, \quad (9)$$

where $\mathbb{I}$ is the indicator function. In this way, we can ensure that the retained samples have a positive impact across different types of data. Consequently, training the model on such diversified and informative samples enhances its ability to maintain strong performance across different tasks.

Fine-tuning. Finally, we apply these three proposed filtering criteria to ensure the data quality, then fine-tune the model on quality-checked training data $\mathcal{D}$, to get the initialized detection model:

$$\mathcal{L}_{\text{SFT}} = -\mathbb{E}_{\hat{s}\sim\mathcal{D}}[\log \mathcal{M}(\hat{cot}, \hat{e}, y_{\text{gt}} \mid doc, c)]. \quad (10)$$

Thus, the model is equipped with the ability to detect hallucinations and generate explanations.

3.2 Reinforcement Learning Training

The SFT-initialized model can easily memorize the simple training samples and struggles to generalize to complex detection tasks. Also, the model may generate correct explanations but often lacks clarity or informativeness, as it is trained to imitate training data rather than explicitly optimize for explanation quality. To further enhance effectiveness and trustworthiness, we frame it as a rule-based RL problem and propose well-designed rewards from prediction correctness and explanation quality.

3.2.1 Reinforcement Learning Protocol

For the RL training of LLMs, policy optimization methods such as PPO (Schulman et al., 2017) and GRPO (Shao et al., 2024) have been well-explored. Given the advantages of GRPO, e.g., eliminating the need for a reward model, we utilize the GRPO algorithm to optimize our model $\mathcal{M}_{\text{ours}}$.

For each document-claim pair (doc, c), the detection model generates a group of G explanations $\{e_1, \ldots, e_G\}$, and G candidate corresponding predictions $\{p_1, \ldots, p_G\}$. Each output is evaluated using a designed composite rule-based reward (§ 3.2.2). GRPO utilizes the relative performance of candidates within the group to compute an advantage A_i for each output, guiding policy updates according to the following objective:

$$\mathcal{L}_{\text{GRPO}}(\mathcal{M}_{\text{ours}}) = \mathbb{E}_{(doc,c),\{e_i,p_i\}\sim\mathcal{M}_{\text{old}}}\left[\frac{1}{G}\sum_{i=1}^{G}\mathcal{L}_i - \beta\mathbb{D}_{KL}(\mathcal{M}_{\text{ours}}||\mathcal{M}_{\text{ref}})\right], \quad (11)$$

$$\mathcal{L}_i = \min\left(w_i A_i, \text{clip}(w_i, 1-\epsilon, 1+\epsilon)A_i\right), \quad (12)$$

where $w_i = \frac{\mathcal{M}_{\text{ours}}(e_i,p_i|doc,c)}{\mathcal{M}_{\text{old}}(e_i,p_i|doc,c)}$, $\mathcal{M}_{\text{ref}}$ is the reference policy (i.e., the initialized model), $\mathcal{M}_{\text{old}}$ is the policy before the update, ϵ and β are hyperparameters for the update step and divergence regularization, and A_i is estimated advantage within the group.

3.2.2 Reward Design

Having a well-designed reward is key to the effectiveness of RL training (Kimi-Team et al., 2025). An intuitive method is to use a correctness reward to check whether the prediction from the model is correct, ensuring the models can achieve better detection capabilities. However, this method cannot ensure that the generated explanations are high-quality, as the training signal only explicitly reinforces accurate hallucination detection. Meanwhile, directly evaluating the quality of free-form explanation via the rule-based verification continues to pose an unresolved challenge (OpenAI, 2025). To achieve the balance of trustworthiness and effectiveness, we introduce a prediction correctness reward to improve detection performance and an explanation quality reward to enhance the informativeness and clarity of generated explanations.

Prediction Correctness Reward. This reward assesses whether the detection prediction y_{pred} from the model matches the ground-truth answer y_{gt}, ensuring that the training signal explicitly reinforces accurate hallucination detection. Formally,

$$R_{\text{pred}} = \begin{cases} 1 & \text{if } y_{\text{pred}} = y_{\text{gt}}, \\ 0 & \text{otherwise.} \end{cases} \quad (13)$$

In this way, we can further enhance the model's prediction accuracy beyond SFT, leading to more reliable detection across diverse scenarios.

Explanation Quality Reward. Directly evaluating the quality of free-form content via the rule-based verification remains challenging. Thus, we attempt to use the proposed explanation quality reward to evaluate it implicitly. Specifically, we thoroughly assess a generated explanation by checking if it can help a novice-level model $\mathcal{M}_{\text{nov}}$ (e.g., Llama-3.1-8B-Instruct) correctly predict the ground-truth answer. The idea behind this reward is that if the generated explanation e enables a novice-level model to generate the correct prediction, it indicates that the explanation is sufficiently coherent and informative for conveying the relevant evidence. Formally,

$$R_{\text{exp}} = \begin{cases} 1, & \text{if } y_{\text{pred}}^{\mathcal{M}_{\text{nov}}}(doc, c, e) = y_{\text{gt}}, \\ 0, & \text{otherwise,} \end{cases} \quad (14)$$

where $y_{\text{pred}}^{\mathcal{M}_{\text{nov}}}(doc, c, e)$ denotes the final binary prediction produced by the novice-level model conditioned on the provided document doc, claim c, and generated explanation e. This ensures that only high-quality explanations that are sufficiently coherent and informative are rewarded.

Format Reward. To enforce the desired output format, we assign a format reward to evaluate whether the whole generated response contains the proper tags described in the prompt. Formally,

$$R_{\text{format}} = \begin{cases} 1, & \text{if correct formatting,} \\ 0, & \text{if incorrect formatting.} \end{cases} \quad (15)$$

Final Reward. Finally, we use the sum of these three rewards as the final composite reward R_{final}:

$$R_{\text{final}} = R_{\text{pred}} + R_{\text{exp}} + R_{\text{format}}. \quad (16)$$

By doing so, we can leverage the well-designed rewards to improve both the detection performance and explanation quality, achieving a distinctive balance of effectiveness and trustworthiness.

Model	Agg-CNN	Agg-XSum	Claim Verify	Expert QA	FC-GPT	LfQA	RAG Truth	Reveal	Tofu-MediaS	Tofu-MeetB	Wice	HoVer	Overall Std (σ) ↓	Overall Avg (μ) ↑
The State-of-the-Art LLMs														
GPT-4o	62.3	74.9	78.3	68.3	86.0	75.0	81.8	86.9	71.5	76.9	77.9	73.6	7.0	76.1
o1	68.3	76.7	77.1	72.3	85.0	76.1	79.6	85.9	65.8	76.1	78.7	79.9	5.9	76.8
DeepSeek-V3.2-Non-Think	75.5	65.5	75.4	74.4	87.9	72.9	80.8	91.0	65.5	82.9	72.7	76.7	7.8	76.8
DeepSeek-V3.2-Think	86.8	76.8	88.0	80.7	88.0	77.4	85.9	92.1	83.5	91.4	81.8	80.0	5.1	84.4
Claude-3.7-Sonnet	75.6	73.6	83.7	74.4	86.9	86.0	87.0	88.0	85.4	84.0	86.0	80.2	5.3	82.6
Llama-3.1-405B-Inst	65.5	71.6	80.7	68.8	82.0	76.3	80.7	84.0	67.3	78.8	72.5	81.6	6.4	75.8
GPT-4.1	74.1	73.6	81.6	80.3	91.3	81.1	89.1	93.2	75.9	86.3	86.4	82.6	6.5	83.0
o3-mini	64.4	81.5	80.2	73.0	86.0	81.5	80.7	84.8	77.1	78.3	84.0	78.5	5.9	79.2
o3	67.8	77.2	83.3	79.6	86.9	87.7	80.6	92.2	82.9	83.8	82.1	81.1	6.0	82.1
Specialized Detection Models														
AlignScore	45.7	68.0	79.8	75.0	83.7	86.6	83.6	92.2	75.8	76.5	67.3	73.3	12.0	75.6
FactCG	76.9	68.1	76.2	75.3	89.0	86.5	79.6	90.0	79.1	71.9	72.2	73.1	7.0	78.2
MiniCheck	70.0	72.7	85.6	72.9	86.8	89.0	86.9	91.0	74.3	77.8	**85.9**	74.9	7.5	80.7
ClearCheck	72.8	78.6	85.4	72.7	87.9	87.0	83.8	87.0	67.8	75.8	81.8	80.3	6.6	80.1
FaithLens	**84.9**	**79.0**	**89.4**	**79.6**	**92.4**	**92.1**	**86.8**	**92.2**	**85.1**	**87.2**	85.6	**82.9**	**4.6**	**86.4**
Δ Compared to Llama-3.1-8B-Inst.	+41.8	+30.4	+25.8	+29.8	+22.6	+46.0	+34.8	+14.0	+34.3	+24.9	+37.4	+19.3	-6.3	+30.1

Table 1: **Effectiveness Results.** We report experimental results on 12 various datasets from LLM-AggreFact and HoVer benchmarks. Bold numbers indicate the best performance of specialized detection models. Our FaithLens simultaneously outperforms other specialized models and advanced LLMs such as GPT-4.1 and o3.

4 Experiments

In this section, we conduct experiments and analyses to show the advantages of our FaithLens.

4.1 Experiment Settings

Evaluation. We use LLM-AggreFact (Tang et al., 2024a) and HoVer (Jiang et al., 2020) as evaluation benchmarks. LLM-AggreFact contains 11 different faithfulness hallucination detection tasks, such as summarization, RAG, and dialogue, to fully evaluate the effectiveness and generalization. HoVer benchmark further focuses on more complex multi-hop reasoning tasks. Also, Seo et al. (2025) found that the original two benchmarks contain a large number of annotation errors and ambiguous examples. Thus, we use the refined version of LLM-AggreFact and HoVer, then apply *macro-F1* as our metric, following Seo et al. (2025) for a fair comparison. More details are shown in Appendix B.

Baselines. We compare several baselines, including **(1) The State-of-the-Art LLMs**: We evaluate the most advanced LLMs, including GPT-4o, o1, GPT-4.1, o3-mini, o3, DeepSeek-V3.2, Llama-3.1-405B-Inst, and Claude-3.7-Sonnet (Anthropic, 2025). **(2) Specialized Detection Models:** We further compare open-source detection models. AlignScore (Liu et al., 2023a) trains a 355M-parameter detection model on 4.7M data from 7 different tasks. MiniCheck (Tang et al., 2024a) proposes a data synthesis pipeline and uses 35K private data synthesized from Llama-3.1-405B-Inst to train a 7B model. FactCG (Lei et al., 2025) uses the context graph to generate complex multi-hop synthetic data to train a 435M-parameter model. ClearCheck (Seo et al., 2025) uses 57K ANLI examples, 25K private data, and CoT distilled from Llama-3.1-405B-Inst to train Llama-3.1-8B-Inst with multi-task training. Details are shown in Appendix C.

Implementation Details. For a fair comparison with previous works (Seo et al., 2025), our main experiments are conducted on Llama-3.1-8B-Inst. For training FaithLens, we use the same training data as FactCG (Lei et al., 2025), as it is based on public data instead of private ones. Specifically, we utilize the same ANLI (Nie et al., 2020) subset, C2D, and D2C sets following Lei et al. (2025) as our initial SFT data, then use our explanation synthesis and filtering strategies, and finally apply SFT on the filtered data. For the RL stage, we use the CG2C-MHQA and CG2C-Doc sets from Lei et al. (2025) to train our SFT-initialized model. In this way, we use the same data as FactCG (Lei et al., 2025) throughout the training process, without introducing additional data. For computing our explanation quality reward (§ 3.2.2), we also use Llama-3.1-8B-Inst as our novice-level model. We use DeepSeek-V3.2-Think instead of other advanced LRMs (e.g., o3) to synthesize data (§ 3.1.1), as these models do not allow us to access the CoT content. More details are shown in Appendix D, e.g., the sentence embedding model used for data diversity (§ 3.1.2) and hyperparameters.

4.2 Results

Effectiveness Results. As shown in Table 1, our FaithLens achieves SOTA overall performance on 12 different tasks. Compared with specialized models, FaithLens not only achieves better results on cross-task scenarios (LLM-AggreFact), but also

Model	Read.	Help.	Info.	Avg
GPT-4o	94.4	84.8	73.0	84.1
o1	91.8	81.6	75.4	82.9
DeepSeek-V3.2-Non-Think	93.0	90.6	84.2	89.3
DeepSeek-V3.2-Think	94.4	92.6	83.0	90.0
Claude-3.7-Sonnet	95.7	94.6	83.7	93.5
Llama-3.1-405B-Inst	90.6	79.6	81.0	83.7
GPT-4.1	99.8	95.2	83.2	92.7
o3-mini	94.6	88.2	71.6	84.8
o3	97.6	97.6	85.2	93.5
ClearCheck	85.2	79.0	67.8	77.3
CoT from FaithLens	81.4	76.6	68.4	75.5
FaithLens	**92.4**	**93.4**	**85.4**	**90.4**
△ Compared to Llama-3.1-8B-Inst.	+17.1	+21.1	+17.2	+18.5

Table 2: **Explainability Results.** We use GPT-4.1 to evaluate the generated explanations from three dimensions, including readability (Read.), helpfulness (Help.), and informativeness (Info.). Bold numbers indicate the best performance of specialized detection models.

Model	Cost($)	Model	Cost($)
GPT-4o	7.3	o1	140.6
DeepSeek-V3.2-Non-Think	0.8	o3-mini	5.9
DeepSeek-V3.2-Think	1.2	o3	8.8
GPT-4.1	11.4	Claude-3.7-Sonnet	14.5
Llama-3.1-405B-Inst	16.7	**FaithLens (8B)**	**0.1**

Table 3: **Inference Efficiency Results.** Inference cost on 1.2K samples from 12 datasets. FaithLens delivers SOTA performance with lowest cost ($ 0.8/GPU-hour).

significantly improves the performance in the complex reasoning detection task (HoVer). Meanwhile, FaithLens can achieve better performance than advanced LLMs with much lower cost, e.g., GPT-4.1 and o3. It shows strong generalization abilities, achieving the lowest standard deviation and the most stable performance across tasks.

Explainability Results. We further evaluate the quality of generated explanations using GPT-4.1 as a judge to show the trustworthiness. To ensure the correctness and usability of the explanations, we only evaluate the explanations corresponding to the samples that were correctly predicted by the model. To obtain explanations from advanced LLMs, we adjust the prompts used in effectiveness experiments to require models to generate the explanations before giving their predictions, as FaithLens, which has little to no effect on the model's prediction performance. Most of the specialized models (e.g., MiniCheck) treat hallucination detection as a binary classification task and cannot provide explanations. One exception is ClearCheck, which first generates a CoT and then produces the prediction. Thus, we use the CoT from ClearCheck to evaluate its explainability. Specifically, we consider three dimensions for explanations: readability, helpfulness, and informativeness. As shown in Table 2, FaithLens can produce high-quality explanations

Model	# Data	Data Source	Is Explainable?
AlignScore	4,700K	Public	No
FactCG	52K	Public	No
MiniCheck	35K	Private	No
ClearCheck	82K	Private	Partial
FaithLens	**28K**	**Public**	**Yes**
- w/o. Data Filtering	52K	**Public**	**Yes**

Table 4: **Data Efficiency Results.** Comparison of specialized detection models on training data sizes, data source, and explainability.

Method	Effectiveness		Explainability
	Std ↓	Avg ↑	Avg ↑
Llama-3.1-8B-Inst	10.9	56.3	71.9
Direct SFT on 52K Data	6.1	79.1	N/A
FaithLens	**4.6**	**86.4**	**90.4**
- w/o. Cold-start SFT Stage	5.7	83.4	88.1
- w/o. Data Filtering	6.7	81.2	82.3
- w/o. Label Correctness Filtering	5.3	83.5	86.0
- w/o. Explanation Quality Filtering	4.8	85.8	83.4
- w/o. Data Diversity Filtering	6.4	85.0	89.3
- w/o. Rule-based RL Stage	6.0	82.6	83.8
- w/o. Explanation Quality Reward	5.1	85.7	84.7

Table 5: **Ablation Study.** N/A means the trained model cannot provide the corresponding explanations.

even compared to advanced LLMs. This is because our designed data filtering strategy can ensure the quality of explanations used for the SFT stage. Our explanation quality reward requires LLMs to generate fluent and helpful explanations for a novice model, which further optimizes the quality. In comparison, unsupervised CoT content from FaithLens cannot serve as high-quality explanations, further showing the effectiveness of our design. More details can be found in Appendix E.

Efficiency Results. As shown in Table 3, we compare the inference cost with advanced API-based LLMs. Specifically, our proposed FaithLens delivers SOTA performance with the lowest cost, achieving the balance of effectiveness and efficiency. We also show the comparison of specialized models as shown in Table 4. Our model can achieve reliable performance and provide the corresponding explanations without relying on private data. With our data filtering strategy, our method can efficiently utilize data to achieve better performance. In this way, our FaithLens can achieve efficiency in both inference cost and training data.

4.3 Analysis

Ablation Study. We conduct an ablation study to show the effectiveness of our methods in Table 5. The results reveal that each of our designed components can significantly enhance the model. For our data filtering strategy, we find that each consid-

Method	Original		Decontextualization		Decomposition	
	Std ↓	Avg ↑	Std ↓	Avg ↑	Std ↓	Avg ↑
GPT-4o	7.0	76.1	6.9	76.1	6.6	76.6
o1	5.9	76.8	6.0	76.5	5.6	77.2
GPT-4.1	6.5	83.0	6.5	83.0	6.2	83.3
o3	6.0	82.1	6.0	82.0	5.7	82.5
AlignScore	12.0	75.6	11.9	75.4	11.5	76.1
FactCG	7.0	78.2	6.8	78.0	7.1	78.6
MiniCheck	7.5	80.7	7.5	80.6	7.3	80.8
ClearCheck	6.6	80.1	6.6	80.1	6.4	80.2
FaithLens	**4.6**	**86.4**	**4.6**	**86.4**	**4.4**	**86.6**

Table 6: **Claim Decontextualization and Claim Decomposition Study.** We use GPT-4.1 to perform these two operations for claims as new inputs.

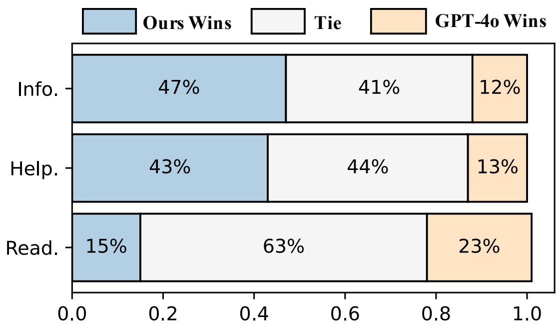

Figure 3: **Human Evaluation.** We compare the explanations from FaithLens and GPT-4o on 120 samples.

Method	Effectiveness		Explainability
	Std ↓	Avg ↑	Avg ↑
Llama-3.1-8B-Inst	10.9	56.3	71.9
Llama-3.1-70B-Inst	8.7	70.1	83.5
Llama-3.1-405B-Inst	6.4	75.8	83.7
FaithLens-8B	**4.6**	**86.4**	**90.4**
Qwen2.5-3B-Inst	9.1	73.3	79.3
Qwen2.5-7B-Inst	11.3	73.9	81.7
Qwen2.5-32B-Inst	8.6	73.1	84.2
FaithLens-3B	**4.9**	**83.4**	**88.3**
FaithLens-7B	**4.2**	**84.9**	**90.3**

Table 7: **Generalization Across Foundation Models.** The impact of different backbones of the trained models.

ered dimension plays its expected role. Specifically, the label correctness filtering affects the model's prediction performance. The explanation quality filtering influences the model's explainability, and the data diversity filtering impacts the consistency of the model's cross-task performance. Meanwhile, the proposed rule-based RL stage with a composite reward can further enhance the performance and explainability of the SFT-initialized model. The designed explanation quality reward effectively improves the quality of corresponding explanations and enhances the final model performance. More detailed results can be found in Appendix F.

Claim Decontextualization and Claim Decomposition Study. We revisit two typical stages in detection pipelines: claim decontextualization and claim decomposition. Decontextualization (Choi et al., 2021) aims to address coreference and ellipsis in the claims, which may make sentences difficult to ground. Decomposition (Min et al., 2023) tries to decompose each claim into atomic facts and use the detection model to predict the label for each atomic fact. If all atomic facts are supported by the document, then the claim is supported; otherwise, the claim is not supported. We use GPT-4.1 to conduct these two operations, then use the modified claims as new inputs. As shown in Table 6, claim decontextualization is not needed for our model as it can effectively capture the context-dependent relations. Also, we can find that claim decomposition can further improve the performance of FaithLens. Details can be found in Appendix G.

Human Evaluation. We conduct the human evaluation for the explanations from FaithLens and GPT-4o on 120 selected samples. For each comparison, the final result is determined by majority voting for three dimensions: readability, helpfulness, and informativeness. Results from Figure 3 show the effectiveness of our method. Details are shown in Appendix H, e.g., evaluation principles.

Generalization Across Foundation Models. As shown in Table 7, using our designed process to train the detection model on different foundational models, e.g., Qwen-2.5-Inst (Yang et al., 2024) and Llama-3.1-Inst, can consistently improve performance compared to the original ones.

Parameter Study, Variant Methods Testing, and Case Study. We also perform these additional analyses in the Appendix J-L to show the effectiveness.

5 Conclusion

In this paper, we introduce a cost-efficient and effective model, FaithLens, for detecting faithfulness hallucinations while providing corresponding explanations for real-world users. We first synthesize training data with explanations and apply a well-defined data filtering strategy to ensure data quality. We then fine-tune the model on these well-curated data as a cold start and optimize it with reinforcement learning, using rewards for both prediction correctness and explanation quality. In this way, FaithLens can deliver advanced detection effectiveness across 12 different tasks and offer high-quality explanations at a much lower cost. Overall, our FaithLens achieves a distinctive balance of trustworthiness, efficiency, and effectiveness.

Limitations

Although FaithLens demonstrates strong empirical results and is widely applicable, it still has some limitations. In this section, we outline these limitations below and explain why they are beyond the scope of this work. First, we focus exclusively on textual faithfulness hallucination detection and do not address multi-modal settings. Extending our FaithLens to multi-modal settings would require fundamentally different grounding signals and explanation formats, which are beyond the scope of this study. To ensure the comparability with prior work, we therefore restrict our investigation to the textual domain. Also, our FaithLens generates its CoT, explanation, and predicted label sequentially. Although this design substantially improves trustworthiness and explainability, it introduces additional inference overhead compared to models of similar size that output only predicted labels. Finally, following standard practice in existing works, FaithLens outputs only binary labels (faithful vs. hallucinated). While more fine-grained hallucination categories may benefit real-world applications, current datasets lack a unified taxonomy for such distinctions. We therefore leave fine-grained hallucination detection as future work. These limitations reflect deliberate choices made to maintain methodological consistency and ensure a fair evaluation, and we view them as promising avenues for extending FaithLens in future research.

References

Anthropic. 2025. Claude 3.7 sonnet system card.

Yauhen Babakhin, Radek Osmulski, Ronay Ak, Gabriel Moreira, Mengyao Xu, Benedikt Schifferer, Bo Liu, and Even Oldridge. 2025. Llama-embed-nemotron-8b: A universal text embedding model for multilingual and cross-lingual tasks. *Preprint*, arXiv:2511.07025.

Baolong Bi, Shaohan Huang, Yiwei Wang, Tianchi Yang, Zihan Zhang, Haizhen Huang, Lingrui Mei, Junfeng Fang, Zehao Li, Furu Wei, Weiwei Deng, Feng Sun, Qi Zhang, and Shenghua Liu. 2025. Context-DPO: Aligning language models for context-faithfulness. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 10280–10300, Vienna, Austria. Association for Computational Linguistics.

Hung-Ting Chen, Fangyuan Xu, Shane A Arora, and Eunsol Choi. 2023. Understanding retrieval augmentation for long-form question answering. *arXiv preprint arXiv:2310.12150*.

Chanyeol Choi, Junseong Kim, Seolhwa Lee, Jihoon Kwon, Sangmo Gu, Yejin Kim, Minkyung Cho, and Jy yong Sohn. 2024. Linq-embed-mistral technical report. *Preprint*, arXiv:2412.03223.

Eunsol Choi, Jennimaria Palomaki, Matthew Lamm, Tom Kwiatkowski, Dipanjan Das, and Michael Collins. 2021. Decontextualization: Making sentences stand-alone. *Transactions of the Association for Computational Linguistics*, 9:447–461.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanjia Zhao, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948.

Shehzaad Dhuliawala, Mojtaba Komeili, Jing Xu, Roberta Raileanu, Xian Li, Asli Celikyilmaz, and Jason Weston. 2024. Chain-of-verification reduces hallucination in large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 3563–3578, Bangkok, Thailand. Association for Computational Linguistics.

Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. 2019. ELI5: Long form question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3558–3567, Florence, Italy. Association for Computational Linguistics.

Charlie George and Andreas Stuhlmueller. 2023. Factored verification: Detecting and reducing hallucination in summaries of academic papers. In *Proceedings of the Second Workshop on Information Extraction from Scientific Publications*, pages 107–116, Bali, Indonesia. Association for Computational Linguistics.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant

Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*.

Yebowen Hu, Timothy Ganter, Hanieh Deilamsalehy, Franck Dernoncourt, Hassan Foroosh, and Fei Liu. 2023. MeetingBank: A benchmark dataset for meeting summarization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16409–16423, Toronto, Canada. Association for Computational Linguistics.

Lei Huang, Xiaocheng Feng, Weitao Ma, Yuchun Fan, Xiachong Feng, Yangfan Ye, Weihong Zhong, Yuxuan Gu, Baoxin Wang, Dayong Wu, Guoping Hu, and Bing Qin. 2025. Improving contextual faithfulness of large language models via retrieval heads-induced optimization. *Preprint*, arXiv:2501.13573.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2024. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Trans. Inf. Syst.* Just Accepted.

Alon Jacovi, Yonatan Bitton, Bernd Bohnet, Jonathan Herzig, Or Honovich, Michael Tseng, Michael Collins, Roee Aharoni, and Mor Geva. 2024. A chain-of-thought is as strong as its weakest link: A benchmark for verifiers of reasoning chains. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4615–4634, Bangkok, Thailand. Association for Computational Linguistics.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. 2024. Openai o1 system card. *arXiv preprint arXiv:2412.16720*.

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

Yichen Jiang, Shikha Bordia, Zheng Zhong, Charles Dognin, Maneesh Singh, and Mohit Bansal. 2020. HoVer: A dataset for many-hop fact extraction and claim verification. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3441–3460, Online. Association for Computational Linguistics.

Ryo Kamoi, Tanya Goyal, Juan Diego Rodriguez, and Greg Durrett. 2023. WiCE: Real-world entailment for claims in Wikipedia. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7561–7583, Singapore. Association for Computational Linguistics.

Kimi-Team, Yifan Bai, Yiping Bao, Guanduo Chen, Jiahao Chen, Ningxin Chen, Ruijue Chen, Yanru Chen, Yuankun Chen, Yutian Chen, Zhuofu Chen, Jialei Cui, Hao Ding, Mengnan Dong, Angang Du, Chenzhuang Du, Dikang Du, Yulun Du, Yu Fan, Yichen Feng, Kelin Fu, Bofei Gao, Hongcheng Gao, Peizhong Gao, Tong Gao, Xinran Gu, Longyu Guan, Haiqing Guo, Jianhang Guo, Hao Hu, Xiaoru Hao, Tianhong He, Weiran He, Wenyang He, Chao Hong, Yangyang Hu, Zhenxing Hu, Weixiao Huang, Zhiqi Huang, Zihao Huang, Tao Jiang, Zhejun Jiang, Xinyi Jin, Yongsheng Kang, Guokun Lai, Cheng Li, Fang Li, Haoyang Li, Ming Li, Wentao Li, Yanhao Li, Yiwei Li, Zhaowei Li, Zheming Li, Hongzhan Lin, Xiaohan Lin, Zongyu Lin, Chengyin Liu, Chenyu Liu, Hongzhang Liu, Jingyuan Liu, Junqi Liu, Liang Liu, Shaowei Liu, T. Y. Liu, Tianwei Liu, Weizhou Liu, Yangyang Liu, Yibo Liu, Yiping Liu, Yue Liu, Zhengying Liu, Enzhe Lu, Lijun Lu, Shengling Ma, Xinyu Ma, Yingwei Ma, Shaoguang Mao, Jie Mei, Xin Men, Yibo Miao, Siyuan Pan, Yebo Peng, Ruoyu Qin, Bowen Qu, Zeyu Shang, Lidong Shi, Shengyuan Shi, Feifan Song, Jianlin Su, Zhengyuan Su, Xinjie Sun, Flood Sung, Heyi Tang, Jiawen Tao, Qifeng Teng, Chensi Wang, Dinglu Wang, Feng Wang, Haiming Wang, Jianzhou Wang, Jiaxing Wang, Jinhong Wang, Shengjie Wang, Shuyi Wang, Yao Wang, Yejie Wang, Yiqin Wang, Yuxin Wang, Yuzhi Wang, Zhaoji Wang, Zhengtao Wang, Zhexu Wang, Chu Wei, Qianqian Wei, Wenhao Wu, Xingzhe Wu, Yuxin Wu, Chenjun Xiao, Xiaotong Xie, Weimin Xiong, Boyu Xu, Jing Xu, Jinjing Xu, L. H. Xu, Lin Xu, Suting Xu, Weixin Xu, Xinran Xu, Yangchuan Xu, Ziyao Xu, Junjie Yan, Yuzi Yan, Xiaofei Yang, Ying Yang, Zhen Yang, Zhilin Yang, Zonghan Yang, Haotian Yao, Xingcheng Yao, Wenjie Ye, Zhuorui Ye, Bohong Yin, Longhui Yu, Enming Yuan, Hongbang Yuan, Mengjie Yuan, Haobing Zhan, Dehao Zhang, Hao Zhang, Wanlu Zhang, Xiaobin Zhang, Yangkun Zhang, Yizhi Zhang, Yongting Zhang, Yu Zhang, Yutao Zhang, Yutong Zhang, Zheng Zhang, Haotian Zhao, Yikai Zhao, Huabin Zheng, Shaojie Zheng, Jianren Zhou, Xinyu Zhou, Zaida Zhou, Zhen Zhu, Weiyu Zhuang, and Xinxing Zu. 2025. Kimi k2: Open agentic intelligence. *Preprint*, arXiv:2507.20534.

Diederik P. Kingma and Jimmy Ba. 2017. Adam: A method for stochastic optimization. *Preprint*, arXiv:1412.6980.

Philippe Laban, Tobias Schnabel, Paul N. Bennett, and Marti A. Hearst. 2022. SummaC: Re-visiting NLI-based models for inconsistency detection in summarization. *Transactions of the Association for Computational Linguistics*, 10:163–177.

Deren Lei, Yaxi Li, Mengya Hu, Mingyu Wang, Vincent Yun, Emily Ching, and Eslam Kamal. 2023. Chain of natural language inference for reducing large language model ungrounded hallucinations. *Preprint*, arXiv:2310.03951.

Deren Lei, Yaxi Li, Siyao Li, Mengya Hu, Rui Xu, Ken Archer, Mingyu Wang, Emily Ching, and Alex Deng. 2025. FactCG: Enhancing fact checkers with graph-based multi-hop data. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5002–5020, Albuquerque, New Mexico. Association for Computational Linguistics.

Anguo Li and Lei Yu. 2025. Summary factual inconsistency detection based on LLMs enhanced by universal information extraction. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 25450–25465, Vienna, Austria. Association for Computational Linguistics.

Fuxiao Liu, Kevin Lin, Linjie Li, Jianfeng Wang, Yaser Yacoob, and Lijuan Wang. 2023a. Aligning large multi-modal model with robust instruction tuning. *arXiv preprint arXiv:2306.14565*.

Nelson Liu, Tianyi Zhang, and Percy Liang. 2023b. Evaluating verifiability in generative search engines. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 7001–7025, Singapore. Association for Computational Linguistics.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023c. G-eval: NLG evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522, Singapore. Association for Computational Linguistics.

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

Chaitanya Malaviya, Subin Lee, Sihao Chen, Elizabeth Sieber, Mark Yatskar, and Dan Roth. 2024. ExpertQA: Expert-curated questions and attributed answers. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 3025–3045, Mexico City, Mexico. Association for Computational Linguistics.

Potsawee Manakul, Adian Liusie, and Mark Gales. 2023. 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*, pages 9004–9017, Singapore. Association for Computational Linguistics.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023.

FActscore: Fine-grained atomic evaluation of factual precision in long form text generation. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. 2023. Mteb: Massive text embedding benchmark. *Preprint*, arXiv:2210.07316.

Diyana Muhammed, Gollam Rabby, and Sören Auer. 2025. Selfcheckagent: Zero-resource hallucination detection in generative large language models. *Preprint*, arXiv:2502.01812.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Çağlar Gulçehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence RNNs and beyond. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, pages 280–290, Berlin, Germany. Association for Computational Linguistics.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, Brussels, Belgium. Association for Computational Linguistics.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial NLI: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4885–4901, Online. Association for Computational Linguistics.

OpenAI. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

OpenAI. 2025. Deep research system card. Technical report, OpenAI.

OpenAI. 2025. Gpt-5 technical report.

Hae-Sang Park and Chi-Hyuck Jun. 2009. A simple and fast algorithm for k-medoids clustering. *Expert systems with applications*, 36(2):3336–3341.

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

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *Preprint*, arXiv:1707.06347.

Wooseok Seo, Seungju Han, Jaehun Jung, Benjamin Newman, Seungwon Lim, Seungbeen Lee, Ximing Lu, Yejin Choi, and Youngjae Yu. 2025. Verifying the verifiers: Unveiling pitfalls and potentials in fact verifiers. *arXiv preprint arXiv:2506.13342*.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *Preprint*, arXiv:2402.03300.

Shuzheng Si, Haozhe Zhao, Gang Chen, Cheng Gao, Yuzhuo Bai, Zhitong Wang, Kaikai An, Kangyang Luo, Chen Qian, Fanchao Qi, Baobao Chang, and Maosong Sun. 2025a. Aligning large language models to follow instructions and hallucinate less via effective data filtering. *Preprint*, arXiv:2502.07340.

Shuzheng Si, Haozhe Zhao, Gang Chen, Yunshui Li, Kangyang Luo, Chuancheng Lv, Kaikai An, Fanchao Qi, Baobao Chang, and Maosong Sun. 2025b. GATEAU: Selecting influential samples for long context alignment. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 7391–7422, Suzhou, China. Association for Computational Linguistics.

Shuzheng Si, Haozhe Zhao, Cheng Gao, Yuzhuo Bai, Zhitong Wang, Bofei Gao, Kangyang Luo, Wenhao Li, Yufei Huang, Gang Chen, et al. 2025c. Teaching large language models to maintain contextual faithfulness via synthetic tasks and reinforcement learning. *arXiv preprint arXiv:2505.16483*.

Shuzheng Si, Haozhe Zhao, Kangyang Luo, Gang Chen, Fanchao Qi, Minjia Zhang, Baobao Chang, and Maosong Sun. 2025d. A goal without a plan is just a wish: Efficient and effective global planner training for long-horizon agent tasks. *Preprint*, arXiv:2510.05608.

Liyan Tang, Tanya Goyal, Alex Fabbri, Philippe Laban, Jiacheng Xu, Semih Yavuz, Wojciech Kryscinski, Justin Rousseau, and Greg Durrett. 2023. Understanding factual errors in summarization: Errors, summarizers, datasets, error detectors. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11626–11644, Toronto, Canada. Association for Computational Linguistics.

Liyan Tang, Philippe Laban, and Greg Durrett. 2024a. Minicheck: Efficient fact-checking of llms on grounding documents. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Liyan Tang, Igor Shalyminov, Amy Wong, Jon Burnsky, Jake Vincent, Yu'an Yang, Siffi Singh, Song Feng, Hwanjun Song, Hang Su, Lijia Sun, Yi Zhang, Saab Mansour, and Kathleen McKeown. 2024b. TofuEval: Evaluating hallucinations of LLMs on topic-focused dialogue summarization. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4455–4480, Mexico City, Mexico. Association for Computational Linguistics.

David Wan, Mengwen Liu, Kathleen McKeown, Markus Dreyer, and Mohit Bansal. 2023. Faithfulness-aware decoding strategies for abstractive summarization. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2864–2880, Dubrovnik, Croatia. Association for Computational Linguistics.

David Wan, Koustuv Sinha, Srini Iyer, Asli Celikyilmaz, Mohit Bansal, and Ramakanth Pasunuru. 2024. ACUEval: Fine-grained hallucination evaluation and correction for abstractive summarization. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 10036–10056, Bangkok, Thailand. Association for Computational Linguistics.

Binjie Wang, Steffi Chern, Ethan Chern, and Pengfei Liu. 2024. Halu-j: Critique-based hallucination judge. *Preprint*, arXiv:2407.12943.

Yuxia Wang, Revanth Gangi Reddy, Zain Muhammad Mujahid, Arnav Arora, Aleksandr Rubashevskii, Jiahui Geng, Osama Mohammed Afzal, Liangming Pan, Nadav Borenstein, Aditya Pillai, Isabelle Augenstein, Iryna Gurevych, and Preslav Nakov. 2023. Factcheck-gpt: End-to-end fine-grained document-level fact-checking and correction of llm output. *ArXiv*, abs/2311.09000.

Zhitong Wang, Cheng Gao, Chaojun Xiao, Yufei Huang, Shuzheng Si, Kangyang Luo, Yuzhuo Bai, Wenhao Li, Tangjian Duan, Chuancheng Lv, Guoshan Lu, Gang Chen, Fanchao Qi, and Maosong Sun. 2025. Document segmentation matters for retrieval-augmented generation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 8063–8075, Vienna, Austria. Association for Computational Linguistics.

Jason Wei, Nguyen Karina, Hyung Won Chung, Yunxin Joy Jiao, Spencer Papay, Amelia Glaese, John Schulman, and William Fedus. 2024a. Measuring short-form factuality in large language models. *Preprint*, arXiv:2411.04368.

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

Yuanhao Wu, Juno Zhu, Siliang Xu, Kashun Shum, Cheng Niu, Randy Zhong, Juntong Song, and Tong Zhang. 2023. Ragtruth: A hallucination corpus for developing trustworthy retrieval-augmented language models. *Preprint*, arXiv:2401.00396.

Rongwu Xu, Zehan Qi, Zhijiang Guo, Cunxiang Wang, Hongru Wang, Yue Zhang, and Wei Xu. 2024. Knowledge conflicts for LLMs: A survey. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8541–8565, Miami, Florida, USA. Association for Computational Linguistics.

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, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. 2024. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *Forty-first International Conference on Machine Learning*.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. 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)*, pages 11328–11348, Toronto, Canada. Association for Computational Linguistics.

Wenxuan Zhou, Sheng Zhang, Hoifung Poon, and Muhao Chen. 2023. Context-faithful prompting for large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 14544–14556, Singapore. Association for Computational Linguistics.

Chenguang Zhu, Yang Liu, Jie Mei, and Michael Zeng. 2021. MediaSum: A large-scale media interview dataset for dialogue summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5927–5934, Online. Association for Computational Linguistics.

Appendix

This appendix is organized as follows.

- In Section A, we detail the related work to comprehensively show our motivation.
- In Section B, we go into detail about the datasets used in our experiments.
- In Section C, we show the details of the baselines during the evaluation.
- In Section D, we list the details of the implementation, e.g., hyperparameters.
- In Section E, we further show the details of the explanation evaluation.
- In Section F, we report the detailed results of the ablation study, e.g., the detailed results.
- In Section G, we go into details about claim decontextualization and claim decomposition study, e.g., the detailed results.
- In Section H, we show the implementation details of human evaluation.
- In Section I, list the details of the generalization test across the foundation models.
- In Section J, we conduct experiments to explore the impact of hyperparameters.
- In Section K, we conduct fine-grained variant method testing to validate the effectiveness of our proposed designs.
- In Section L, we come up with a practical case study to show the effectiveness of FaithLens.

A Related Work

Hallucinations in LLMs. Hallucinations occur when the generated content from LLMs seems believable but does not match factual or contextual knowledge (Ji et al., 2023; Rawte et al., 2023; Huang et al., 2024). Hallucinations in LLMs can be categorized into factuality hallucinations (Min et al., 2023; Wei et al., 2024b) and faithfulness hallucinations (Huang et al., 2025; Si et al., 2025a). Factuality hallucinations arise when LLMs rely solely on their parametric knowledge and generate statements that contradict real-world facts (Wei et al., 2024a). Faithfulness hallucinations occur when the model's output is inconsistent with or unsupported by the given input, such as a grounding document and retrieved evidence (Wan et al., 2023; Zhou et al., 2023; Bi et al., 2025; Si et al., 2025b). LLMs are prone to faithfulness hallucinations across various settings, generating information that cannot be supported by the given context. For instance, in retrieval-augmented generation, models may generate supplementary information that is not supported by retrieved documents (Xu et al., 2024). Even when provided with gold source text—e.g., in summarization or simplification—LLMs still produce inconsistent and hallucinated outputs, exhibiting diverse error patterns across domains (Li and Yu, 2025). In this work, we focus on faithfulness hallucinations, aiming to train an effective and explainable detection model that can assess whether LLM-generated claims remain faithful to the given context.

Hallucination Detection. There are two main approaches to detecting faithfulness hallucinations in LLM-generated outputs. One relies on advanced LLMs evaluating the LLM-generated outputs, like SelfCheckGPT (Manakul et al., 2023), or further leveraging Chain-of-Thought (CoT) strategies to improve effectiveness (Liu et al., 2023c; Dhuliawala et al., 2024; Lei et al., 2023). However, these methods are inefficient for real-world applications because they rely on large and advanced models to achieve reliable performance. Deploying large open-source models requires substantial computing resources, while using advanced API-based models can be very costly. To reduce cost, the other focuses on training cost-efficient detection models. SummaC (Laban et al., 2022) adapts natural language inference (NLI) models for document-level faithfulness evaluation. However, NLI-based approaches struggle with the diverse error patterns and fine-grained faithfulness hallucinations, limiting their robustness across tasks and domains. Recent studies thus turn to synthetic data generation for training more capable detection models. AlignScore (Zha et al., 2023) develops a unified training framework by integrating a large diversity of data sources, resulting in 4.7M training examples from 7 well-established tasks. MiniCheck (Tang et al., 2024a) synthesizes training data using advanced LLMs and outperforms previous work. FactCG (Lei et al., 2025) further improves models by enhancing LLM-generated data complexity using knowledge graphs. ClearCheck (Seo et al., 2025) uses synthetic data and multi-task training, enabling the model to engage in CoT reasoning before answering. However, despite these advances in prediction performance, current models still provide only binary labels without accompanying explanations for real-world users, and often exhibit inconsistent performance across tasks. In this work, we fill these gaps by creating high-quality synthetic data using well-defined data filtering strategies and a carefully crafted rule-based RL stage. This enables us to develop FaithLens, a compact detection model that offers a unique combination of trustworthiness, efficiency, and effectiveness.

B Dataset Details

We introduce 12 various datasets from both LLM-AggreFact and HoVer for our evaluation. According to Seo et al. (2025), label ambiguity and annotation errors in the original datasets can significantly impact the evaluations. Seo et al. (2025) point out that 9.1% of the examples are ambiguous, and 6.6% are mislabeled. Thus, Seo et al. (2025) further constructs a refined and well-labeled version of these two benchmarks. For a fair comparison, we use the cleaned version following Seo et al. (2025) to conduct our experiments. Specifically, the LLM-AggreFact includes 11 different faithfulness hallucination detection tasks, including:

Agg-CNN & Agg-XSum. AggreFact (Tang et al., 2023) is an evaluation benchmark for summarization targeting **CNN(/DM)** (Nallapati et al., 2016) and **XSum** (Narayan et al., 2018). It focuses on the SOTA sets, where documents are from the original CNN and XSum datasets and summaries are generated from SOTA finetuned summarizers, since their analysis suggests that summaries are more challenging to evaluate for hallucination compared to summaries generated by pre-SOTA summarizers.

ClaimVerify. This dataset (Liu et al., 2023b) evaluates the correctness of responses from four generative search engines in answering user queries. The dataset contains annotations on whether check-worthy sentences from the engines' responses can be fully supported by their cited documents.

ExpertQA. It contains responses from 6 different systems to queries curated by experts from 32 fields (Malaviya et al., 2024). These systems answer queries either in a closed-book fashion, with-/without in-line citations, or based on retrieved document(s). For each sentence in the response, the sentence is verified against the concatenation of cited or retrieved document(s), if any.

FC-GPT. FactCheck-GPT (Wang et al., 2023) contains factual consistency annotations for LLMs' responses to search queries. In this dataset, each sentence from LLMs' responses is first decomposed into atomic facts, and those atomic facts are then decontextualized so that they can stand alone.

LfQA. LFQA (Chen et al., 2023) contains LLM-generated responses to questions from the ELI5 dataset (Fan et al., 2019). LLMs generate responses based on documents retrieved by humans, models, or randomly selected. Human annotators then evaluate each sentence in the LLM-generated responses against the corresponding document set, classifying them into supported or not supported.

RAGTruth. It is a hallucination detection corpus in various tasks within the RAG setting (Wu et al., 2023). It comprises naturally generated responses from diverse LLMs using RAG. These responses have undergone meticulous manual annotations at both the individual cases and word levels, incorporating evaluations of hallucination intensity.

Reveal. REVERL (Jacovi et al., 2024) is a benchmark that evaluates the correctness of reasoning chains generated by LLMs in the context of open-domain question-answering. The dataset includes annotations at the sentence level, covering various aspects of response correctness.

Tofu-MediaS & Tofu-MeetB. These two datasets are collected from TofuEval (Tang et al., 2024b). It is a benchmark for dialogue summarization, targeting MediaSum (Zhu et al., 2021) and MeetingBank (Hu et al., 2023). It includes topic-focused dialogue summaries generated by 6 LLMs, with sentence-level annotations by linguists.

Wice. WiCE (Kamoi et al., 2023) is a textual entailment dataset that consists of naturally occurring claims from Wikipedia and their cited documents.

To evaluate the performance in complex reasoning scenarios, we include the HoVer benchmark.

HoVer. HoVer is an open-domain, many-hop hallucination detection dataset built upon the Wikipedia corpus. In HoVer, the claims require evidence to be extracted from as many as four English Wikipedia articles and embody reasoning graphs of diverse shapes. Most of the 3/4-hop claims are written in multiple sentences, which adds to the complexity of understanding long-range dependency relations such as coreference.

C Baseline Details

In our work, we compare several baselines, including both advanced LLMs and specialized detection models. In this part, we will detail the version of these models used and the technical details.

AlignScore. It is an entailment-based model that has been trained on 4.7M data from a wide range of tasks such as NLI, QA, fact verification, and summarization. We use the strongest, largest, and released model trained based on RoBERTa-Large (Liu et al., 2019) in our experiments[1]. Meanwhile, we set the prediction threshold as 0.5, then AlignScore outputs a label of either 0 or 1.

MiniCheck. MiniCheck proposes a data synthesis pipeline to automatically get training samples that reflect the complexity of LLM faithfulness hallucination detection. MiniCheck introduces Claim to Doc (C2D) and Doc to Claim (D2C) generation technologies to generate synthetic documents that require models to be able to check multiple facts in the claim against multiple sentences each, and to generate claims and pair them with portions of these human-written documents, resulting in C2D and D2C sets. Combined with the ANLI training subset, MiniCheck-FT5 can outperform all systems of comparable size and reach GPT-4 performance. The authors further train a 7B-level SOTA model MiniCheck-7B[2] on 35K private data synthesized from Llama-3.1-405B-Inst based on the proposed C2D and D2C technologies. In this paper, we compare the MiniCheck-7B in experiments.

FactCG. This work investigates the difference between state-of-the-art synthetic generated claims and real LLM-generated claims. Then, FactCG proposes a new synthetic data generation approach, CG2C, that leverages the context graph to generate complex multi-hop claims without relying on LLMs to decide data labels, resulting in the CG2C-MHQA and CG2C-Doc sets. Then the authors use the same ANLI subset, C2D set, D2C set directly from MiniCheck, along with proposed CG2C-MHQA and CG2C-Doc sets (totaling 52K data) to train FactCG-DBT based on DeBERTa-v3-Large (He et al., 2020). FactCG-DBT[3] leverages this generated data to achieve state-of-the-art performance compared with models of similar parameter size and even outperforms GPT-4-o, which is used to construct the CG2C dataset. We compare FactCG-DBT in our paper, as it is the only released version of FactCG. Also, we set the prediction threshold as 0.5 for each task.

ClearCheck. Seo et al. (2025) found that a small fine-tuned model underperforms larger models by a huge margin, particularly for instances requiring complex reasoning (e.g., HoVer dataset). Then the authors introduce a simple method to build synthetic multi-hop detection data based on Wikipedia and Llama-3.1-405B-Inst, and experiments show that fine-tuning the model on this data largely improves its performance on examples from the Hover dataset. ClearCheck again uses Llama-3.1-405B-Inst to generate direct answers for the given documents and claims, and then CoT reasoning traces on 57K ANLI examples and 25.2K private synthetic multi-hop data as training data. ClearCheck fine-tunes the Llama-3.1-8B-Inst with multi-task training, enabling the model can provide direct answers or engage in CoT reasoning before answering. We use the released model of ClearCheck-8B[4] to conduct our experiments. Seo et al. (2025) point out that using CoT or providing direct answers does not give different evaluation results; however, CoT makes the verifier output legible to humans so that possible errors can be detected. Thus, we report the results from ClearCheck with CoT and use the corresponding CoT as the explanation to measure the explainability and trustworthiness.

Here, we list the API versions of the advanced LLMs we used as baselines, including *gpt-4o-2024-08-06* for GPT-4o, *o1-2024-12-17* for o1, *gpt-4.1-2025-04-14* for GPT-4.1, *o3-2025-04-16* for o3, *o3-mini-2025-01-31* for o3-mini, *deepseek-chat* for DeepSeek-V3.2-Non-Think, *deepseek-reasoner* for DeepSeek-V3.2-Think, and *claude-3-7-sonnet-20250219* for Claude-3.7-Sonnet.

D Implementation Details

Hyperparameters and Devices. For the designed data synthesis stage, we use DeepSeek-V3.2-Think to prepare the training data with explanations and set the temperature to 1.0, as it can offer the generated CoTs used for the SFT stage. For our data filtering stage, we set the number of clusters K as 10, which is used in data diversity filtering according to Eq.(9). Also, we use the embedding model Llama-Embed-Nemotron-8B (Babakhin et al., 2025) to get the clusters, which is based on the Llama-3.1-8B model. Meanwhile, we report the data remaining after applying the filtering strategy shown in Table 8. Specifically, we sequentially perform Label Cor-

[1] https://huggingface.co/yzha/AlignScore

[2] https://huggingface.co/bespokelabs/Bespoke-MiniCheck-7B

[3] https://huggingface.co/yaxili96/FactCG-DeBERTa-v3-Large

[4] https://huggingface.co/just1nseo/ClearCheck-8B

Model	# Data
Initial Whole Data (i.e., - w/o. Data Filtering)	52,268
Initial SFT Data (i.e., - w/o. Data Filtering)	35,554
Initial Data For RL	16,714
Filtered Data from whole Data Filtering	23,625
Filtered Data from Label Correctness Filtering	14,258
Filtered Data from Explanation Quality Filtering	4,363
Filtered Data from Data Diversity Filtering	5,004
Final Data For SFT	11,929
Final Data For RL	16,714
FaithLens	**28,643**

Table 8: **The Number of The Used Data.** We list the number of filtered data by our proposed filtering strategy and the used data for training FaithLens.

rectness Filtering, Explanation Quality Filtering, and Data Diversity Filtering, resulting in the final filtered numbers. This is because by first applying Label Correctness Filtering, a large number of useless samples can be removed, eliminating the need to compute the metrics for explanation quality and data diversity on all samples. Meanwhile, we only apply the filtering strategy during the SFT stage, as the RL data we selected consists of verified, high-quality, and more challenging samples by Lei et al. (2025). For SFT training, we use the Adam optimizer (Kingma and Ba, 2017) to train our model, with a 1×10^{-5} learning rate with a weight decay of 0.1, and a batch size of 16, steering the training across 3 epochs. We conduct our SFT stage with DeepSpeed+ZeRO3 and BF16. For RL training, the learning rate is set to 1×10^{-6} for the actor. We use a group size G of 7, and the rollout temperature is set to 0.6, which is the same as the temperature during the evaluation stage. Also, for the novice-level model used to compute the explanation quality reward, we set the temperature to 0.6. The mini-batch size is set to 16, a total of 112 across 7 GPUs for 2 epochs, the KL-divergence loss coefficient β is set to 0.001, and ϵ is set to 0.2. The gamma-decay factor α is set to 0.2. To enforce the desired output format, we assign a format reward on the whole generated response to evaluate whether it contains the proper three types of XML tags, as shown in Figure 4. Our experiments are conducted on NVIDIA A800 SXM4 80G GPUs.

Evaluation. During the evaluation for baselines, we infer them twice to report the final results, e.g., AlignScore, or directly use the results from Seo et al. (2025). Meanwhile, for FaithLens, we also infer our model twice to obtain stable results.

Prompt Templates. We use the same prompt template as shown in Figure 4 for training and evaluation of FaithLens. For data synthesis, we use the prompt shown in Figure 5 to query the DeepSeek-V3.2-Think. For data filtering, we use the prompt shown in Figure 6 to evaluate the explanation quality, and utilize the prompt in Figure 7 to evaluate whether a tested sample can help diverse samples towards correct labels in the data diversity filtering. When computing the explanation quality reward, we use the prompt template shown in Figure 8 to assess whether a generated explanation can help a novice model correctly predict the correct answer.

E Explainability Results Details

To assess explanation quality, we use GPT-4.1 as an automatic judge. All judgments reported in § 4.2 are obtained by querying the GPT-4.1 API (version *gpt-4.1-2025-04-14*) with default parameters and asking it to score each explanation along the three dimensions described in the main paper (readability, helpfulness, and informativeness). The exact prompt template we used to query GPT-4.1 for scoring is provided in Figure 9. We report the percentage as the final results. For baselines that by default only produce a binary prediction (i.e., no explanation), including API-based LLMs and Llama-3.1-Inst-series, we modify the prompt used in Seo et al. (2025) so that the models are asked to produce both the explanation and the final binary decision. The modified prompt is shown in Figure 10. We note that requiring an explanation before the final answer has little to no effect on the numeric prediction outcome, as shown in Table 1 and Table 13 while making the output legible for downstream explainability evaluation. At the same time, we also investigate whether there is any bias in our use of LLM-as-a-judge. Therefore, we conduct additional experiments using a different LLM as the judge. Specifically, we used GPT-5-mini API (*gpt-5-mini-2025-08-07*) as the judge. As shown in Table 10, we can observe that helpfulness and informativeness scores remain stable, while the readability scores fluctuate, especially for GPT-4.1. This indicates that when a model is used to evaluate its own outputs, it tends to assign higher scores for readability. Regardless of the LLMs used for scoring, our model consistently achieves improvements and maintains advanced performance compared with API-based LLMs. We also conduct the human evaluation in Appendix H to demonstrate the effectiveness. By combining both automatic and human evaluation results, we can find that the explanations from FaithLens

Model	Agg-CNN	Agg-XSum	Claim Verify	Expert QA	FC-GPT	LfQA	RAG Truth	Reveal	Tofu-MediaS	Tofu-MeetB	Wice	HoVer	Overall Std (σ) ↓	Overall Avg (μ) ↑
Ablation Study														
Llama-3.1-8B-Inst	43.1	48.6	63.6	49.8	69.8	46.1	52	78.2	50.8	62.3	48.2	63.6	10.9	56.3
Direct SFT on 52K Data	72.2	70.1	77.6	76.8	85.9	87.2	80.4	90.9	78.5	75.3	78.1	76.2	6.1	79.1
FaithLens	84.9	79.0	89.4	79.6	92.4	92.1	86.8	92.2	85.1	87.2	85.6	82.9	4.6	86.4
- w/o. Cold-start SFT Stage	82.9	73.9	85.8	78.3	89.1	88.8	85.3	93.3	75.3	84.7	83.3	80.3	5.7	83.4
- w/o. Data Filtering	78.2	72.5	86.9	73.5	87.9	87.7	82.7	90.9	70.9	79.2	85.3	78.6	6.7	81.2
- w/o. Label Correctness Filtering	77.7	82.4	86.0	79.0	92.4	84.1	84.5	93.2	75.9	81.0	84.4	81.2	5.3	83.5
- w/o. Explanation Quality Filtering	84.9	83.4	84.7	76.8	92.4	87.6	85.5	94.4	82.4	88.4	87.8	81.6	4.8	85.8
- w/o. Data Diversity Filtering	82.1	74.7	87.1	79.4	94.6	92.1	91.3	91.1	76.4	85.5	84.4	81.8	6.4	85.0
- w/o. Rule-bsaed RL Stage	78.1	78.0	87.2	79.2	87.0	89.7	82.9	94.4	74.1	79.8	84.4	76.9	6.0	82.6
- w/o. Explanation Quality Reward	84.9	83.1	87.1	76.3	89.1	93.2	90.2	92.1	80.4	82.1	87.8	82.1	5.1	85.7
Claim Decontextualization Study														
GPT-4o	62.3	74.9	78.3	68.6	85.9	75.0	82.0	86.7	71.6	76.7	77.6	73.8	6.9	76.1
o1	68.1	76.6	77.1	71.8	85.0	76.2	78.8	86.3	65.8	76.3	76.6	79.9	6.0	76.5
GPT-4.1	74.1	73.6	81.6	80.3	91.3	81.1	89.1	93.2	75.9	86.3	86.4	82.6	6.5	83.0
o3	67.6	77.3	83.2	79.2	86.9	87.6	80.6	92.1	83.2	82.7	82.1	81.3	6.0	82.0
AlignScore	45.3	67.2	79.8	75.6	83.7	86.7	82.5	91.1	75.9	77.2	67.1	72.4	11.9	75.4
FactCG	76.4	68.6	76.2	75.7	89.0	85.6	78.3	89.7	79.0	71.4	72.5	73.2	6.8	78.0
MiniCheck	70.0	72.7	85.6	72.9	86.8	88.9	86.9	91.0	74.3	77.8	85.6	74.9	7.5	80.6
ClearCheck	72.8	78.6	85.4	72.7	87.9	87.2	83.8	86.7	67.8	75.8	81.8	80.3	6.6	80.1
FaithLens	84.9	79.0	89.4	79.6	92.4	92.1	86.8	92.2	85.1	87.2	85.6	82.9	4.6	86.4
Claim Decomposition Study														
GPT-4o	63.6	75.8	78.2	68.6	85.9	75.1	82.0	86.7	72.6	77.8	78.3	74.1	6.6	76.6
o1	69.2	78.3	77.2	72.3	85.0	76.3	79.6	85.6	66.8	76.9	79.0	80.2	5.6	77.2
GPT-4.1	75.3	74.2	82.0	81.0	91.3	81.5	89.1	93.3	75.9	86.2	86.6	82.8	6.2	83.3
o3	68.9	79.1	82.2	79.6	86.9	87.7	81.7	92.2	82.9	84.6	82.9	81.4	5.7	82.5
AlignScore	47.2	69.2	80.1	75.9	83.7	86.7	84.3	92.2	75.4	76.7	68.1	74.2	11.5	76.1
FactCG	78.2	69.2	76.3	76.1	89.0	87.3	78.3	91.1	79.6	71.9	72.5	73.6	7.1	78.6
MiniCheck	72.0	74.3	86.2	71.6	86.8	88.8	87.1	90.9	74.6	77.0	85.6	74.5	7.3	80.8
ClearCheck	73.5	78.8	86.2	73.3	87.9	87.3	83.8	85.6	68.3	75.3	82.0	80.4	6.4	80.2
FaithLens	85.3	79.5	89.6	80.0	92.4	92.1	86.9	92.0	85.2	87.4	85.6	82.9	4.4	86.6
Generalization Study														
Llama-3.1-8B-Inst	43.1	48.6	63.6	49.8	69.8	46.1	52.0	78.2	50.8	62.3	48.2	63.6	10.9	56.3
Llama-3.1-70B-Inst	56.3	62.8	68.3	62.7	81.9	64.0	72.9	81.8	62.0	78.3	71.4	79.0	8.7	70.1
Llama-3.1-405B-Inst	65.5	71.6	80.7	68.8	82.0	76.3	80.7	84.0	67.3	78.8	72.5	81.6	6.4	75.8
FaithLens-8B	84.9	79.0	89.4	79.6	92.4	92.1	86.8	92.2	85.1	87.2	85.6	82.9	4.6	86.4
Qwen-2.5-3B-Inst	55.2	67.9	78.0	80.1	78.1	78.3	67.0	90.0	65.3	79.5	70.0	70.6	9.1	73.3
Qwen-2.5-7B-Inst	52.0	63.6	78.4	80.7	85.8	79.5	68.0	92.1	62.3	70.1	82.1	72.0	11.3	73.9
Qwen-2.5-32B-Inst	53.8	65.0	76.8	66.1	83.0	71.5	75.0	82.9	69.6	77.8	81.9	74.1	8.6	73.1
FaithLens-3B	82.3	78.3	91.1	77.6	90.2	80.6	85.3	91.1	80.2	83.2	80.3	80.6	4.9	83.4
FaithLens-7B	83.6	79.3	87.6	78.3	91.3	82.1	87.1	92.1	84.2	85.2	84.3	83.1	4.2	84.9

Table 9: **Detailed Effectiveness Results for Ablation Study, Claim Study, and Generalization Study.** We report experimental results on 12 various datasets from LLM-AggreFact and HoVer benchmarks.

can even surpass GPT-4o, especially in terms of helpfulness and informativeness.

F Ablation Study Details

We further provide the full results corresponding to the ablation study summarized in Table 5 of the main paper. The complete results are reported in Table 9 and Table 10. Meanwhile, we find that directly applying SFT on Llama-3.1-8B-Inst with all the data from FactCG (Lei et al., 2025) does not greatly improve performance, indicating that simple scaling does not substantially enhance the capabilities of hallucination detection models. It demonstrates the necessity of introducing a specifically designed method to train the detection model.

G Claim Decontextualization and Claim Decomposition Study Details

Claim Decontextualization. During the faithfulness hallucination detection, phenomena like coreference and ellipsis may make sentences difficult to ground out of context. Previous methods (Choi et al., 2021; Tang et al., 2024a) attempt to address this with an explicit decontextualization step. We prompt GPT-4.1 API (version *gpt-4.1-2025-04-14*) for decontextualization as shown in Figure 11, using the previous claims as context to expand the claim following Tang et al. (2024a). More detailed results can be found in Table 9.

Claim Decomposition. We also experiment with a setting using claim decomposition. In this setting, we decompose each claim c into atomic facts $[af_1, af_2, ..., af_k]$ with the prompt from Kamoi et al. (2023); Tang et al. (2024a) and use the detection model to predict the label for each document-facts pair. If all atomic facts are supported by the document, then the claim is supported, and unsupported otherwise. There are typically 2-4 atomic facts per claim across datasets. We prompt GPT-4.1 API (version *gpt-4.1-2025-04-14*) for decomposition as shown in Figure 12. As shown in our ex-

Model	Read.	Help.	Info.	Avg
Using GPT-5-mini as a Judge				
GPT-4o	90.2	84.6	72.5	82.4
o1	89.2	80.9	75.6	81.9
DeepSeek-V3.2-Non-Think	90.6	90.5	85.1	88.7
DeepSeek-V3.2-Think	91.2	93.1	84.1	89.5
Claude-3.7-Sonnet	90.7	93.8	83.5	89.3
Llama-3.1-405B-Inst	88.6	78.3	81.6	82.8
GPT-4.1	95.3	94.6	82.8	90.9
o3-mini	91.8	88.3	70.8	83.6
o3	93.2	96.9	85.5	91.9
ClearCheck	83.2	78.6	68.2	76.7
CoT from FaithLens	79.2	75.3	67.6	74.0
FaithLens	91.9	93.7	85.6	90.4
Δ Compared to Llama-3.1-8B-Inst.	+18.7	+23.5	+18.0	+20.1
Ablation Study				
Llama-3.1-8B-Inst	75.3	72.3	68.2	71.9
Direct SFT on 52K Data	N/A	N/A	N/A	N/A
FaithLens	92.4	93.4	85.4	90.4
- w/o. Cold-start SFT Stage	90.2	90.2	83.8	88.1
- w/o. Data Filtering	85.2	82.8	78.9	82.3
- w/o. Label Correctness Filtering	88.3	88.2	81.6	86.0
- w/o. Explanation Quality Filtering	85.5	84.3	80.3	83.4
- w/o. Data Diversity Filtering	91.7	91.7	84.5	89.3
- w/o. Rule-bsaed RL Stage	87.8	84.2	79.5	83.8
- w/o. Explanation Quality Reward	88.5	85.6	80.1	84.7
Generalization Study				
Llama-3.1-8B-Inst	75.3	72.3	68.2	71.9
Llama-3.1-70B-Inst	90.8	78.6	81.2	83.5
Llama-3.1-405B-Inst	90.6	79.6	81.0	83.7
FaithLens-8B	92.4	93.4	85.4	90.4
Qwen-2.5-3B-Inst	86.7	78.6	72.6	79.3
Qwen-2.5-7B-Inst	88.6	80.2	76.4	81.7
Qwen-2.5-32B-Inst	90.4	81.6	80.6	84.2
FaithLens-3B	90.6	91.6	82.6	88.3
FaithLens-7B	93.5	92.6	84.8	90.3

Table 10: **Detailed Explainability Results for Using GPT-5-mini as a Judge, Ablation Study and Generalization Study.** We evaluate the generated explanations from three dimensions, including readability (Read.), helpfulness (Help.), and informativeness (Info.). We use GPT-4.1 as a judge for Ablation Study and Generalization Study. N/A means that the trained model cannot provide the corresponding explanations.

periments, using claim decomposition can improve the final results to a certain extent. However, this approach increases the inference time and costs by a factor of 2-4 for different datasets, depending on the average number of atomic facts per claim. We believe it should not be used until it provides a significant accuracy benefit. More detailed results about claim decomposition are shown in Table 9.

H Human Evaluation Details

Evaluating free-form content from LLMs remains challenging. Thus, we conduct a pairwise human evaluation on the 120 samples from 12 different datasets used in our evaluation. We assess these samples across three dimensions: readability, helpfulness, and informativeness. For each comparison, three options are given (FaithLens Wins, Tie, and GPT-4o Wins), and the majority voting determines the final result. The participants follow the principles in Figure 13 to make the decision. We invite three participants pursuing bachelor's or master's degrees to compare the explanations generated by

Model	Read.	Help.	Info.	Avg
Study 1				
FaithLens (K=10)	92.4	93.4	85.4	90.4
- w. Setting K as 6	93.1	91.6	84.6	89.8
- w. Setting K as 14	91.5	92.3	84.4	89.4
- w. Setting K as 20	92.2	91.7	85.1	89.7
Study 2				
FaithLens (- w. Llama-Embed-Nemotron-8B)	92.4	93.4	85.4	90.4
- w. Linq-Embed-Mistral-7B	92.1	92.8	84.2	89.7
- w. Gemini-Embedding-001	92.2	92.6	85.1	90.0
Study 3				
FaithLens (- w. Using Llama-3.1-8B-Inst)	92.4	93.4	85.4	90.4
- w. Using Qwen-2.5-7B-Inst	90.6	90.7	83.9	88.4
- w. Using DeepSeek-V3.2-Think	89.2	88.3	82.1	86.5
Question 1				
- w. Using both CoTs and Explanations for SFT	87.8	84.2	79.5	83.8
- w. Using only the Explanations for SFT	85.3	82.1	78.3	81.9
- w. Using only the CoTs for SFT	81.0	75.8	68.2	75.0
Question 2				
FaithLens (- w. Using Correctness as Metrics)	92.4	93.4	85.4	90.4
- w. Using PPL as Metrics	90.8	91.7	82.2	88.2

Table 11: **Detailed Explainability Results for Parameter Study and Variant Methods Testing.** We use GPT-4.1 to evaluate the generated explanations from three dimensions, including readability (Read.), helpfulness (Help.), and informativeness (Info.).

the models. Before participants begin to make judgments, we describe the principles of our design in detail and ensure that each participant correctly understands the principles. If the final result cannot be determined by majority voting, we will hold a discussion among the participants and vote on the result again. We compare two models during the evaluation, including FaithLens as our method and GPT-4o as the advanced model.

I Generalization Study Details

We explore the impact of different model backbones shown in § 4.3. We also report the detailed results in Table 9 and Table 10. Meanwhile, to get the generated explanations from the initial models, e.g., Llama-3.1-8B-Inst and Qwen-2.5-7B-Inst, we use the same prompt as detailed in Appendix E.

J Parameter Study

In this section, we further conduct a parameter study to evaluate the effectiveness of our designed modules and gain a better understanding of them.

Study 1: The Impact of Different Numbers of Clusters in Data Diversity Filtering. As the only hyperparameter introduced in our proposed method, we further conduct tests on the hyperparameter K introduced in the data diversity filtering stage. As shown in Table 11 and Table 12, our designed data diversity filtering is robust to the hyperparameter K. Meanwhile, increasing the number of clusters may introduce additional computing time used for

Model	Agg-CNN	Agg-XSum	Claim Verify	Expert QA	FC-GPT	LfQA	RAG Truth	Reveal	Tofu-MediaS	Tofu-MeetB	Wice	HoVer	Overall Std (σ) ↓	Overall Avg (μ) ↑
Study 1														
FaithLens (K=10)	84.9	79.0	89.4	79.6	92.4	92.1	86.8	92.2	85.1	87.2	85.6	82.9	4.6	86.4
- w. Setting K as 6	84.3	79.1	88.2	78.3	90.2	91.1	86.0	92.2	84.5	86.2	84.4	82.6	4.4	85.6
- w. Setting K as 14	84.9	79.0	89.1	79.0	91.3	92.1	85.3	91.1	84.0	87.2	86.3	83.1	4.4	86.0
- w. Setting K as 20	85.2	78.6	88.2	80.1	90.2	92.0	85.3	90.7	85.5	88.2	86.3	82.8	4.1	86.1
Study 2														
FaithLens (- w. Llama-Embed-Nemotron-8B)	84.9	79.0	89.4	79.6	92.4	92.1	86.8	92.2	85.1	87.2	85.6	82.9	4.6	86.4
- w. Linq-Embed-Mistral-7B	84.5	79.3	88.2	78.1	91.3	92.2	86.0	92.1	85.1	87.0	85.4	82.6	4.6	86.0
- w. Gemini-Embedding-001	84.5	78.6	88.8	79.8	92.4	91.1	85.3	88.9	84.2	88.1	84.2	81.9	4.3	85.7
Study 3														
FaithLens (- w. Using Llama-3.1-8B-Inst)	84.9	79.0	89.4	79.6	92.4	92.1	86.8	92.2	85.1	87.2	85.6	82.9	4.6	86.4
- w. Using Qwen-2.5-7B-Inst	84.6	79.5	88.2	78.3	90.1	93.2	91.1	90.4	78.6	81.2	86.7	81.3	5.4	85.3
- w. Using DeepSeek-V3.2-Think	84.9	82.7	87.1	82.6	90.2	89.6	87.1	92.1	79.3	84.3	88.3	82.3	3.8	85.9
Question 1														
- w. Using both the CoTs and Explanations for SFT	78.1	78.0	87.2	79.2	87.0	89.7	82.9	94.4	74.1	79.8	84.4	76.9	6.0	82.6
- w. Using only the Explanations for SFT	59.9	71.0	79.8	77.9	85.7	85.1	69.1	94.4	69.4	75.0	75.4	66.8	9.5	75.8
- w. Using only the CoTs for SFT	77.3	74.1	83.5	78.7	86.9	89.8	80.6	92.2	72.4	78.3	83.3	76.8	6.1	81.2
Question 2														
FaithLens (- w. Using Correctness as Metrics)	78.1	78.0	87.2	79.2	87.0	89.7	82.9	94.4	74.1	79.8	84.4	76.9	6.0	82.6
- w. Using PPL as Metrics	77.6	78.0	84.9	77.9	86.9	91.1	83.4	91.1	83.2	88.6	84.4	80.1	4.9	83.9

Table 12: **Detailed Effectiveness Results for Parameter Study and Variant Methods Testing.** We report experimental results on 12 various datasets from LLM-AggreFact and HoVer benchmarks.

calculating the perplexity score, and does not significantly improve the performance.

Study 2: The Impact of Different Embedding Models in the Data Diversity Filtering. We further explore the impact of different embedding models used in data diversity filtering. We compare Llama-Embed-Nemotron-8B (Babakhin et al., 2025), Linq-Embed-Mistral-7B (Choi et al., 2024), and Gemini-Embedding-001[5], which are advanced sentence embedding models according to MTEB leaderboard (Muennighoff et al., 2023). We can find that using different advanced embedding models achieve stable results. Therefore, we recommend using advanced open-source embedding models, e.g., Llama-Embed-Nemotron-8B, for implementation to ensure effectiveness and reduce costs.

Study 3: The Impact of Different Novice-level Models for Explanation Quality Reward. When calculating the explanation quality reward, our motivation is that if the generated explanation enables a novice-level model to produce the correct prediction, it indicates that the explanation is sufficiently coherent and informative for conveying the relevant evidence. Therefore, we further investigate how to select the novice-level model, and what would happen if an expert-level model is used to compute the explanation quality reward. We first explore whether the backbone of the novice-level model needs to be consistent with that of the policy model, that is, whether the policy model and the novice-level model should be homologous models (Yu et al., 2024; Si et al., 2025d). As shown in Table 11 and Table 12, even though both Llama-3.1-8B-Inst and Qwen-2.5-7B-Inst perform poorly on faithfulness hallucination detection and can be considered novice-level models, we can find that using Llama-3.1-8B-Inst as the novice-level model can achieve better performance than using a heterologous model, i.e., Qwen-2.5-7B-Inst. This may be due to different pre-training data, language styles, or sensitivity to instruction formats, which can result in a particular model being unable to correctly predict labels based on the provided explanation. Also, by using the homologous novice-level model, we also avoid having the measurement of explanation quality biased by factors other than the novice-level model's problem-solving skills, such as context windows. We further test using an expert-level model instead of a novice model for reward computing; specifically, we use DeepSeek-V3.2-Think to perform the experiment. We find that using the expert-level model results in suboptimal performance, particularly regarding the quality of the generated explanations. This may be because the expert model can ignore the incorrect explanations provided and still predict the correct label. As a result, low-quality explanations are assigned high rewards, which in turn weakens the quality of the explanations generated by the policy model.

K Variant Methods Testing

In this section, we further explore the variant methods in our designs to gain a better understanding of our proposed FaithLens and design choices, including both SFT and rule-based RL stages.

Question 1: Can We Provide Explanations Directly Instead of Presenting CoTs First? In this

[5]https://ai.google.dev/gemini-api/docs/embeddings

work, we require FaithLens to first generate a CoT, then the corresponding explanation, and finally the predicted answer. In this paradigm, whether it is possible to directly provide the corresponding explanation without first generating a CoT remains an open question. This is because generating a CoT increases inference time, which is critical for real-time hallucination detection. Therefore, in the SFT stage, we remove the CoT module and retain only the explanation and answer parts of the synthetic data (as shown in Figure 14) to train the detection model, allowing the model to directly generate the corresponding explanation and the final prediction. As shown in Table 11 and Table 12, removing the CoT during the cold start stage greatly impairs the model's performance and results in a poorly initialized model. This indicates that explanations alone cannot serve the same role as the CoT in improving faithfulness hallucination detection performance, and further demonstrates that the process of first generating the CoT and then the corresponding explanation is a well-justified design choice. We also provide experimental results of directly using the CoTs for SFT as shown in Table 11 and Table 12. The used prompt is shown in Figure 15. The experiments show that its effectiveness performance is inferior to using both CoTs and explanations for SFT together. This is because our data filtering strategy ensures the quality of explanations, which allows these explanations to help the model achieve better performance. At the same time, using only CoTs for SFT results in the absence of the explanation generation process, and the content from CoTs is difficult to serve as high-quality explanations.

Question 2: Why Don't We Use Perplexity as a Metric for Calculating Rewards During the RL Stage? During the data selection stage, especially in explanation quality filtering and data diversity filtering, we design these data filtering methods based on calculating perplexity scores. One of the main motivations is that, compared to methods that require the model to perform generation and inference for data filtering, perplexity-based approaches can significantly reduce the cost of the filtering process. However, during the RL stage and computing the explanation quality reward, we thoroughly assess a generated explanation by checking if it can help a novice-level model correctly predict the ground-truth answer, rather than merely judging whether the novice model succeeds in reducing the corresponding perplexity on the final predicted answers. Therefore, we further investigate whether this variant approach is effective. Specifically, when calculating the explanation quality reward, if the generated explanation successfully reduces the novice-level model's perplexity on the ground-truth answer, a reward of 1 is assigned; otherwise, the reward is 0. As shown in Table 11 and Table 12, this variant perplexity-based approach is not as effective as the method we proposed in the main text, i.e., using correctness as a metric for explanation quality reward. This may be because reducing perplexity is a simpler task compared to using correctness, which limits the model's ability to explore more effective policies during the rule-based RL stage and thus prevents it from achieving better performance.

L Case Study

We conduct the case study for generated explanations from correctly predicted samples in Figure 16 and Figure 17 to visually show the advantages of FaithLens compared with the advanced LLMs, including GPT-4o and o1.

Case Study from LLM-AggreFact. Figure 16 illustrates a hallucination detection task involving a compound claim regarding the Federal Lanham Act and the FTC Act. FaithLens first summarizes the claim and supporting documents to enhance readability, enabling users to clearly understand the input content. Subsequently, rather than simply pointing out the hallucinatory part (the Lanham Act), FaithLens compares it against other pieces of evidence detailed in the documents (such as the Truth in Lending Act, the Fair Credit Reporting Act, etc.). By explicitly pointing out that the documents list several specific statutes without mentioning the Lanham Act, FaithLens provides a rigorous, evidence-based explanation for its conclusion, using the omission as strong proof and thereby significantly improving helpfulness and informativeness. For baselines, GPT-4o provides only a general summary of the document, while o1 directly offers a brief negative judgment. GPT-4o does not explain the content of the claim, and merely describes what the document includes and notes the absence of mention of the Lanham Act, which severely undermines the readability of the explanation. Furthermore, the generated explanation from GPT-4o does not cite evidence from the original text, which reduces its helpfulness and informativeness. The explanation from o1 merely states that the document does not mention the Lanham Act at all, lacking any breakdown of the claim itself and supplementary details from the document.

Case Study from HoVer. As shown in Figure 17, the explanation from FaithLens demonstrates superior quality compared to the ones from GPT-4o and o1. First, the explanation generated by FaithLens demonstrates better readability by restating the claim at the beginning of the explanation, which enhances clarity and allows readers to directly understand the topic under discussion. It then cites evidence from the document to provide the final conclusion, enhancing the clarity. Additionally, FaithLens analyzes other atomic facts within the claim and clearly points out that, although there is a hallucination regarding 1940 in the claim, some parts of the claim are correct. This prevents users from misunderstanding and increases the informativeness and helpfulness of the generated explanation. In this scenario, GPT-4o adopts a misleading flow, beginning its explanation by validating the correct definition of animation; this creates initial ambiguity regarding the claim's overall truthfulness. The o1 model provides a correct verdict but lacks sufficient explanatory detail to be actionable.

Model	Agg-CNN	Agg-XSum	Claim Verify	Expert QA	FC-GPT	LfQA	RAG Truth	Reveal	Tofu-MediaS	Tofu-MeetB	Wice	HoVer	Overall	
													Std (σ) ↓	Avg (μ) ↑
Results by using the modified prompt shown in Figure 10 for baselines														
GPT-4o	63.2	73.7	78.3	69.2	84.8	75.0	81.2	86.4	70.5	75.3	76.3	74.6	6.5	75.7
o1	67.3	75.6	78.3	73.2	84.8	76.5	80.1	84.3	66.7	76.5	79.2	80.8	5.7	76.9
DeepSeek-V3.2-Non-Think	63.4	66.3	78.0	78.0	89.0	70.0	81.9	91.0	69.6	84.0	71.9	74.8	8.8	76.5
DeepSeek-V3.2-Think	84.3	73.5	87.8	76.8	89.1	77.1	86.8	91.0	77.5	85.2	83.0	74.8	6.0	82.2
Claude-3.7-Sonnet	77.3	75.2	82.6	75.1	87.0	88.9	88.9	87.3	85.1	84.0	87.5	81.2	5.1	83.3
Llama-3.1-405B-Inst	67.3	72.1	81.3	65.6	80.9	77.8	82.3	86.2	65.2	80.1	73.2	81.6	7.2	76.1
Llama-3.1-8B-Inst	42.0	47.2	62.6	48.1	70.1	49.1	52.4	77.2	51.3	64.2	52.1	65.2	10.7	56.8
GPT-4.1	74.1	72.5	88.0	78.4	92.4	84.9	88.0	92.1	80.4	90.1	84.3	82.8	6.6	84.0
o3-mini	65.3	82.1	81.3	74.2	85.6	82.0	81.0	85.3	77.6	79.6	80.9	80.1	5.4	79.6
o3	68.1	80.1	82.6	80.5	87.0	88.9	78.2	92.2	83.6	84.1	83.3	81.5	6.0	82.5

Table 13: **Effectiveness Results by Using the Modified Prompt Shown in Figure 10 for Baselines to Generate the Corresponding Explanations.** Combined with the results from Table 1, we cannote that requiring an explanation before the final answer for LLM-based baselines has little to no effect on the numeric prediction outcome.

Prompt used for training and inference of FaithLens

Determine whether the provided claim is consistent with the corresponding document.
Consistency in this context implies that all information presented in the claim is substantiated by the document. If not, it should be considered inconsistent.

- First, think step by step about whether all the information in the claim is fully supported by the document within <think> and </think> tags.

- Then, please provide an easy-to-understand explanation for your answer within <reason> and </reason> tags.

- Finally, assess the claim's consistency with the document by responding with either "Yes" or "No" and wrap your final answer in <answer> and </answer> tags.

Document: [DOCUMENT]
Claim: [CLAIM]

Figure 4: Prompt used for training and inference of FaithLens.

Prompt used for data synthesis

Determine whether the provided claim is consistent with the corresponding document.
Consistency in this context implies that all information presented in the claim is substantiated by the document. If not, it should be considered inconsistent.

- First, think step by step about whether all the information in the claim is fully supported by the document within <think> and </think> tags.

- Then, please provide an easy-to-understand explanation for your answer within <reason> and </reason> tags.

- Finally, assess the claim's consistency with the document by responding with either "Yes" or "No" and wrap your final answer in <answer> and </answer> tags.

Document: [DOCUMENT]
Claim: [CLAIM]

Figure 5: Prompt used for data synthesis.

Prompts used for our designed explanation quality filtering

Determine whether the provided claim is consistent with the corresponding document.
Consistency in this context implies that all information presented in the claim is substantiated by the document. If not, it should be considered inconsistent.

- First, think step by step about whether all the information in the claim is fully supported by the document within <think> and </think> tags.

- Finally, assess the claim's consistency with the document by responding with either "Yes" or "No" and wrap your final answer in <answer> and </answer> tags.

Document: [DOCUMENT]
Claim: [CLAIM]

<think>[CoT]</think><answer>[Answer]</answer>

- - -

Determine whether the provided claim is consistent with the corresponding document.
Consistency in this context implies that all information presented in the claim is substantiated by the document. If not, it should be considered inconsistent.

- First, think step by step about whether all the information in the claim is fully supported by the document within <think> and </think> tags.

- Then, please provide an easy-to-understand explanation for your answer within <reason> and </reason> tags.

- Finally, assess the claim's consistency with the document by responding with either "Yes" or "No" and wrap your final answer in <answer> and </answer> tags.

Document: [DOCUMENT]
Claim: [CLAIM]

<think>[CoT]</think><reason>[Explanation]</reason><answer>[Answer]</answer>

Figure 6: Prompts used for our designed explanation quality filtering. To assess the explanation quality, we concatenate the generated CoT (<think>) and explanation (<reason>) as the input and compute the perplexity of the corresponding [answer]. The upper part of the prompt is used to measure the model's perplexity for the ground-truth label using only the document doc, claim c, and the synthetic CoT $\hat{cot}$. The lower part of the prompt reflects the model's confidence in generating the correct label by conditioning on the tested explanation, i.e., by concatenating both the synthetic CoT and the corresponding explanation as inputs.

Prompts used for our designed data diversity filtering

Determine whether the provided claim is consistent with the corresponding document.
Consistency in this context implies that all information presented in the claim is substantiated by the document. If not, it should be considered inconsistent.

- First, think step by step about whether all the information in the claim is fully supported by the document within <think> and </think> tags.

- Then, please provide an easy-to-understand explanation for your answer within <reason> and </reason> tags.

- Finally, assess the claim's consistency with the document by responding with either "Yes" or "No" and wrap your final answer in <answer> and </answer> tags.

Document: [DOCUMENT]
Claim: [CLAIM]

<think>[CoT]</think><reason>[Explanation]</reason><answer>[Answer]</answer>

- - -

Determine whether the provided claim is consistent with the corresponding document.
Consistency in this context implies that all information presented in the claim is substantiated by the document. If not, it should be considered inconsistent.

- First, think step by step about whether all the information in the claim is fully supported by the document within <think> and </think> tags.

- Then, please provide an easy-to-understand explanation for your answer within <reason> and </reason> tags.

- Finally, assess the claim's consistency with the document by responding with either "Yes" or "No" and wrap your final answer in <answer> and </answer> tags.

Document: [DOCUMENT]
Claim: [CLAIM]

Example: [Tested Sample]

<think>[CoT]</think><reason>[Explanation]</reason><answer>[Answer]</answer>

Figure 7: Prompts used for data diversity filtering. The upper part of the prompt is used to measure the model's perplexity for the ground-truth label based the document doc', claim c', the synthetic CoT $\hat{cot}'$ and explanation $\hat{e}'$. The lower part of the prompt reflects the model's confidence in generating the correct label by incorporating candidate sample $\hat{s}$ as an in-context demonstration and recompute the perplexity.

Prompt used for computing explanation quality reward

Determine whether the provided claim is consistent with the corresponding document.
Consistency in this context implies that all information presented in the claim is substantiated by the document. If not, it should be considered inconsistent.

- First, please refer to the provided explanation to assist you to answer the question.

- Then, please assess the claim's consistency with the document by responding with either "Yes" or "No". Please wrap your final answer in <answer> and </answer>.

Document: [DOCUMENT]
Claim: [CLAIM]
Explanation: [EXPLANATION]

Figure 8: Prompt used for computing explanation quality reward.

Prompt used for scoring the generated explanations using the LLM as a judge

You are an evaluator. Another model was tasked with assessing whether a source document supports a given claim, and it successfully arrived at the correct determination based on the provided task instruction. The model then generated an explanation for its conclusion. Your role is to evaluate the quality of that explanation along the specified dimensions.

Scoring Criteria:
1. Readability (1–5): The explanation should be written in a clear and well-structured manner that enables the reader to easily follow the reasoning behind the model's conclusion. Beyond sentence fluency, focus on whether the explanation presents ideas in a logical sequence, avoids ambiguity, and makes it straightforward for the user to correctly understand why the model arrived at its prediction.
2. Helpfulness (1–5): The explanation should effectively guide the user to understand why the model arrived at its conclusion. Focus on whether the reasoning is clear and logically connected to the claim and document, enabling the user to act on, adapt, or reconsider the claim if needed.
3. Informativeness (1–5): The explanation should provide detailed, specific, and substantive information relevant to the claim and document. Focus on the richness and completeness of content, such as explicit evidence cited, nuanced reasoning, or contextual details that give a deeper understanding, even beyond what is strictly needed to justify the conclusion.

Output Format (JSON only):
{
"readability": <1-5>,
"helpfulness": <1-5>,
"informativeness": <1-5>
}

Task Instruction (includes the claim and document):
[Task Instruction]

Explanation to Evaluate:
[Explanation_Text]

Figure 9: Prompt used for scoring the generated explanations using the LLM as a judge.

Prompts used for evaluating LLM-based baselines

Instructions:
1. You have been given a STATEMENT and some DOCUMENT.
2. Determine whether the given STATEMENT is supported by the given DOCUMENT. The STATEMENT does not need to be explicitly supported by the DOCUMENT but should be strongly implied by the DOCUMENT.
3. Before showing your answer, think step-by-step and show your specific reasoning. As part of your reasoning, summarize the main points of the DOCUMENT.
4. If the STATEMENT is supported by the DOCUMENT, be sure to show the supporting evidence.
5. After stating your reasoning, restate the STATEMENT and then determine your final answer based on your reasoning and the STATEMENT.
6. Your final answer should be either [Attributable] or [Not Attributable], or [Contradictory].
7. Wrap your final answer in square brackets.

DOCUMENT:
[DOCUMENT PLACEHOLDER]
STATEMENT:
[STATEMENT PLACEHOLDER]

- - -

Instructions:
1. You have been given a STATEMENT and some DOCUMENT.
2. Determine whether the given STATEMENT is supported by the given DOCUMENT. The STATEMENT does not need to be explicitly supported by the DOCUMENT, but should be strongly implied by the DOCUMENT.
3. Before showing your explanation and answer, think step-by-step and show your chain of thought and specific reasoning. As part of your reasoning, summarize the main points of the DOCUMENT.
4. If the STATEMENT is supported by the DOCUMENT, be sure to show the supporting evidence.
5. After stating your reasoning, restate the STATEMENT and then determine your final answer based on your reasoning and the STATEMENT.
6. After your reasoning but before the final answer, provide a human-readable explanation (<explanation>) that clearly and concisely justifies your conclusion, citing specific parts or descriptions from the DOCUMENT that support or contradict the STATEMENT. This explanation should be understandable to a human reader and should not reveal the model's internal chain of thought.
7. Your final answer should be either [Attributable] or [Not Attributable], or [Contradictory].Wrap your final answer in square brackets.
8. Your final output must follow the exact structure: <think>step-by-step reasoning (your internal reasoning)</think> <reason>human-readable justification using evidence from the document</reason> <answer>[Attributable] or [Not Attributable] or [Contradictory]</answer>

DOCUMENT:
[DOCUMENT PLACEHOLDER]
STATEMENT:
[STATEMENT PLACEHOLDER]

Figure 10: Prompts used for evaluating LLM-based baselines. The upper part of prompt is adapted from Seo et al. (2025) and is used to evaluate the effectiveness of LLM-based baselines. The label "Attributable" is mapped to the absence of hallucination, while the labels "Not Attributable" and "Contradictory" are mapped to the presence of hallucination. The prompt below shows our modification to the original prompt, enabling the model to output both an explanation and a final prediction, without affecting the model's final prediction performance.

Prompt used for claim decontextualization

You are provied with a context and a claim. Please first determine if the claim can stand alone whitout the conext. If not, provide a decontextualzied version of the claim that incorporates necessary information from the context to make it self-contained. The revision should be as minimum as possible. Please respond with a JSON format: {“label”: “yes”/“no”, “decontext”: “NA”/decontextualized claim}.

Example 1:
Context: There are many reasons why poetry is important for children. Poetry can help children build confidence through memorizing and reciting poems. It can also provide an easy way for children to remember a lesson or value.
Claim: It can also provide an easy way for children to remember a lesson or value.
Answer: {“label”: “no”, “decontext”: “Poetry can provide an easy way for children to remember a lesson or value.”}

Example 2: Context: Yes, ancient societies had concepts of rights. The concept of rights first appeared in the theory of natural law which existed in the state of nature. In this state, people enjoyed certain rights sanctioned by natural law.
Claim: In this state, people enjoyed certain rights sanctioned by natural law.
Answer: {“label”: “no”, “decontext”: “In the state of nature, people enjoyed certain rights sanctioned by natural law”}

Example 3:
Context: The ancient Greeks had some concept of human rights, although there is no single word in classical Greek that captures the sense of “rights” as it is used in modern political thought. However, Greek customs and institutions provided protection to private property unique in the ancient world, instilling a strong sense of equality. The idea of human rights spread quickly from Babylon to Greece and eventually Rome, where the concept of “natural law” arose.
Claim: The idea of human rights spread quickly from Babylon to Greece and eventually Rome, where the concept of “natural law” arose.
Answer: {“label”: “yes”, “decontext”: “NA”}

Your Turn:
Context: [CONTEXT]
Claim: [CLAIM]
Answer:

Figure 11: Prompt used for claim decontextualization.

Prompt used for claim decomposition

Segment the following sentence into individual facts:

Sentence: Other title changes included Lord Steven Regal and The Nasty Boys winning the World Television Championship and the World Tag Team Championship respectively.
Facts:
- Lord Steven Regal won the World Television Championship.
- The Nasty Boys won the World Tag Team Championship.

Sentence: The parkway was opened in 2001 after just under a year of construction and almost two decades of community requests.
Facts:
- The parkway was opened in 2001.
- The parkway was opened after just under a year of construction.
- The parkway was opened after two decades of community requests.

Sentence: Touring began in Europe in April-June with guitarist Paul Gilbert as the opening act, followed by Australia and New Zealand in July, Mexico and South America in late July-August, and concluding in North America in October-November.
Facts:
- Touring began in Europe in April-June.
- The opening act of the tour was guitarist Paul Gilbert.
- The tour was in Australia and New Zealand in July.
- The tour was in Mexico and South America in late July-August.
- The tour was concluded in North America in October-November.

Sentence: In March 2018, the company partnered With Amazon Web Services (AWS) to offer Al-enabled conversational solutions to customers in India.
Facts:
- The company partnered with Amazon Web Services (AWS) in March 2018.
- The two companies partnered to offer Al-enabled conversational solutions to customers in India.

Sentence: The most significant of these is in Germany, which now has a Yazidi community of more than 200,000 living primarily in Hannover, Bielefeld, Celle, Bremen, Bad Oeynhausen, Pforzheim and Oldenburg.
Facts:
- The most significant of these is in Germany.
- Germany now has a Yazidi community of more than 200,000.
- Yazidi community in Germany lives primarily in Hannover, Bielefeld, Celle, Bremen, Bad Oeynhausen, Pforzheim and Oldenburg.

Sentence: A previous six-time winner of the Nations' Cup, Sebastian Vettel became Champion of Champions for the first time, defeating Tom Kristensen, who made the final for the fourth time, 2-0.
Facts:
- Sebastian Vettel is a previous six-time winner of the Nations' Cup.
- Sebastian Vettel became Champion of Champions for the first time, defeating Tom Kristensen, 2-0.
- Tom Kristensen made the final for the fourth time.

Sentence: [SENTENCE]
Facts:

Figure 12: Prompt used for claim decomposition.

The principles of human evaluation

You are asked to evaluate the responses generated by different models. You should choose the preferred responses according to the following perspectives independently:

1. Readability: The explanation should be written in a clear and well-structured manner that enables the reader to easily follow the reasoning behind the model's conclusion. Beyond sentence fluency, focus on whether the explanation presents ideas in a logical sequence, avoids ambiguity, and makes it straightforward for the user to correctly understand why the model arrived at its prediction.
2. Helpfulness: The explanation should effectively guide the user to understand why the model arrived at its conclusion. Focus on whether the reasoning is clear and logically connected to the claim and document, enabling the user to act on, adapt, or reconsider the claim if needed.
3. Informativeness: The explanation should provide detailed, specific, and substantive information relevant to the claim and document. Focus on the richness and completeness of content, such as explicit evidence cited, nuanced reasoning, or contextual details that give a deeper understanding, even beyond what is strictly needed to justify the conclusion.

Finally, please make a decision among the 3 opinions, including FaithLens Wins, Tie, and GPT-4o Wins.

Figure 13: The principles of human evaluation.

Prompt used for question 1 in variant methods testing

Determine whether the provided claim is consistent with the corresponding document.
Consistency in this context implies that all information presented in the claim is substantiated by the document. If not, it should be considered inconsistent.

- First, please provide an easy-to-understand explanation for your answer within <reason> and </reason> tags.

- Finally, assess the claim's consistency with the document by responding with either "Yes" or "No" and wrap your final answer in <answer> and </answer> tags.

Document: [DOCUMENT]
Claim: [CLAIM]

Figure 14: Prompt used for question 1 in variant methods testing.

Prompt used for question 1 in variant methods testing

Determine whether the provided claim is consistent with the corresponding document.
Consistency in this context implies that all information presented in the claim is substantiated by the document. If not, it should be considered inconsistent.

- First, think step by step about whether all the information in the claim is fully supported by the document within <think> and </think> tags.

- Finally, assess the claim's consistency with the document by responding with either "Yes" or "No" and wrap your final answer in <answer> and </answer> tags.

Document: [DOCUMENT]
Claim: [CLAIM]

Figure 15: Prompt used for question 1 in variant methods testing.

Case study from LLM-AggreFact

Document: {Document Content, approximately 3,500 words}
Claim: The federal Lanham Act and the FTC Act both prohibit unfair or deceptive advertising.

GPT-4o: The document details the FTC Act's provisions against unfair or deceptive advertising practices, explaining the requirements for truthfulness and evidence-based substantiation of claims. It highlights the importance of not misleading consumers and presents various FTC guidelines and rules in support of this. However, it does not mention the Lanham Act, and thus there is nocontent in the DOCUMENT that supports the claim regarding the Lanham Act's role in prohibiting deceptive advertising.

o1: The DOCUMENT references the FTC Act's prohibition on unfair or deceptive advertising but does not mention the Lanham Act at all. Consequently, there is no information confirming that the Lanham Act prohibits such advertising in the DOCUMENT.

FaithLens (Ours): The claim states that both the federal Lanham Act and the FTC Act prohibit unfair or deceptive advertising. The document explicitly discusses the FTC Act, which it mentions prohibits "unfair or deceptive advertising in any medium." This directly supports the claim regarding the FTC Act. However, the document does not mention the Lanham Act at all. it lists other federal laws like the truth in lending act, fair credit billing act, fair credit reporting act, equal credit opportunity act, and others, but no reference is made to the Lanham Act. Since the claim asserts that the Lanham Act also prohibits unfair or deceptive advertising, and this information is not present in the document, the claim is not fully substantiated. For the claim to be consistent, all information in it must be supported by the document, which is not the case here.

Figure 16: Case study from LLM-AggreFact.

Case study from HoVer

Document: Animation is the process of making the illusion of motion and the illusion of change by means of the rapid succession of sequential images that minimally differ from each other. The illusion—as in motion pictures in general—is thought to rely on the phi phenomenon and beta movement, but the exact causes are still unclear. Tom and Jerry: A Nutcracker Tale is a 2007 animated holiday themed fantasy comedy direct-to-video film starring "Tom and Jerry" produced by Warner Bros. Animation and directed by Spike Brandt and Tony Cervone. It is loosely based on the story "The Nutcracker and the Mouse King" by E. T. A. Hoffmann, with Jerry in the role of the Nutcracker, while Tom played one of the King Mouse's (who, in this version, is replaced with a cat) henchmen.
Claim: Tom and Jerry: A Nutcracker Tale, released in 1940, is an example of an Animation motion picture that makes the illusion of motion and change by the rapid succession of sequential images that minimally differ from each other.

GPT-4o: According to the DOCUMENT, "Tom and Jerry: A Nutcracker Tale" is indeed an animated film that employs the technique described—creating the illusion of motion through successive images that minimally differ. However, the STATEMENT incorrectly lists its release year as 1940, whereas the DOCUMENT states it was released in 2007. The incorrect release year makes the STATEMENT contradictory to the DOCUMENT.

o1: The DOCUMENT states that Tom and Jerry: A Nutcracker Tale is a 2007 animated film, which conflicts with the STATEMENT's assertion that it was released in 1940. While it is indeed an example of an animated motion picture (as described in the DOCUMENT's explanation of animation), the mismatch in release date renders the STATEMENT contradictory overall.

FaithLens (Ours): The claim states that "Tom and Jerry: A Nutcracker tal" was released in 1940. However, the document explicitly says the film is a "2007 animated holiday themed fantasy comedy." Since the release year (2007) is directly provided in the document, the claim's assertion that it was released in 1940 is not supported. Additionally, the claim calls the film an "Animation motion picture" and describes it as making motion through "rapid succession of sequential images," which aligns with the document's description of animation. Since the year 1940 is incorrect based on the document, the claim contains information not substantiated by the document.

Figure 17: Case study from HoVer.