Generate but Verify: Answering with Faithfulness in RAG-based Question Answering

Simone Filice simone.filice@tii.ae
Elad Haramaty elad.haramaty@tii.ae
Guy Horowitz guy.horowitz@tii.ae
Zohar Karnin zohar.karnin@tii.ae
Liane Lewin-Eytan liane.lewineytan@tii.ae
Alex Shtoff alexander.shtoff@tii.ae

Technology Innovation Institute

Abstract

Retrieval-Augmented Generation (RAG) enhances LLMs by grounding answers in retrieved passages, which is key in factual Question Answering. However, generated answers may still be unfaithful to the passages, either due to retrieval or generation errors. Many RAG downstream applications rely on assessing answer faithfulness for applying fallback strategies, yet address it implicitly, without a consistent evaluation methodology. We introduce the task of Answering with Faithfulness (AwF), which brings faithfulness prediction to the forefront, explicitly coupling it with answer generation. We define variants of the precision and recall metrics tailored to this task, facilitating direct evaluation and comparison of different AwF methods. We then demonstrate, both theoretically and empirically, that for RAG applications using AwF as a sub-procedure, an improvement to the AwF metrics translates to an improvement to the downstream performance. This results in improved performance for recently published results.

1 Introduction

Retrieval-Augmented Generation (RAG) enhances Large Language Models (LLMs) by grounding answers in an external corpus, ensuring that answers are based on retrieved evidence rather than only the LLM parametric memory. An answer is said to be faithful if it is indeed grounded by the retrieved content. This property is especially critical for factual questions requiring precise information.

Ensuring faithfulness in RAG is challenging, as errors may stem from irrelevant retrievals or unfaithful generations due to hallucinations or misinterpretation. Existing methods—such as adaptive retrieval (Zhang et al., 2023; Shi et al., 2024), chain-of-thought reasoning (Wei et al., 2024), or parallel generation (Lewis et al., 2020)—aim to mitigate such issues and improve answer quality, but typically address faithfulness only implicitly. In this work, we bring the task of faithfulness prediction to the spotlight, treating it as an explicit objective crucial not only for answering quality but also for transparency, by enabling answers to be verifiably grounded in retrieved content.

Several downstream applications in RAG-based QA systems implicitly rely on the model's ability to assess faithfulness. For example, Wang et al. (2024a); Asai et al. (2023) propose an adaptive RAG system that decides whether the retrieved content should be used in generation, based on a faithfulness assessment. Ye et al. (2024); Wei et al. (2024) provide a chain of thought (CoT) technique that implicitly evaluates the relevance of individual passages and filters the irrelevant ones before generating an answer. Yoran et al. (2024); Jin et al. (2024) highlight that insufficient/irrelevant context can cause the LLM to err, even if the answer is in its parametric memory. In these examples and others, the subcomponent of evaluating faithfulness or context sufficiency is chosen arbitrarily, limiting both transparency and control. By defining a unified framework, one can allow future works to choose the most suitable subcomponent implementation, thereby achieving superior results in their respective task.

To address this gap, we formally define the task of Answering with Faithfulness (AwF): given a user question and retrieved passages, the model generates both an answer and a faithfulness prediction indicating whether the answer is grounded in the passages. Rather than proposing a new application or method, we present a conceptual and evaluative framework centered on this task to enable explicit control over when to trust a generated response. To the best of our knowledge, this is the first work to define AwF as a distinct task, along with introducing tailored evaluation metrics of *AwF precision* and *AwF recall*, to support systematic comparison and analysis of different AwF methods. To illustrate the value of this comparison, we demonstrate

Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, pages 1017–1037
December 20-24, 2025 ©2025 Association for Computational Linguistics

that our metrics remain consistent across diverse benchmarks and language models. Moreover, using the AwF framework and metrics, we show both theoretically and empirically that improvements in AwF directly enhance answering quality across RAG-based downstream applications.

Summarizing, our contributions are as follows: (i) We define the AwF task, allowing explicit tuning of faithfulness predictions, and provide tailored precision–recall metrics within a unified framework that supports direct comparison across AwF methods (Section 3). (ii) We show that our formulation unifies methods originally developed for other tasks (Section 4). (iii) We conduct a comprehensive study across models and benchmarks, revealing consistent performance trends of AwF methods (Section 5). (iv) We demonstrate both theoretically and empirically how AwF improvements lead to better performance in downstream applications with different strategies for handling unfaithful answers (Section 6).

2 Related Work

Evaluating Context Relevance Faithfulness is closely related to the problem of evaluating whether the retrieved context is relevant or sufficient for an adequate answer. There are several approaches towards assessing the quality of retrieved context. Thakur et al. (2024) provide a dataset (NoMiracl) of queries and related passages along with labels for answerability of queries. Using this dataset, they show that LLMs perform poorly in identifying these unanswerable cases. Wang et al. (2024a) propose an adaptive RAG system that decides whether retrieved content should be used in the generation phase. Ye et al. (2024) and Wei et al. (2024) fine-tune an LLM to generate a response using CoT, where it first decides which passages are useful, then generates a response. Meng et al. (2024) propose using LLM-generated binary relevance labels that are subsequently used to compute continuous scores assessing the quality of the retrieved content in terms of a desired retrieval metric, such as the precision-oriented reciprocal rank, or the recall-oriented NDCG. Joren et al. (2025) take a different perspective on retrieval quality assessment by exploring several ways of determining whether the context is sufficient to answer a question. They apply their approach to design a Q&A system that can abstain and is measured by its abstention rate and answer accuracy on the non-abstained questions. Finally, some papers (Yoran et al., 2024; Jin et al., 2024) have an implicit approach to the problem, where rather than letting the LLM or another model decide whether retrieved content is useful, they fine-tune the LLM to be robust to irrelevant data. We note that while our methods aim to evaluate faithfulness, a key insight we present is that for many downstream tasks, when replacing the retrieval quality estimations listed here with faithfulness prediction, the overall quality improves.

Evaluating Faithfulness Here, the challenge is to decide whether a given response has sufficiently high quality given the retrieved content, or phrased inversely, whether the content was used in generating the response. A natural way of doing so is to determine whether the answer is implied by the retrieved content. This challenge is closely related to fact-checking (Wang et al., 2024b), where NLI is a popular approach for verifying a statement given evidence (see Honovich et al. (2022) and references within). A computationally expensive alternative to standard NLI models is represented by RAGAS faithfulness (Es et al., 2024), based on prompting a powerful LLM. While the above treat faithfulness as an evaluation task, our work treats it as a prediction task tightly coupled to answer generation.

Wu et al. (2024) studied the inclination of RAG models to prefer their parametric memory over the provided context, and vice versa. They provide a test for faithfulness in which they compare the perplexity of an answer generated by an LLM with and without retrieved content. (Asai et al., 2023) provide a RAG framework where, among other things, they generate sentences in parallel with and without retrieval and choose an output based on a (self) evaluation of faithfulness. While these works do couple faithfulness assessment with answer generation, they do not formulate it as a distinct prediction task with dedicated evaluation metrics. In contrast, our work explicitly defines and evaluates this task, and systematically studies its behavior across language models, datasets, and its impact on downstream system performance.

Uncertainty Estimation. A somewhat related body of work involves estimating uncertainty or confidence (Geng et al., 2024). From a high-level perspective, both AwF and uncertainty estimation attempt to estimate answer quality. However, the definition of that quality is quite different; in AwF, it relates to retrieved content, while in uncertainty estimation, it relates to the parametric memory of the LLM. Due to this significant difference, we do not consider such methods as AwF methods.

3 Task Definition & Metrics

We provide a formal definition of the AwF task and then show how a line of methods fits into this framework. The input to the AwF task consists of a question q, and a collection of passages P, typically obtained via retrieval. Our goal is building an *AwF method* M that computes

$$M(q, P) = (a, v),$$

where a is the generated answer, and $v \in \{0, 1\}$ is its predicted faithfulness indicator. The faithfulness indicator aims to predict the *true* faithfulness of an answer given the passages:

$$V_{q,P}(a) = \begin{cases} 1 & P \text{ supports the statement:} \\ & \text{“the answer to } q \text{ is } a\text{”}, \\ 0 & \text{otherwise.} \end{cases}$$

$V_{q,P}(a)$ can be estimated by human annotators, or a judge LLM (Chiang and Lee, 2023; Zheng et al., 2023; Es et al., 2024).

The predictions a and v are highly related, and their quality should be evaluated as a whole. In particular, the metrics measuring the performance of M should capture the fact that when $v = 0$, the quality of a is not important. Indeed, one can think of making use of v as a gating mechanism to invoke a different generation process when $v = 0$, thereby ignoring a in this case. Similarly, when M fails to produce a faithful answer, v should be 0 even if a supported answer can be generated from the passages. Moreover, note that the cost of providing a wrong answer vs. the cost of not providing an answer when a proper answer can be inferred from the passages, depends on the specific use case. Thus, we want to maximize two competing objectives that capture this tradeoff. To that end, we introduce a tailored notion of precision and recall, defined below.

Assume we are given a set of question and passage pairs $\{(q_i, P_i)\}_{i=1}^{N}$, and M is used to append to each such pair its predictions a_i, v_i. We define our metrics w.r.t. to the set of tuples $\{(q_i, P_i, a_i, v_i)\}_{i=1}^{N}$. Throughout, all sums are over these N tuples, and we denote their corresponding ground truth labels as $V_i = V_{q_i,P_i}(a_i)$.

AwF Precision is similar to the standard classification precision, the fraction of answers the generator correctly deemed faithful, out of the total number of faithful answers. The number of correctly classified faithful answers (True Positives) is $\text{True-Pos} = \sum_i v_i \cdot V_i$, and the total number of answers that were classified as faithful (Predicted Positive) is $\text{Pred-Pos} = \sum_i v_i$. The *answering with faithfulness precision* is therefore

$$\text{AwF-Precision} = \frac{\text{True-Pos}}{\text{Pred-Pos}}$$

We note that even though the precision appears identical to the standard classifier precision at first glance, it also depends on the generated answers as well, since the ground truth label V_i depends on the answer a_i.

AwF Recall is the fraction of answers correctly deemed faithful, out of the total number of questions for which a faithful answer exists. Adopting the terminology coined by Joren et al. (2025), these are are exactly the questions for which we have a *sufficient context*. Formally, let

$$V_{q,P}^{\text{SfC}} = \begin{cases} 1 & \exists a^* \text{ s.t. } V_{q,P}(a^*) = 1, \\ 0 & \text{otherwise,} \end{cases}$$

and $V_i^{\text{suff}} \equiv V_{q_i,P_i}^{\text{suff}}$. The number of answerable questions is $\text{F-Answerable} = \sum_i V_i^{\text{SfC}}$, and the recall is defined by:

$$\text{AwF-Recall} = \frac{\text{True-Pos}}{\text{F-Answerable}}$$

A connection to the classical notion of classifier recall can be obtained from a simple reformulation. Denoting by Faithful the number of faithful generated answers, $\text{Faithful} = \sum_i V_i$, the recall can be reformulated as

$$\text{AwF-Recall} = \underbrace{\frac{\text{True-Pos}}{\text{Faithful}}}_{\text{classifier recall}} \cdot \underbrace{\frac{\text{Faithful}}{\text{F-Answerable}}}_{\text{answering recall}} \quad (1)$$

Thus, our notion of recall is the classifier recall given the answers, multiplied by the ability of the generator to produce faithful answers whenever a faithful answer exists.

Connection to Context Sufficiency We note that AwF is similar to post-retrieval *query performance prediction* (QPP) (Arabzadeh et al., 2024) and content sufficiency (Joren et al., 2025) (SfC for brevity), with the distinction that the predicted indicator V evaluates whether P supports a correct answer a^*_q, rather than the generated answer a. To measure the quality of such a prediction, we define SfC-Precision and SfC-Recall to be the same as above when replacing V_i with V_i^{SfC} in the TRUE-POS formula. We note that recall becomes the standard recall metric since the Faithful and F-Answerable sets become the same (Equation (1)). Due to the similarity of the methods, techniques designed for AwF can be evaluated for SfC.

4 Methods

We consider various methods that fit within the AwF framework, demonstrating how our formulation unifies approaches originally designed for different problems, such as answer generation. In some cases, we make slight adaptations to align these approaches with AwF (e.g., pairing answer generation with a simple faithfulness prediction that always sets $v = 1$). Some of the methods we consider provide a hard classification result, i.e., $v \in \{0, 1\}$, whereas others provide a continuous decision function that can be thresholded to obtain $v \in \{0, 1\}$. We first present *unified* methods that simultaneously output both an answer and its faithfulness indicator. Then, we provide *composed* methods that combine answering modules with faithfulness prediction ones. The exact LLM prompts we used in the following methods are available in Appendix C.

4.1 Unified Methods

Intrinsic Abstention. A straightforward technique where we prompt an LLM to answer only if the answer appears in the context and reply with "DONT KNOW" when it does not. We set $v = 1$ if and only if the answer is not "DONT KNOW".

CoT few-shot Hybrid. A variant of the Intrinsic Abstention method using both chain-of-thought and few-shot examples. It is inspired by the method described in (Wei et al., 2024), where the LLM is instructed to reason about the relevance of the passages before answering and is given two examples comprising a question, passages, and the reasoning. We adapt the original method by prompting the LLM to answer "DONT KNOW" if an answer cannot be deduced from the passages ($v = 0$).

Dual Generation. A method proposed by Wu et al. (2024). The idea is to generate an answer both with and without P, then compare the (normalized) perplexity percentiles of both answers in order to choose one. We define a continuous decision function for v as the difference between the perplexities.

4.2 Composed Methods

We consider methods that compose two components for producing the AwF output (a, v): an answer generation method to generate a, and a faithfulness prediction method to produce v. Below, we describe concrete answer generation and faithfulness prediction methods we consider in this paper.

4.2.1 Answer Generation

Vanilla. The straightforward approach for answering questions. Here, we instruct the LLM to answer the question given the passages.

InstructRAG. This is a variant of the Vanilla method using both chain-of-thought and few-shot examples proposed by Wei et al. (2024). We slightly modified the in-context examples and instructions to enable a structured response, from which we can extract only the final answer.

4.2.2 Faithfulness Prediction

Trivial. A simple baseline that always predicts $v = 1$, meaning that it believes the answer from the generation method is always faithful.

Pre-Answering Prediction. A method originally designed for SfC. Given q, P, we ask the LLM to evaluate whether P contains an answer to q. We ask for a single yes/no answer given all the passages and obtain a continuous decision function for v by inspecting the logits of the generated response. We use the prompt given in (Thakur et al., 2024).

Post-Answering NLI. A faithfulness prediction method mimicking $V_{q,P}(a)$. Here, we first invoke one of the answering methods described above to generate the response a, then use the question, passages, and the generated answer to decide whether the question-answer pair is faithful to the passages. For small/medium scale LLMs ($< 10B$) we use a dedicated NLI model based on DeBERTa Laurer et al. (2024) ($< 1B$ parameters). For the larger-scale models, we prompt the tested LLM to provide a binary classification and use the logits for a continuous decision function. Further details are available in Appendix A.

5 Empirical Comparison of AwF Methods

We conducted a series of experiments to assess the performance of various AwF methods under multiple settings and to examine the consistency of their results. Specifically, we compare each method's precision and recall; for methods that output a continuous decision score, we compare the full trade-off curve obtained by varying the decision threshold.

Our objective is to understand how the relative performance of the AwF methods behaves when we vary benchmarks, language models, and even the task itself (between AwF and SfC). We first detail the experimental setup and then present results that highlight this cross-setting consistency.

5.1 Experimental Setup

For our experiments, we use question-answering benchmarks where each entry consists of a question, one or more retrieved passages, a reference answer, and a binary relevance label indicating whether the reference answer can be inferred from the passages. We focus on single-hop factoid questions, where the answer is fully contained within a single passage. To compute precision and recall, as defined in Section 3, we estimate $V_{q,P}(a)$ as follows. We consider $V_{q,P}(a)$ to be 1 if: (1) a is equivalent to the reference answer, as judged by a strong language model (Claude 3.5 Sonnet), and (2) the reference answer is supported by the passages

Note that this setup indeed estimates V: Since q is a factoid question, it has a single correct answer. Any answer that is not equivalent to the reference answer is therefore incorrect and cannot be supported by the passages (i.e., $V_{q,P}(a) = 0$). Conversely, if a is equivalent to the reference, then it is supported by the passages (i.e., $V_{q,P}(a) = 1$) if and only if the reference itself supported by them.

We evaluate our methods on three public benchmarks: NQ (Kwiatkowski et al., 2019), NoMIRACL (Thakur et al., 2024), and BioASQ (Krithara et al., 2023). NQ is a widely used QA dataset consisting of real-user queries with answers retrieved from Wikipedia. NoMIRACL is a benchmark based on real-user queries and includes annotations indicating whether each context is sufficient or insufficient, which are used to assess whether LLMs have the ability to abstain when retrieval fails. To increase topical and corpus diversity, we also include BioASQ which focuses on biomedical questions from PubMed abstracts. Table 1 provides benchmark statistics; further details on the benchmarks collection and pre-processing are provided in Appendix B, and the resulting benchmark files are available online[1].

Benchmark	size (#QAs)	% of answerable questions	avg passages per question
NQ	5K	82%	5
NoMIRACL	3.2K	81%	10.1
BioASQ	2.9K	50%	6.5

Table 1: Benchmarks Statistics.

For each benchmark, we test the unified and composed methods for the AwF task, as presented in Section 4. For the composed methods, we test all combinations of answer generation and faithfulness prediction methods. Since AwF methods rely on instruction-tuned generative models, we conduct experiments using Llama 3 Instruct (3B, 8B, 70B), Falcon 3 Instruct (3B, 10B), and Qwen 2.5 Instruct (72B). Models are referred to by their first letter and size, e.g., F10B.

5.2 Results

For each AwF method, LLM, and dataset, we compute the AwF precision, AwF recall, and their F1 score. For the methods outputting a continuous score (e.g., Post-Answering NLI), we evaluate their F1 across all thresholds and report the max value. Table 2 presents the average F1 score obtained by each of the methods over our three benchmarks. When using names of answer generation methods, we implicitly refer to those methods composed with the Trivial faithfulness prediction method. Elaborated tables including all benchmarks of both F1 scores and area under the curve (AUC), together with a 95% confidence interval appear in Appendix D.1 (Tables 6 and 7). To provide a more meaningful view of the precision-recall tradeoff, we present a representative plot of Falcon-3B in Figure 1. Plots for the other LLMs are available in Appendix D.2 (Figure 6).

One evident trend is that chain-of-thought improves performance: InstructRAG consistently outperforms Vanilla (as it is a variant of Vanilla with CoT), and CoT few-shot Hybrid outperforms Intrinsic Abstention. Notably, this advantage persists with composition methods. Namely, when one answering method outperforms another (InstructRAG consistently outperforms Vanilla), this ordering remains unchanged after their composition with any

[1] https://github.com/alexshtf/awf_datasets

Method	F3B	F10B	L3B	L8B	L70B	Q72B
Intrinsic	52.9	61.3	29.3	55.1	69.8	71.8
Trivial Vanilla	56.4	63.8	37.0	58.5	65.8	67.7
CoT	60.6	66.8	54.7	63.8	69.4	72.3
Trivial InstRAG	61.8	66.2	56.7	61.9	66.4	67.7
Pre-Ans Vanilla	56.5	65.2	37.5	59.4	69.2	68.4
Pre-Ans InstRAG	61.8	67.4	56.9	62.8	70.2	68.3
NLI Vanilla	59.6	66.1	41.6	61.4	69.8	71.4
NLI InstRAG	63.7	67.6	59.0	64.5	70.8	72.5
Dual Gen	56.4	63.7	37.3	58.5	65.8	67.8

Table 2: Average F1, on the scale 0-100, defined by the harmonic mean of the average precision and recall over the datasets of every method and model. The results of each dataset appear in Appendix D.1.

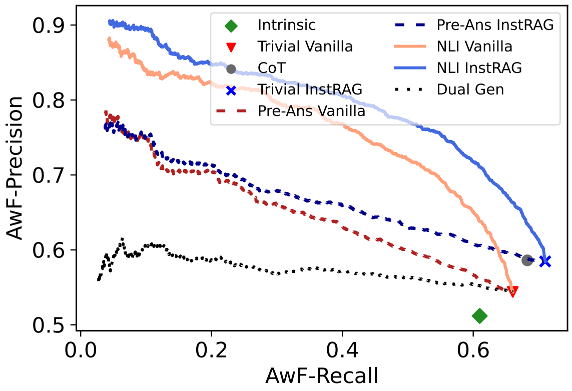


Figure 1: AwF-Precision and AwF-Recall of AwF methods using F3B on NQ benchmark.

faithfulness prediction method. We note that for most cases (LLMs, benchmarks), the curve of one fully dominates the other (a stronger statement than having a better F1 score), but there are a handful of exceptions.

Another, less obvious trend is a clear hierarchy between faithfulness prediction methods. Across the board, the Post-Answering NLI methods outperform the Pre-Answering Prediction counterparts. Here we note that beyond a better F1 score, Figure 6 shows a full dominance of the precision-recall curve. This reinforces the intuition that considering the generated answer improves faithfulness prediction. Dual Generation can be viewed as an exception as it does make use of the answer, but its performance shows that it is less suited for the AwF task.

A final observation relates to the behavior as a function of the LLM size. Here, we see a difference between the small/medium ($<10B$) and large ($70B$) scale LLMs, in that the differences become smaller. For the large models, a simple method such as Intrinsic Abstention performs quite well, and achieves closer performance to the leading method of Post-Answering NLI coupled with InstructRAG. This being said, we note that (1) composed methods still provide a superior F1 score ($\sim$1% gap for both Llama and Qwen), and more importantly, (2) Intrinsic Abstention lacks a decision function, and produces fixed precision-recall values. Thus, in scenarios requiring explicit control over the precision or recall (e.g., medical queries requiring high precision), alternative methods are required.

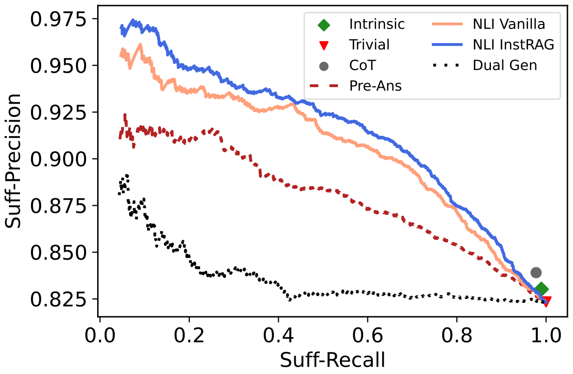


Figure 2: SfC-Precision and SfC-Recall of AwF methods using F3B on NQ benchmark

Relation to Sufficient Context Consider SfC-Precision and SfC-Recall as defined in Section 3. We present the evaluation of those metrics using a representative example of a medium-scale LLM (F3B) in Figure 2[2] (a full visual description appears in Figure 7 of Appendix D.2). Recall that Pre-Answering Prediction is designed to predict the SfC objective, whereas Post-Answering NLI is designed for the AwF objective. Nevertheless, the same trends as before remain, in particular the superior performance of Post-Answering NLI. This is somewhat surprising and could bring insights into future solutions for the SfC problem.

6 Applications

We consider downstream applications of AwF, each using different corrective actions to handle cases where the generated answer is predicted to be unfaithful ($v = 0$). We will provide a formal analysis of this setting providing theoretical

[2] Since SfC-Precision and SfC-Recall are independent of the generated answer, all methods that estimate faithfulness without considering the answer produce identical results. In particular, this applies to all methods based on Trivial (which always predicts $v = 1$) and on Pre-Answering Prediction (where faithfulness prediction is performed before answer generation). The respective composed methods are referred shortly as Trivial and Pre-Answering Prediction in Figure 2.

justification for why improvements in AwF lead to enhanced performance in downstream applications (Theorem 1). We then provide empirical evidence that employing AwF methods with better AwF-Precision/AwF-Recall trade-offs leads to improved system performance.

6.1 Formal analysis

Due to space restrictions, we provide a short informal analysis here and defer the full rigorous analysis to Appendix F. To model the different fallback methods, we denote by M the AwF method that provides both an answer and a faithfulness score. We denote by F a fallback mechanism that given a query provides an alternative response to M. This can be a constant "Don't know" , a no-RAG response, the output of an expensive LLM or something else. For each query both the answers of M and of F have an associated utility.

The key assumption in our analysis is that the utility of F is independent of v, the faithfulness score of M; namely, that its distribution (over queries) given $v = 0$ and $v = 1$ are the same. While this is not always the case, the empirical study shows that the dependence is weak enough for the end conclusion to hold.

Theorem 1 (Informal)**.** *Let M_1 and M_2 be two AwF methods and F a fallback mechanism independent of both. Assume that M_1 outperforms M_2 with respect to both* Precision *and* Recall*. Then the utility of the combination of F with M_1 is greater than that of F and M_2.*

6.2 Experiments

6.2.1 No-RAG Fallback

In this application, we couple a RAG system with a fallback mechanism that, whenever $v = 0$, replaces the answer with one generated by prompting the LLM to respond based on its parametric memory, that is, without access to retrieved content. This fallback mechanism is utilized among other places, in (Asai et al., 2023), motivated by LLMs' tendency to get distracted from irrelevant passages (Amiraz et al., 2025).

Figure 3 illustrates the No-RAG strategy for Llama3B on BioASQ questions, when using the composition of InstructRAG and Post-Answering NLI. The figure presents overall answer accuracy (i.e., the percentage of generated answers that match the reference) as a function of the fallback rate, which can be controlled via different thresholding of the soft score Post-Answering NLI generates for v. Incorporating No-RAG fallback improves accuracy over InstructRAG for fallback rates up to 70%, peaking around 50% rate before declining. These results are not surprising, since 50% of BioASQ questions are not answerable from the passages (i.e., having insufficient context); this is a demonstration of AwF ability to detect those cases. This can be further explained by comparing InstructRAG and No-RAG answers when $v = 0$ (w.r.t. InstructRAG answer): in low fallback rates, No-RAG outperforms InstructRAG, so replacing the answers enhances overall accuracy. However, as fallback increases, the accuracy gap between the two narrows, and beyond 70%, InstructRAG surpasses No-RAG, making further fallback detrimental.

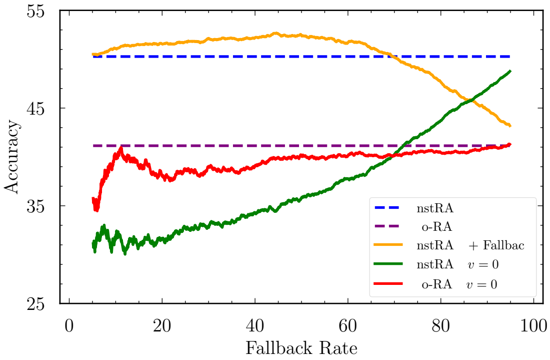


Figure 3: No-RAG fallback. Accuracies of different types of answers as a function of the fallback rate. Orange: InstructRAG accuracy with No-RAG fallback when NLI predicts $v = 0$. Green: Avg. accuracy of InstructRAG answers predicted as unfaithful. Red: Avg. accuracy of No-RAG answers for unfaithful InstructRAG cases.

In Table 3, we compare Pre-Answering Prediction and Post-Answering NLI (both composed with InstructRAG) for this application[3]. We present here results only for BioASQ, since for NQ and NoMIRACL, we observe little to no improvement in overall system accuracy for most LLMs, likely due to them having mostly (∼82%) questions with relevant passages (See Table 8 for results across all benchmarks). In BioASQ, however, only 50% of the questions contain relevant context, and the overall improvement is significant for most LLMs.

We see that the Post-Answering NLI techniques clearly outperform the Pre-Answering Prediction

[3] For each evaluated benchmark and model, we use 5-fold cross-validation, optimizing the threshold on four folds and evaluating performance on the fifth.

LLM	Pre-Ans	NLI
Q72B	$4.26_{\pm 0.69}\%$	$8.07_{\pm 1.40}\%$
L70B	$7.39_{\pm 1.99}\%$	$10.36_{\pm 0.65}\%$
L8B	$1.33_{\pm 0.86}\%$	$3.13_{\pm 1.66}\%$
L3B	$-0.10_{\pm 0.12}\%$	$2.32_{\pm 0.88}\%$
F10B	$0.51_{\pm 0.63}\%$	$0.20_{\pm 0.80}\%$
F3B	$0.03_{\pm 0.06}\%$	$0.07_{\pm 0.12}\%$

Table 3: Accuracy improvement with No-RAG fallback over InstructRAG answers, using Pre-Answering Prediction or Post-Answering NLI for faithfulness prediction on BioASQ. Values in subscript represent 95% confidence intervals.

technique when evaluated on the downstream task, fully matching their advantage when tested with AwF-Precision/AwF-Recall. In Appendix E.1 we show that this same trend persists across the other AwF methods.

6.2.2 Switching to a Larger Model

This strategy matches a scenario where the RAG system primarily uses a small and cheap LLM, but when $v = 0$, it switches to a larger, more expensive model. The system balances two competing objectives: (i) quality, measured by accuracy, and (ii) cost, measured by switch rate, i.e., the proportion of answers replaced by the larger model. Figure 4 illustrates the trade-off between accuracy and switch rate for Falcon3B and Llama70B on the NQ benchmark. The ranking of the faithfulness methods from Section 5.2 remains consistent, showing that better AwF-Precision/AwF-Recall curves lead to a more favorable trade-off. Note that the set of input-independent baselines that switch the answer randomly according to some fixed probability form a linear curve, similar to the behavior exhibited by Dual Generation.

This same trend persists across the other benchmarks and LLM choices (full results can be found in Appendix E.2).

6.2.3 Selective Accuracy

Joren et al. (2025) present another example of implicitly using AwF. They study the problem of answering with abstention using two objectives: coverage — the fraction of answered questions, and selective accuracy — the answer accuracy on questions for which the model did not abstain. They examine continuous decision functions, resulting in a tradeoff curve between the two objectives.

The paper provides a baseline method called P[True] (Kadavath et al., 2022), defined as the probability of the LLM to answer "True" when asked whether the answer is correct. They show that combining P[True] with a SfC score, indicating whether the context is sufficient to answer the question, yields a better tradeoff between coverage and selective accuracy. This combination is motivated by cases in which the LLM is confident in its answer despite lacking support from the retrieved context. In such cases, the LLM should be allowed to answer using its parametric memory. In the paper, they train a logistic regression model that maps the two predictions to a single score. To estimate SfC, they use a prompt similar to our Pre-Answering Prediction.

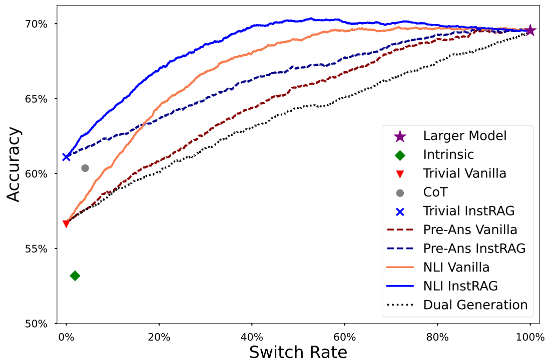


Figure 4: Switching to a larger model. Accuracy vs Switch Rate, when using InstructRAG and replacing F3B answers with L70B for cases where $v = 0$ on NQ.

Building on our finding that SfC can be framed as AwF, and that AwF can be more effectively solved using NLI, we propose replacing their SfC score based on Pre-Answering Prediction with a AwF method based on Post-Answering NLI. In Table 4, we compare their original baseline (P[True]) and method (Suff+P[True]) with our proposed approach (AwF+P[True]). We evaluated the methods on the NQ and BioASQ datasets[4], using answers generated by the Vanilla method (§4.2) across multiple LLMs. We report the area under the curve between coverage (x-axis) and selective accuracy (y-axis). Figure 5 shows the full curve for a representative case using Llama-3-8B on the NQ dataset. Full implementation details and additional results, including those for NQ and the InstructRAG answering method, are provided in Appendix E.3.

Our experiments replicate the findings of Joren et al. (2025), showing that combining SfC with P[True] consistently matches or outperforms

[4] The NoMIRACL benchmark lacks ground-truth answers for questions without sufficient context, making it invalid for this experiment.

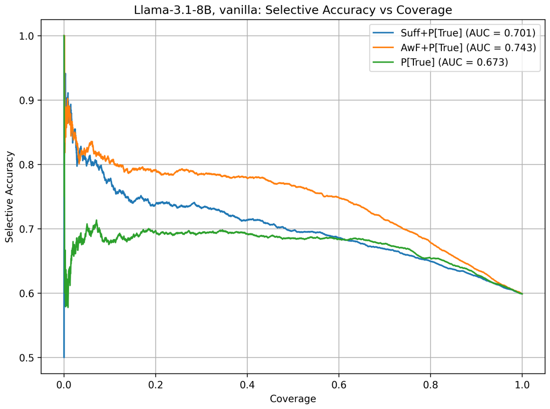


Figure 5: Selective accuracy vs. Coverage using L8B on NQ benchmark.

Dataset	Decision Function	F3B	F10B	L3B	L8B	L70B
NQ	P[True]	69.7	76.7	68.3	67.3	74.8
	Suff+P[True]	69.4	76.5	67.3	70.1	78.9
	AwF+P[True] (ours)	**74.3**	**77.5**	**72.9**	**74.3**	**79.6**
BioAsq	P[True]	63.6	73.2	25.5	58.8	71.5
	Suff+P[True]	63.1	70.9	22.7	60.7	74.9
	AwF+P[True] (ours)	**67.6**	**74.0**	**31.3**	**66.4**	**76.5**

Table 4: Area under the curve for different answering methods, over the NQ and BioAsq dataset

P[True] alone. We also draw a new conclusion: by formulating SfC as AwF and selecting a better AwF method, the results can be further and significantly improved. This is reflected in the superior performance of the AwF+P(True) method across all evaluations (Tables 4, 11).

7 Discussion

Our work introduces the Answering with Faithfulness problem along with tailored precision and recall metrics, providing a unified framework for evaluation. By making faithfulness prediction an explicit output, we generalize diverse prior approaches that implicitly address answer faithfulness, enabling direct comparisons across methods.

Comparing different AwF methods across diverse settings shows that the same trends and conclusions hold across benchmarks, language models, and even between tasks (AwF and SfC). The consistency of AwF enables us to draw broad conclusions—for example, solutions based on Post-Answering NLI consistently outperform those using Pre-Answering Prediction. In addition, we show both theoretically and empirically, that our AwF formulation is useful: improving AwF metrics leads to better performance of downstream tasks.

Finally, we demonstrate that applying these insights to tasks that implicitly rely on AwF (such as selective accuracy) by improving their existing AwF methods, result in superior performance sometimes surpassing the current state of the art. A promising direction for future work is to explore more sophisticated AwF methods, which could further enhance downstream performance.

8 Limitations

The AwF problem applies to any benchmark where RAG provides a suitable solution. In this study, we focused on question-answering benchmarks, specifically those with factoid questions. We focused our attention on these benchmarks since other types would admit additional technical challenges that are outside the scope of our study, making it difficult to understand the core problem and the analysis of our results. For example, with long-form answers, faithfulness ceases to become a binary score since an answer can be partially supported by the documents. An additional limitation to our study is the language: we restricted our focus to English benchmarks and corpora and left the analysis of additional languages to future work.

Finally, our focus was on methods that do not require fine-tuning an LLM. This choice is due to two reasons. (1) The popularity of such choices in real settings, as it is much more convenient to use an off-the-shelf LLM as opposed to fine-tuning one. (2) The added technical challenges related to such methods, such as searching for the right hyperparameters for training, the cost of training, and the complexity related to in-distribution vs out-of-distribution performance.

References

Chen Amiraz, Florin Cuconasu, Simone Filice, and Zohar Karnin. 2025. The distracting effect: Understanding irrelevant passages in RAG. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 18228–18258, Vienna, Austria. Association for Computational Linguistics.

Negar Arabzadeh, Chuan Meng, Mohammad Aliannejadi, and Ebrahim Bagheri. 2024. Query performance prediction: From fundamentals to advanced techniques. In *European Conference on Information Retrieval*, pages 381–388. Springer.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2023. Self-rag: Learning to retrieve, generate, and critique through self-reflection. In *The Twelfth International Conference on Learning Representations*.

Cheng-Han Chiang and Hung-yi Lee. 2023. Can large language models be an alternative to human evaluations? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15607–15631, Toronto, Canada. Association for Computational Linguistics.

Shahul Es, Jithin James, Luis Espinosa Anke, and Steven Schockaert. 2024. Ragas: Automated evaluation of retrieval augmented generation. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 150–158.

Jiahui Geng, Fengyu Cai, Yuxia Wang, Heinz Koeppl, Preslav Nakov, and Iryna Gurevych. 2024. A survey of confidence estimation and calibration in large language models. 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 6577–6595.

Or Honovich, Roee Aharoni, Jonathan Herzig, Hagai Taitelbaum, Doron Kukliansy, Vered Cohen, Thomas Scialom, Idan Szpektor, Avinatan Hassidim, and Yossi Matias. 2022. True: Re-evaluating factual consistency evaluation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3905–3920.

Bowen Jin, Jinsung Yoon, Jiawei Han, and Sercan O Arik. 2024. Long-context llms meet rag: Overcoming challenges for long inputs in rag. *arXiv preprint arXiv:2410.05983*.

Hailey Joren, Jianyi Zhang, Chun-Sung Ferng, Da-Cheng Juan, Ankur Taly, and Cyrus Rashtchian. 2025. Sufficient context: A new lens on retrieval augmented generation systems. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. 2022. Language models (mostly) know what they know. *CoRR*.

Anastasia Krithara, Anastasios Nentidis, Konstantinos Bougiatiotis, and Georgios Paliouras. 2023. Bioasq-qa: A manually curated corpus for biomedical question answering. *Scientific Data*, 10(1):170.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

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

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc.

Chuan Meng, Negar Arabzadeh, Arian Askari, Mohammad Aliannejadi, and Maarten de Rijke. 2024. Query performance prediction using relevance judgments generated by large language models. *arXiv preprint arXiv:2404.01012*.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Richard James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. 2024. REPLUG: Retrieval-augmented black-box language models. 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 8371–8384, Mexico City, Mexico. Association for Computational Linguistics.

Nandan Thakur, Luiz Bonifacio, Crystina Zhang, Odunayo Ogundepo, Ehsan Kamalloo, David Alfonso Hermelo, Xiaoguang Li, Qun Liu, Boxing Chen, Mehdi Rezagholizadeh, et al. 2024. “knowing when you don’t know”: A multilingual relevance assessment dataset for robust retrieval-augmented generation. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 12508–12526.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2022. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*.

Xi Wang, Procheta Sen, Ruizhe Li, and Emine Yilmaz. 2024a. Adaptive retrieval-augmented generation for conversational systems. *arXiv preprint arXiv:2407.21712*.

Yuxia Wang, Revanth Gangi Reddy, Zain Mujahid, Arnav Arora, Aleksandr Rubashevskii, Jiahui Geng, Osama Mohammed Afzal, Liangming Pan, Nadav Borenstein, Aditya Pillai, et al. 2024b. Factcheck-bench: Fine-grained evaluation benchmark for automatic fact-checkers. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 14199–14230.

Zhepei Wei, Wei-Lin Chen, and Yu Meng. 2024. Instructrag: Instructing retrieval augmented generation via self-synthesized rationales. In *Adaptive Foundation Models: Evolving AI for Personalized and Efficient Learning*.

Kevin Wu, Eric Wu, and James Zou. 2024. Clasheval: Quantifying the tug-of-war between an LLM's internal prior and external evidence. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Guangzhi Xiong, Qiao Jin, Zhiyong Lu, and Aidong Zhang. 2024. Benchmarking retrieval-augmented generation for medicine. *arXiv preprint arXiv:2402.13178*.

Xi Ye, Ruoxi Sun, Sercan Arik, and Tomas Pfister. 2024. Effective large language model adaptation for improved grounding and citation generation. 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 6237–6251.

Ori Yoran, Tomer Wolfson, Ori Ram, and Jonathan Berant. 2024. Making retrieval-augmented language models robust to irrelevant context. In *The Twelfth International Conference on Learning Representations*.

Peitian Zhang, Shitao Xiao, Zheng Liu, Zhicheng Dou, and Jian-Yun Nie. 2023. Retrieve anything to augment large language models. *arXiv preprint arXiv:2310.07554*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Advances in Neural Information Processing Systems*, volume 36, pages 46595–46623. Curran Associates, Inc.

A NLI Model

A.1 Implementation details

For using the NLI model to predict whether the answer question-answer pair is faithful to the passages, we created the hypothesis using this template: `The answer to the question "{q}" is: "{a}"`, while each passage serves as an independent premise (in preliminary experiments, we explored rephrasing the question-answer pair into its declarative form using an LLM, but it did not yield an additional advantage). In case the passage and the hypothesis together exceed the context window of the NLI model, we split the passage into chunks with an overlap of 20 words. We then use the maximum score of the NLI model over all premises as the decision function for v.

A.2 Dedicated model selection

We used a subset of 700 questions from NQ with answers generated by Falcon-10B, to compare different NLI models and different ways to format the input (passage, question, answer) to feed the model. Table 5 contains the max F1 scores across all possible thresholds (the same as in Table 2) for each combination of (nli-model, input-format).

The hypothesis formulations we considered are:

Formatted Concatenation

```
Premise: <passage>
-----------------------------------------
Hypothesis: Question: <question>
Answer: <answer>
```

Natural Sentence

```
Premise: <passage>
-----------------------------------------
Hypothesis: The answer to the question: <question> is:
<answer>
```

Zero-Shot Classification

```
Premise: ### Context
<passage>

### Question
<question>

### Answer
<answer>
-----------------------------------------
Hypothesis: The answer is supported by the context
```

For NLI model, we considered four bert-based models, each with fewer than 1 billion parameters:

- MoritzLaurer/DeBERTa-v3-large-mnli-fever-anli-ling-wanli
- MoritzLaurer/deberta-v3-large-zeroshot-v2.0
- MoritzLaurer/ModernBERT-large-zeroshot-v2.0
- MoritzLaurer/bge-m3-zeroshot-v2.0

Additionally, we tested TRUE (Honovich et al., 2022), a T5-XXL-based model with 7 billion parameters.

As shown in the table, the top two models are *MoritzLaurer/deberta-v3-large-zeroshot-v2.0* (with the *Natural Sentence* format) and TRUE (with the *Natural Sentence* and *Formatted Concatenation* formats). Since their results are comparable, we chose to use the DeBERTa-based model in our experiments because of its smaller size (fewer than 1B parameters vs. 7B parameters for TRUE), and to formulate the hypothesis as *Natural Sentence*, due to its optimal fit with our chosen model. We also conducted preliminary experiments with RAGAS faithfulness (Es et al., 2024), using Claude 3.5 Sonnet. However, the observed improvements over the DeBERTa-based model were negligible, and we determined that the additional computational cost of a larger model was not justified. For the 70B LLMs,

NLI Model	Formatted Concatenation	Natural Sentence	Zero-shot Classification
DeBERTa-NLI	0.468	0.451	0.423
DeBERTa-zero-shot-classification	0.459	**0.471**	0.431
ModernBERT-zero-shot-classification	0.453	0.444	0.438
bge-m3-zero-shot-classification	0.441	0.434	0.428
TRUE	**0.472**	**0.471**	0.421

Table 5: Performance of different NLI models under various input formulations.

we used the same LLM with a prompt as the NLI model, since it performs better than DeBERTa.

A.3 Prompt-based NLI

When the answers are generated by large sized models, we use the generating model for the NLI task. The prompt has the following structure:

NLI prompt

```
system: You are an expert in Natural Language Inference. Given
a premise and a hypothesis, determine if the premise entails
the hypothesis. Output 1 if entailed, and 0 if not entailed.
------------------------------------------------------------
user: Determine if the premise entails the hypothesis. Output
1 for entailment and 0 for non-entailment.
Premise: <Premise>
Hypothesis: <Hypothesis>
Output:
```

Instead of asking the model to generate, we use the log-probability of the answer "1" as the score for the NLI task.

B Benchmarks

- **NQ** (Kwiatkowski et al., 2019) is a general knowledge question answering benchmark based on queries of real users. The dataset consists of questions and ground truth answers. Specifically, we sampled, uniformly at random, 5K question-answer pairs. For each question, we retrieved 5 passages from Wikipedia, using E5-base-v2 (Wang et al., 2022) dense retrieval. Each passage was then labeled as relevant if it contains the answer as a (normalized) substring, or according to the TRUE NLI(Honovich et al., 2022) [5].

- **NoMIRACL** (Thakur et al., 2024) is a public benchmark testing whether LLMs have the ability to abstain. Each entry contains a question, passages, and relevance labels for the passages. The original dataset does not have a ground truth answer. To obtain one, we prompted Claude 3.5 Sonnet based only on the passages that were annotated as containing the answers. In addition, in the original dataset, the relevant passages are separated from the non-relevant ones. We shuffle relevant and non-relevant passages together in a random order. We consider only the English part of this dataset, as all language and NLI models we employed, support this language.

- **BioASQ** (Krithara et al., 2023) is a manually generated question-answer dataset based on abstracts of biological academic papers available in the Pubmed corpus (we used the snapshot published by (Xiong et al., 2024)). We used the BioASQ12 training set, out of which we collected the questions labeled as factoid questions, resulting in a collection of 1.48K entries. Each entry contains a question, a ground truth answer, and a list of relevant passages. To obtain irrelevant passages we used BM-25 to extract the top-10 related passages from PubMed and discard those containing the ground truth answer. Finally, we considered each question twice, using two different passage lists: once with only irrelevant passages and once with the same set, but with one randomly selected irrelevant passage replaced by a randomly chosen relevant one.

The benchmarks above were uploaded

C Method prompts

Below are the prompts to the Vanilla, Intrinsic Abstention, and No Context methods.

Vanilla

```
system: You are a helpful assistant that answers a question
based on the context provided. Please be as concise as possible,
do not add any additional information, and do not refer to the
context in anyway.
------------------------------------------------------------
user: Read the following context carefully and answer the
question below.
Question:
<Question>
Context:
<Passage 1>

<Passage 2>

⋮

<Passage n>
```

[5] A manual inspection showed this strategy to be near perfect in the setting of NQ where the answers are very short and contain only a single fact.

Intrinsic Abstention

```
system: You are a helpful assistant that answers a question based on the context provided. Please be as concise as possible, do not add any additional information, and do not refer to the context in anyway. If the answer does not exist in the context, you should output the special string __DONT_KNOW__ .
----------------------------------------
user: Read the following context carefully and answer the question below only if the answer is supported by the context.
Question:
<Question>
Context:
<Passage 1>

<Passage 2>

.
.
.

<Passage n>
```

No context

```
system: You are a helpful assistant that answers a question based on your knowledge. Please be concise as possible.
----------------------------------------
user: <Question>
```

Below are the prompts of the InstructRAG and CoT few-shot Hybrid methods. We note that each dataset has its own set of example questions and "rationales" for analyzing them. Below is the structure of the prompts.

InstructRAG

```
user: Your task is to analyze the provided documents and answer the given question. Please generate a brief explanation of how the contents of these documents lead to your answer. If the provided information is not helpful in answering the question, you only need to respond based on your own knowledge, without referring to the documents. After your analysis, give the final answer in a self-contained manner after a "Response: " prefix.

Below are some examples of how to answer the question:

###

Example 1

Question: <Example question 1>?

Answer: <Rationale 1>

###

Example 2

Question: <Example question 2>?

Answer: <Rationale 2>

###

Now it is your turn to analyze the following documents and answer the given question.

Document 1: <Passage 1>

Document 2: <Passage 2>

.
.
.

Document 4: <Passage n>

Based on your knowledge and the provided information, answer the question:
<Question>?
```

CoT few-shot Hybrid

```
user: Your task is to analyze the provided documents and answer the given question. Please generate a brief explanation of how the contents of these documents lead to your answer. If the provided information is not helpful in answering the question, you need to respond __DONT_KNOW__. After your analysis, give the final answer in a self-contained manner after a "Response: " prefix.

Below are some examples of how to answer the question:

###

Example 1

Question: <Example question 1>?

Answer: <Rationale 1 with instruction to abstain>

###

Example 2

Question: <Example question 2>?

Answer: <Rationale 1 with instruction to abstain>

###

Now it is your turn to analyze the following documents and answer the given question.

Document 1: <Passage 1>

Document 2: <Passage 2>

.
.
.

Document 4: <Passage n>

Either answer the following question based on the provided information, or reply __DONT_KNOW__:
<Question>?
```

Here is an example of a question and the corresponding rationale for NQ:

```
Question: who won season 13 so you think you can dance?
----------------------------------------
Rationale: After analyzing the provided documents, I found that none of them directly mention the winner of Season 13 of "So You Think You Can Dance". However, I can use my own knowledge to answer the question.

According to various online sources, including Wikipedia and other reputable dance websites, the winner of Season 13 of "So You Think You Can Dance" is indeed Leon "Kida" Burns.

To deduce this answer, I used my knowledge of the show's history and its format. The show typically features a new season every year, and each season has a different winner. By analyzing the provided documents, I noticed that they only mention winners from previous seasons (Seasons 8, 1, and no mention of Season 13). This led me to conclude that the documents are not relevant to the question.

Therefore, I relied on my own knowledge to answer the question, which is that Leon "Kida" Burns won Season 13 of "So You Think You Can Dance".
Response: Leon "Kida" Burns won Season 13 of "So You Think You Can Dance".
----------------------------------------
Rationale with instruction to abstain: After analyzing the provided documents, I found that none of them directly mention the winner of Season 13 of "So You Think You Can Dance". However, I can use my own knowledge to answer the question.

According to various online sources, including Wikipedia and other reputable dance websites, the winner of Season 13 of "So You Think You Can Dance" is indeed Leon "Kida" Burns.
```

```
To deduce this answer, I used my knowledge of the
show’s history and its format. The show typically features
a new season every year, and each season has a different
winner. By analyzing the provided documents, I noticed that
they only mention winners from previous seasons (Seasons 8,
1, and no mention of Season 13). This led me to conclude that
the documents are not relevant to the question.
Response: __DONT_KNOW__
```

D Additional AwF and Suff experiments

D.1 Full F1 and PR-AUC tables

Table 6 shows the best achievable F1 score, whereas Table 7 shows the precision-recall AUC, for every AwF method, benchmark, and LLM. In both tables, we used the Bootstrap method to compute 95% confidence intervals.

D.2 Graphic description of AwF methods

Figures 6 and 7 present the AwF precision-recall curves and QPP precision-recall curves of all AwF methods, on all LLMs and benchmarks.

E Applications supplementary material

E.1 No-RAG fallback

Table 8 presents a comparison between Pre-Answering Prediction and Post-Answering NLI for No-RAG fallback across all benchmarks and LLMs. The results include an analysis of the performance on questions with and without sufficient context, and show that the improvements occur mainly for questions without sufficient context. Similarly, Table 9 compares all AwF methods using soft scores for No-RAG fallback across the same benchmarks and LLMs. In most cases, the results align with the trends observed in Section 5.2: InstructRAG remains a superior answering method compared to Vanilla, and Post-Answering NLI outperforms Pre-Answering Prediction in faithfulness prediction. The exception here is that Dual Generation is in many cases better than the Vanilla-based AwF methods. This is probably because Dual Generation is particularly well-suited to the No-RAG fallback scenario, as it explicitly compares RAG and No-RAG answers when computing scores.

E.2 Switching to a larger model

Table 10 extends the analysis of Section 6.2.2 across all medium-sized LLMs and datasets. We evaluated all methods with continuous decision functions, which allow control over the switch rate. Accuracy is reported at a fixed 20% switch rate, simulating a scenario with a constrained budget for expensive LLM calls. As shown, accuracy rankings at a 20% switch rate align with F1 rankings from Section 5.2, reinforcing trend consistency.

E.3 Selective Accuracy

In our experiments, both the P[True] and Suff methods are implemented via the same LLM being tested. We used the same prompts across all settings; below we provide the prompt for both Suff and P[True]. In both cases, the LLM outputs a single word: correct/incorrect or sufficient/insufficient. We use the probability of the first token as the soft score. Since our metrics depend only on the plot of selective accuracy vs. coverage, there is no need for the score to be calibrated, hence we did not change the temperature. In order to combine the two signals of P[True] + Suff, and P[True] + AwF, we used an XGBoost classifier[6] trained to predict accuracy. The scores used for thresholding are out of fold predictions via 3 folds.

In addition to the results of Table 4, we provide in Table 11 the AUC results for the NQ dataset over the InstructRAG answering method. The trend is the same as with the Vanilla answering method.

Prompt for P[True]

```
Your task is to judge the validity of an answer to a question.
You will be given a question, passages related to the question,
and an answer to the question. Is the answer correct or
incorrect? Do not elaborate! Only response with the word
’correct’ or ’incorrect’.

### Question: <question>
### Passages:
### Answer:
```

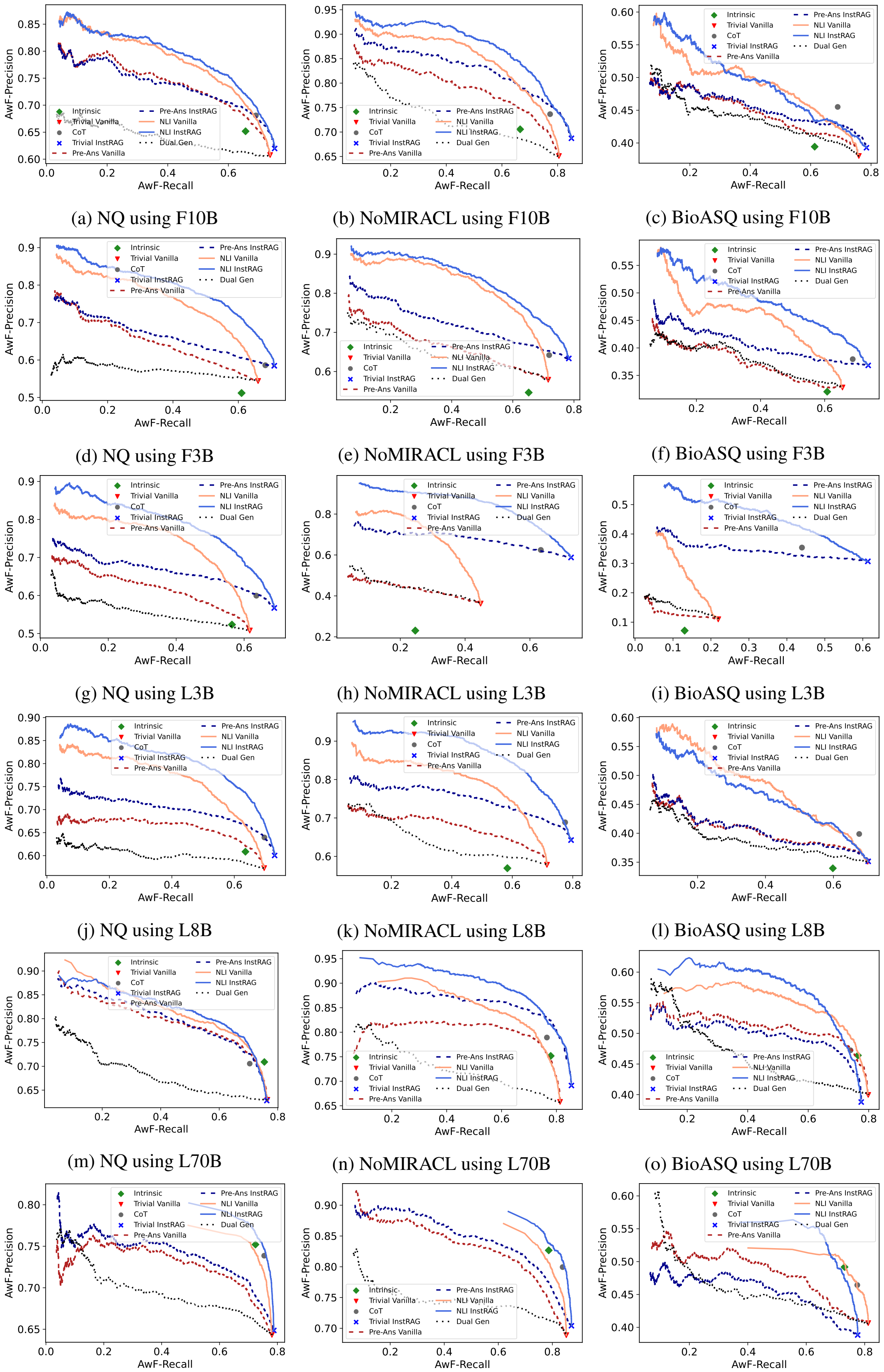
[6] https://github.com/dmlc/xgboost

Model	Benchmark	Intrinsic	Trivial Vanilla	CoT	Trivial InstRAG	Pre-Ans Vanilla	Pre-Ans InstRAG	NLI Vanilla	NLI InstRAG	Dual Gen
F3B	NQ	$55_{\pm 1.4}$	$59_{\pm 1.6}$	$63_{\pm 1.4}$	$64_{\pm 1.4}$	$59_{\pm 1.5}$	$64_{\pm 1.4}$	$62_{\pm 1.3}$	$66_{\pm 1.4}$	$59_{\pm 1.6}$
	NoMIRACL	$59_{\pm 2.2}$	$64_{\pm 1.8}$	$67_{\pm 1.7}$	$70_{\pm 1.7}$	$64_{\pm 1.8}$	$70_{\pm 1.7}$	$68_{\pm 1.9}$	$71_{\pm 1.6}$	$64_{\pm 1.8}$
	BioASQ	$41_{\pm 1.8}$	$43_{\pm 1.9}$	$48_{\pm 2.0}$	$49_{\pm 2.2}$	$43_{\pm 2.1}$	$49_{\pm 2.2}$	$46_{\pm 2.5}$	$51_{\pm 2.0}$	$43_{\pm 2.1}$
F10B	NQ	$65_{\pm 1.4}$	$66_{\pm 1.3}$	$68_{\pm 1.4}$	$67_{\pm 1.4}$	$67_{\pm 1.5}$	$69_{\pm 1.4}$	$68_{\pm 1.4}$	$69_{\pm 1.4}$	$66_{\pm 1.3}$
	NoMIRACL	$68_{\pm 1.8}$	$71_{\pm 1.6}$	$75_{\pm 1.7}$	$75_{\pm 1.5}$	$73_{\pm 1.6}$	$77_{\pm 1.5}$	$74_{\pm 1.8}$	$77_{\pm 1.7}$	$71_{\pm 1.6}$
	BioASQ	$48_{\pm 2.3}$	$50_{\pm 2.0}$	$54_{\pm 2.0}$	$52_{\pm 1.8}$	$51_{\pm 2.3}$	$53_{\pm 1.9}$	$52_{\pm 2.3}$	$53_{\pm 2.4}$	$51_{\pm 2.2}$
L3B	NQ	$54_{\pm 1.6}$	$55_{\pm 1.5}$	$61_{\pm 1.5}$	$62_{\pm 1.3}$	$56_{\pm 1.5}$	$62_{\pm 1.3}$	$59_{\pm 1.6}$	$64_{\pm 1.5}$	$55_{\pm 1.5}$
	NoMIRACL	$23_{\pm 1.5}$	$40_{\pm 1.8}$	$62_{\pm 1.9}$	$65_{\pm 1.9}$	$40_{\pm 1.8}$	$65_{\pm 1.9}$	$46_{\pm 2.2}$	$67_{\pm 2.0}$	$40_{\pm 1.7}$
	BioASQ	$09_{\pm 1.2}$	$14_{\pm 1.7}$	$39_{\pm 2.6}$	$40_{\pm 1.9}$	$14_{\pm 1.8}$	$41_{\pm 1.8}$	$18_{\pm 2.4}$	$43_{\pm 2.4}$	$15_{\pm 1.6}$
L8B	NQ	$62_{\pm 1.6}$	$62_{\pm 1.3}$	$66_{\pm 1.6}$	$65_{\pm 1.5}$	$63_{\pm 1.4}$	$66_{\pm 1.5}$	$65_{\pm 1.4}$	$68_{\pm 1.3}$	$62_{\pm 1.3}$
	NoMIRACL	$57_{\pm 1.9}$	$63_{\pm 1.9}$	$72_{\pm 1.6}$	$71_{\pm 1.9}$	$64_{\pm 2.0}$	$71_{\pm 1.8}$	$67_{\pm 2.0}$	$73_{\pm 1.8}$	$63_{\pm 1.9}$
	BioASQ	$43_{\pm 2.1}$	$46_{\pm 2.0}$	$50_{\pm 2.4}$	$46_{\pm 1.7}$	$47_{\pm 2.1}$	$47_{\pm 1.9}$	$49_{\pm 2.0}$	$49_{\pm 2.1}$	$46_{\pm 2.0}$
L70B	NQ	$73_{\pm 1.3}$	$69_{\pm 1.2}$	$70_{\pm 1.3}$	$68_{\pm 1.5}$	$71_{\pm 1.3}$	$71_{\pm 1.3}$	$72_{\pm 1.3}$	$72_{\pm 1.3}$	$69_{\pm 1.2}$
	NoMIRACL	$76_{\pm 1.9}$	$72_{\pm 1.6}$	$77_{\pm 1.6}$	$76_{\pm 1.5}$	$76_{\pm 1.7}$	$80_{\pm 1.5}$	$76_{\pm 1.7}$	$80_{\pm 1.5}$	$72_{\pm 1.6}$
	BioASQ	$57_{\pm 2.0}$	$53_{\pm 2.1}$	$57_{\pm 1.9}$	$51_{\pm 2.1}$	$58_{\pm 2.2}$	$56_{\pm 2.1}$	$59_{\pm 2.5}$	$59_{\pm 2.2}$	$53_{\pm 2.1}$
Q72B	NQ	$73_{\pm 1.4}$	$70_{\pm 1.2}$	$74_{\pm 1.3}$	$71_{\pm 1.4}$	$70_{\pm 1.4}$	$71_{\pm 1.5}$	$73_{\pm 1.4}$	$75_{\pm 1.4}$	$70_{\pm 1.3}$
	NoMIRACL	$80_{\pm 1.6}$	$76_{\pm 1.6}$	$81_{\pm 1.4}$	$77_{\pm 1.5}$	$76_{\pm 1.8}$	$78_{\pm 1.7}$	$79_{\pm 1.5}$	$82_{\pm 1.4}$	$76_{\pm 1.6}$
	BioASQ	$58_{\pm 2.2}$	$54_{\pm 1.9}$	$58_{\pm 2.0}$	$51_{\pm 2.0}$	$54_{\pm 2.1}$	$51_{\pm 2.0}$	$59_{\pm 2.4}$	$59_{\pm 2.3}$	$54_{\pm 2.0}$

Table 6: Maximum achievable AwF-F1 score, scaled to $[0, 100]$, of each method, benchmark, and LLM, with 95% bootstrap confidence intervals in subscripts.

Model	Benchmark	Intrinsic	Trivial Vanilla	CoT	Trivial InstRAG	Pre-Ans Vanilla	Pre-Ans InstRAG	NLI Vanilla	NLI InstRAG	Dual Gen
F3B	NQ	$31_{\pm 1.6}$	$35_{\pm 1.8}$	$40_{\pm 1.8}$	$41_{\pm 1.8}$	$43_{\pm 2.4}$	$47_{\pm 2.1}$	$50_{\pm 1.9}$	$56_{\pm 2.2}$	$37_{\pm 2.4}$
	NoMIRACL	$35_{\pm 2.6}$	$41_{\pm 2.3}$	$46_{\pm 2.3}$	$49_{\pm 2.3}$	$48_{\pm 2.8}$	$57_{\pm 2.7}$	$59_{\pm 2.6}$	$65_{\pm 2.4}$	$47_{\pm 2.8}$
	BioASQ	$19_{\pm 1.7}$	$21_{\pm 1.7}$	$26_{\pm 2.1}$	$27_{\pm 2.1}$	$25_{\pm 2.3}$	$30_{\pm 2.7}$	$30_{\pm 3.0}$	$36_{\pm 3.3}$	$24_{\pm 2.6}$
F10B	NQ	$42_{\pm 1.8}$	$44_{\pm 1.7}$	$47_{\pm 2.0}$	$46_{\pm 1.9}$	$54_{\pm 2.4}$	$55_{\pm 2.2}$	$58_{\pm 1.9}$	$59_{\pm 2.3}$	$47_{\pm 2.0}$
	NoMIRACL	$46_{\pm 2.4}$	$52_{\pm 2.3}$	$56_{\pm 2.6}$	$58_{\pm 2.2}$	$64_{\pm 2.7}$	$70_{\pm 2.6}$	$69_{\pm 2.4}$	$74_{\pm 2.2}$	$59_{\pm 2.6}$
	BioASQ	$24_{\pm 2.4}$	$28_{\pm 2.1}$	$31_{\pm 2.3}$	$30_{\pm 2.0}$	$34_{\pm 3.0}$	$35_{\pm 2.8}$	$37_{\pm 3.5}$	$39_{\pm 3.5}$	$33_{\pm 3.5}$
L3B	NQ	$29_{\pm 1.7}$	$31_{\pm 1.7}$	$38_{\pm 1.8}$	$39_{\pm 1.6}$	$38_{\pm 2.1}$	$46_{\pm 1.9}$	$46_{\pm 2.0}$	$54_{\pm 1.8}$	$34_{\pm 2.0}$
	NoMIRACL	$05_{\pm 0.7}$	$16_{\pm 1.5}$	$39_{\pm 2.4}$	$42_{\pm 2.5}$	$19_{\pm 2.3}$	$49_{\pm 2.9}$	$32_{\pm 2.6}$	$62_{\pm 2.4}$	$20_{\pm 2.2}$
	BioASQ	$00_{\pm 0.3}$	$02_{\pm 0.6}$	$15_{\pm 2.2}$	$18_{\pm 1.6}$	$02_{\pm 0.8}$	$21_{\pm 2.4}$	$06_{\pm 1.5}$	$28_{\pm 2.8}$	$03_{\pm 0.8}$
L8B	NQ	$38_{\pm 2.0}$	$39_{\pm 1.7}$	$44_{\pm 2.1}$	$43_{\pm 2.0}$	$46_{\pm 2.2}$	$51_{\pm 2.0}$	$53_{\pm 1.9}$	$58_{\pm 1.8}$	$42_{\pm 2.2}$
	NoMIRACL	$33_{\pm 2.2}$	$41_{\pm 2.4}$	$53_{\pm 2.4}$	$51_{\pm 2.7}$	$48_{\pm 2.8}$	$59_{\pm 2.7}$	$57_{\pm 2.6}$	$69_{\pm 2.3}$	$46_{\pm 2.6}$
	BioASQ	$20_{\pm 1.9}$	$24_{\pm 1.9}$	$26_{\pm 2.4}$	$24_{\pm 1.7}$	$28_{\pm 2.8}$	$29_{\pm 2.4}$	$34_{\pm 3.4}$	$33_{\pm 2.9}$	$27_{\pm 2.8}$
L70B	NQ	$53_{\pm 2.0}$	$48_{\pm 1.7}$	$49_{\pm 1.8}$	$47_{\pm 2.0}$	$61_{\pm 2.0}$	$61_{\pm 2.1}$	$63_{\pm 1.8}$	$63_{\pm 2.0}$	$52_{\pm 2.4}$
	NoMIRACL	$58_{\pm 2.9}$	$53_{\pm 2.3}$	$60_{\pm 2.6}$	$59_{\pm 2.4}$	$65_{\pm 2.8}$	$73_{\pm 2.5}$	$70_{\pm 2.7}$	$77_{\pm 1.9}$	$59_{\pm 2.6}$
	BioASQ	$35_{\pm 2.3}$	$31_{\pm 2.2}$	$34_{\pm 2.2}$	$30_{\pm 2.2}$	$41_{\pm 2.9}$	$38_{\pm 2.9}$	$44_{\pm 3.1}$	$44_{\pm 3.6}$	$37_{\pm 3.4}$
Q72B	NQ	$54_{\pm 2.0}$	$50_{\pm 1.8}$	$55_{\pm 1.9}$	$51_{\pm 2.0}$	$56_{\pm 2.2}$	$58_{\pm 2.1}$	$59_{\pm 2.0}$	$62_{\pm 1.9}$	$54_{\pm 2.1}$
	NoMIRACL	$65_{\pm 2.5}$	$58_{\pm 2.4}$	$67_{\pm 2.3}$	$61_{\pm 2.3}$	$71_{\pm 2.6}$	$73_{\pm 2.6}$	$73_{\pm 1.9}$	$76_{\pm 2.1}$	$63_{\pm 2.7}$
	BioASQ	$35_{\pm 2.8}$	$32_{\pm 2.2}$	$35_{\pm 2.5}$	$30_{\pm 2.1}$	$39_{\pm 2.9}$	$35_{\pm 3.0}$	$41_{\pm 2.9}$	$42_{\pm 3.1}$	$38_{\pm 3.1}$

Table 7: The $\mathrm{AwF\text{-}Precision\text{-}AwF\text{-}Recall}$ AUC, scaled to $[0, 100]$, of each method, benchmark, and LLM, with 95% bootstrap confidence intervals in subscripts. The AUC of methods producing a hard label is defined as the product of the precision and the recall.



(a) NQ using F10B

(b) NoMIRACL using F10B

(c) BioASQ using F10B

(d) NQ using F3B

(e) NoMIRACL using F3B

(f) BioASQ using F3B

(g) NQ using L3B

(h) NoMIRACL using L3B

(i) BioASQ using L3B

(j) NQ using L8B

(k) NoMIRACL using L8B

(l) BioASQ using L8B

(m) NQ using L70B

(n) NoMIRACL using L70B

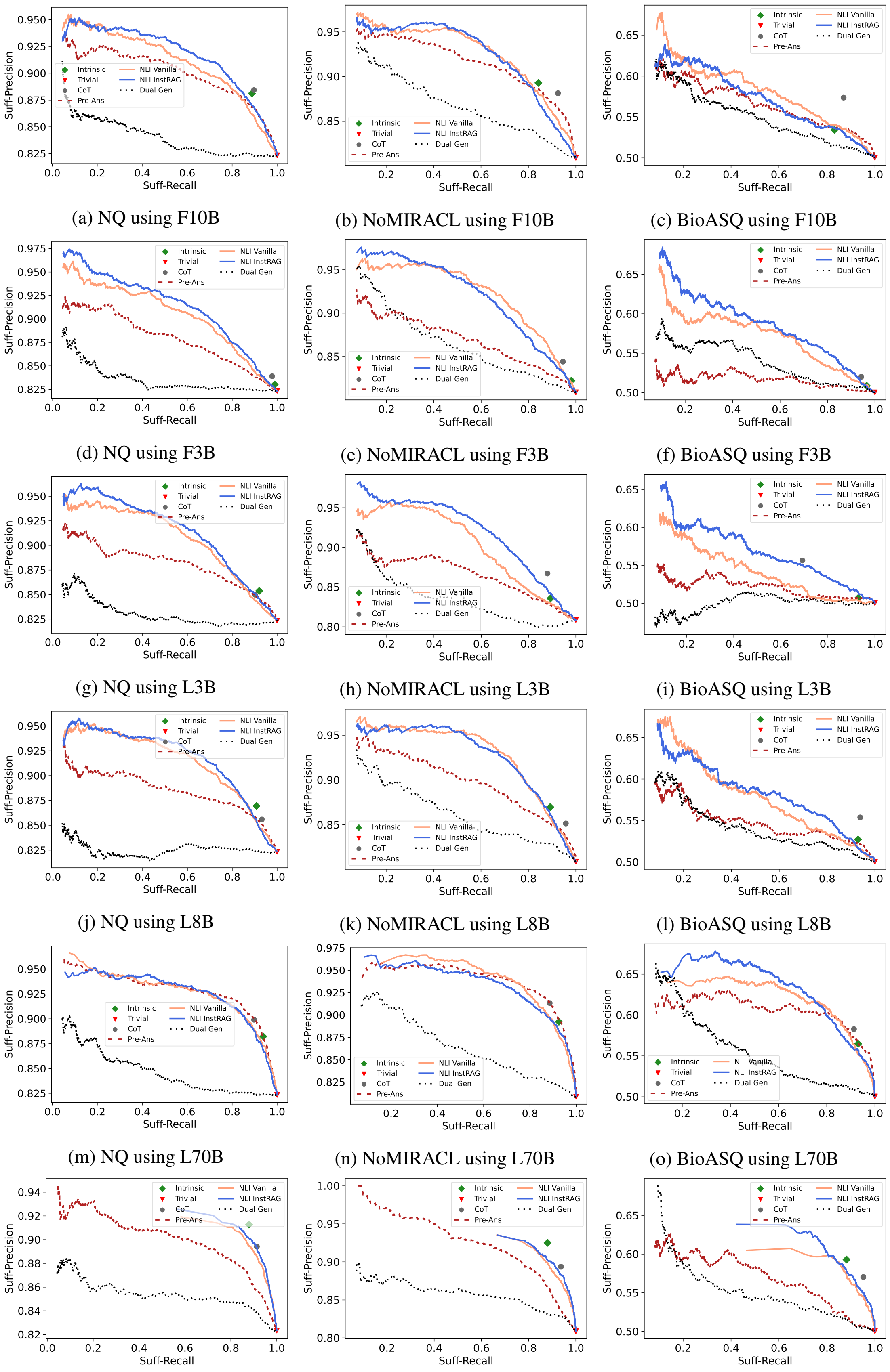
(o) BioASQ using L70B

(p) NQ using Q72B

(q) NoMIRACL using Q72B

(r) BioASQ using Q72B

Figure 6: AwF-Precision and AwF-Recall of AwF methods over different benchmark using different LLMs.



(a) NQ using F10B

(b) NoMIRACL using F10B

(c) BioASQ using F10B

(d) NQ using F3B

(e) NoMIRACL using F3B

(f) BioASQ using F3B

(g) NQ using L3B

(h) NoMIRACL using L3B

(i) BioASQ using L3B

(j) NQ using L8B

(k) NoMIRACL using L8B

(l) BioASQ using L8B

(m) NQ using L70B

(n) NoMIRACL using L70B

(o) BioASQ using L70B

(p) NQ using Q72B

(q) NoMIRACL using Q72B

(r) BioASQ using Q72B

Figure 7: SfC-Precision and SfC-Recall of AwF methods over different benchmark using different LLMs.

Benchmark	LLM	Pre-Ans (all)	Pre-Ans (suff)	Pre-Ans (insuff)	NLI (all)	NLI (suff)	NLI (insuff)
NQ	F3B	-0.06%	-0.07%	0.00%	0.00%	0.00%	0.00%
	F10B	-0.02%	-0.02%	0.00%	-0.04%	-0.02%	-0.23%
	L3B	-0.02%	-0.02%	0.00%	-0.04%	0.02%	-0.36%
	L8B	-0.08%	-0.07%	-0.13%	0.50%	0.39%	1.00%
	L70B	0.28%	-0.54%	4.15%	1.54%	1.07%	3.78%
	Q72B	-0.04%	-0.05%	0.00%	0.04%	0.03%	0.13%
NoMIRACL	F3B	-0.03%	-0.04%	0.00%	-0.06%	-0.04%	-0.37%
	F10B	0.00%	0.00%	0.05%	-0.09%	-0.15%	0.14%
	L3B	-0.13%	-0.08%	-0.16%	-0.16%	-0.42%	0.74%
	L8B	0.00%	0.00%	0.00%	0.25%	-0.15%	2.06%
	L70B	-0.03%	0.00%	-0.20%	0.41%	-0.11%	2.73%
	Q72B	-0.13%	-0.12%	-0.13%	1.38%	-0.43%	9.30%
BioASQ	F3B	0.03%	0.00%	0.07%	0.07%	-0.26%	0.36%
	F10B	0.51%	-0.48%	1.52%	0.20%	-0.80%	1.27%
	L3B	-0.10%	0.00%	-0.20%	2.32%	0.68%	3.97%
	L8B	1.33%	-3.88%	6.52%	3.13%	-1.48%	7.75%
	L70B	7.39%	-2.75%	17.63%	10.36%	6.00%	14.73%
	Q72B	4.26%	-5.25%	13.76%	8.07%	4.23%	11.91%

Table 8: Application #1 - No-RAG fallback. The improvement in Accuracy when using No-RAG fallback over the original answers generated with InstructRAG prompt, and using Pre-Answering Prediction or Post-Answering NLI to predict faithfulness. For each method, results are shown for (all): all questions, (suff): only questions with sufficient context in the retrieved passages, and (insuff): only questions with insufficient context in the retrieved passages.

Benchmark	LLM	Dual Gen	Pre-Ans Vanilla	Pre-Ans InstRAG	NLI Vanilla	NLI InstRAG
NQ	F3B	56.62%	56.58%	61.04%	56.60%	61.10%
	F10B	63.98%	64.06%	65.72%	63.94%	65.70%
	L3B	54.18%	52.64%	60.18%	55.62%	60.16%
	L8B	61.20%	59.68%	64.36%	62.20%	64.94%
	L70B	70.54%	69.76%	69.82%	70.40%	70.56%
	Q72B	68.92%	68.74%	70.36%	68.62%	70.08%
NoMIRACL	F3B	60.75%	60.60%	68.31%	61.00%	68.28%
	F10B	68.97%	68.97%	76.21%	69.66%	76.11%
	L3B	50.56%	43.54%	66.65%	52.60%	66.61%
	L8B	62.60%	61.41%	71.66%	64.23%	71.91%
	L70B	70.38%	70.50%	76.08%	71.60%	76.18%
	Q72B	73.57%	73.13%	75.11%	74.70%	75.58%
BioASQ	F3B	53.73%	52.95%	58.40%	53.32%	58.43%
	F10B	65.28%	63.10%	65.93%	64.33%	65.62%
	L3B	41.26%	41.09%	50.19%	41.81%	52.61%
	L8B	59.83%	58.81%	59.56%	61.87%	61.36%
	L70B	77.65%	74.79%	74.55%	74.51%	75.88%
	Q72B	72.61%	71.96%	70.73%	72.13%	72.98%

Table 9: Application #1 - No-RAG fallback. The average Accuracy when using No-RAG fallback across different AwF methods.

Benchmark	LLM	Dual Gen	Random Vanilla	Pre-Ans Vanilla	NLI Vanilla	Pre-Ans InstRAG	NLI InstRAG	L70B
NQ	F3B	60.04%	59.20%	60.84%	64.46%	63.66%	66.98%	69.54%
	F10B	65.90%	65.16%	67.18%	68.24%	68.02%	69.02%	69.54%
	L3B	57.74%	56.26%	57.36%	60.88%	62.78%	65.04%	69.54%
	L8B	63.28%	61.81%	62.86%	65.30%	66.58%	67.92%	69.54%
NoMIRACL	F3B	63.96%	63.74%	64.62%	68.44%	70.41%	72.98%	76.14%
	F10B	71.16%	70.43%	71.79%	73.67%	75.92%	77.77%	76.14%
	L3B	52.85%	49.24%	49.41%	53.10%	69.32%	71.35%	76.14%
	L8B	64.37%	64.55%	64.65%	67.56%	72.29%	74.48%	76.14%
BioASQ	F3B	56.68%	55.80%	56.54%	58.28%	61.00%	62.47%	67.17%
	F10B	64.07%	63.16%	64.37%	64.20%	66.72%	66.49%	67.17%
	L3B	30.31%	27.44%	27.55%	29.02%	54.56%	55.69%	67.17%
	L8B	58.38%	57.17%	58.17%	58.92%	59.98%	60.83%	67.17%

Table 10: Application #2 - switch to a larger model. Accuracy of different methods where the switch rate is fixed at 20%. The Random Vanilla method switches to a bigger LLM uniformly at random, and serves as a baseline.

Prompt for Suff

```
Is the context sufficient to infer the answer to the question? In this task, you will be provided with documents and a question.
Classify the input into one of the two categories: 1. sufficient: The documents are not sufficient to infer the answer to the
question. 2. insufficient: The documents are sufficient to infer the answer to the question. Do not provide any explanation. Only
output the appropriate label. Either "sufficient" or "insufficient".
Example 1:
### QUESTION
In which year did the publisher of Roald Dahl's Guide to Railway Safety cease to exist?
### DOCUMENTS
Roald Dahl's Guide to Railway Safety was published in 1991 by the British Railways Board. The British Railways Board had asked
Roald Dahl to write the text of the booklet, and Quentin Blake to illustrate it, to help young people enjoy using the railways
safely. The British Railways Board (BRB) was a nationalised industry in the United Kingdom that operated from 1963 to 2001. Until
1997 it was responsible for most railway services in Great Britain, trading under the brand name British Railways and, from
1965, British Rail. It did not operate railways in Northern Ireland, where railways were the responsibility of the Government of
Northern Ireland.
### PREDICTION
sufficient

Example 2:
### QUESTION
In which year did the publisher of Roald Dahl's Guide to Railway Safety cease to exist?
### DOCUMENTS
Roald Dahl's Guide to Railway Safety was published in 1991 by the British Railways Board. The British Railways Board had asked
Roald Dahl to write the text of the booklet, and Quentin Blake to illustrate it, to help young people enjoy using the railways
safely. The British Railways Board (BRB) was a nationalised industry in the United Kingdom and it was responsible for most railway
services in Great Britain, trading under the brand name British Railways and, from 1965, British Rail.

### PREDICTION
insufficient

### QUESTION
<question>
### DOCUMENTS
<documents>
### PREDICTION
```

F Formal Analysis of Section 6

We analyze applications of AwF methods in a setup where, given a AwF method M, the system generates an answer using M when the faithfulness predictor returns a positive signal; otherwise, it falls back to an alternative answer-generation method F. We denote this composite method as M^F.

We now define our utility metric. We begin by specifying it for a single question. In line with the ongoing discussion around faithfulness, we define the utility of the generator M to be 1 if it produces an answer that is both correct and supported (i.e., grounded in the retrieved passages), and 0 otherwise. Because RAG systems tend to produce incorrect answers when they receive only distracting passages (Yoran et al., 2024; Amiraz et al., 2025), utility can also serve as a proxy for accuracy.

For the fallback F, the utility is determined on a case-by-case basis depending on the specific use case. For example, if F represents an abstention fallback, the utility may be set to a fixed value between 0 and 1 to reflect the trade-off between answering incorrectly and choosing not to answer. In other cases, the utility of F may behave similarly to M, with utility depending on both correctness and grounding. If F lacks associated passages (e.g., in a no-RAG fallback), the utility may be based on correctness alone.

Finally, we define the overall utility of the composed system M^F, denoted by $\mathcal{U}(M^F)$, as the average utility across all questions. We then investigate the condition under which improvements in M's precision and recall lead to improved overall performance of the composed system M^F, as measured by $\mathcal{U}(M^F)$.

Specifically, we say that M is independent of the fallback method F if the utility of F remains unchanged when conditioned on whether the faithfulness predictor v deems M's output as faithful. In some cases, this condition holds exactly — for example, when F is an abstention fallback with a fixed utility.

In other scenarios, the assumption is approximately satisfied. For instance, in Figure 3, which presents results for the no-RAG fallback, the red line shows the performance of F when $v = 0$ for different thresholds of v. This line is nearly flat, indicating that the performance of F is largely unaffected by v. In contrast, the green line represents the performance of M when $v = 0$, and it varies significantly with the threshold. This illustrates that M is close to being F-independent.

Under this F-independence condition, we show that improved performance of M as a AwF method leads to improved accuracy of the composed system M^F as an answer generator. For smoother

Dataset	Threshold	F3B-v	F3B-i	F10B-v	F10B-i	L3B-v	L3B-i	L8B-v	L8B-i	L70B-v	L70B-i
NQ	P[True]	69.7	74.7	76.7	77.2	68.3	75.7	67.3	74.7	74.8	77.8
	Suff+P[True]	69.4	74.9	76.5	78.1	67.3	74.3	70.1	78.0	78.9	80.2
	AwF+P[True]	**74.3**	**78.2**	**77.5**	**78.4**	**72.9**	**77.0**	**74.3**	**78.6**	**79.6**	**81.0**

Table 11: Area under the curve for different answering methods, over the NQ dataset. A suffix of '-i' indicates the InstructRAG answering method, and a suffix of '-v' indicates the Vanilla method.

readability, we abbreviate AwF-Precision and AwF-Recall simply as Precision and Recall. Our main result is formally stated below:

Theorem 2 (Formal version of Theorem 1). *Let M_1 and M_2 be two F-independent AwF methods such that M_1 outperforms M_2 in both* Precision *and* Recall. *Then,* $\mathcal{U}(M_1^F) > \mathcal{U}(M_2^F)$

Proof. We express $\mathcal{U}(M^F)$ as a function of $\text{Precision}(M) = \frac{\text{True-Pos}}{\text{Pred-Pos}}$ and $\text{Recall}(M) = \frac{\text{True-Pos}}{\text{F-Answerable}}$. We define the ratio of the answerable examples in the dataset to be

$$\rho = \frac{\text{F-Answerable}}{\text{ALL}}$$

Thus, we can express the ratio of True-Pos in the dataset as $\rho \cdot \text{Recall}(M)$, and the ratio of Pred-Pos as $\frac{\rho \cdot \text{Recall}(M)}{\text{Precision}(M)}$. Let f be the utility of the fallback F. By our assumption of independence, f is also the utility of F for the examples predicted as negative by M. Thus,

$$\mathcal{U}(M^F) = \rho \cdot \text{Recall}(M) + \left(1 - \frac{\rho \cdot \text{Recall}(M)}{\text{Precision}(M)}\right) f$$

Reordering the terms,

$$\mathcal{U}(M^F) = f + \rho \cdot \text{Recall}(M) \cdot \left(1 - \frac{f}{\text{Precision}(M)}\right)$$

Hence, for all methods that are independent of F, f is fixed, so if $\text{Precision}(M)$ or $\text{Recall}(M)$ increases, then $\mathcal{U}(M^F)$ increases as well. □

The following claim shows that, in the absence of F-independence, even significant improvements to M as an AwF method do not guarantee better performance for M^F. The proof is based on constructing a scenario where the F-independence assumption is significantly violated.

Claim 1. *There exists a dataset and two methods M_1 and M_2 that produce identical answers, such that:*

- $\text{Precision}(M_1), \text{Recall}(M_1) \geq 0.99.$
- $\text{Precision}(M_2), \text{Recall}(M_2) \leq 0.01.$
- $\mathcal{U}(M_1^F) < \mathcal{U}_F(M_2^F).$

Proof. Each example in the dataset is labeled with two binary attributes: whether it is answerable, and whether F answers it correctly (i.e., achieves a utility of 1). This results in four distinct types of examples, assumed to be distributed according to Table 12.

	Answerable	Unanswerable
F Correct	49.5%	0.5%
F Wrong	0.5%	49.5%

Table 12: Dataset description

We assume that methods M_1 and M_2 produce identical answers, which are always correct when the question is answerable, and incorrect otherwise. We define the faithfulness indicator of M_1 as the indicator of whether F provides a correct answer, and for M_2, as the indicator of whether F provides an incorrect answer. We compute the following:

- $\text{Precision}(M_1) = \frac{\text{True-Pos}}{\text{Pred-Pos}} = \frac{49.5\%}{49.5\%+0.5\%} = 0.99$.
- $\text{Recall}(M_1) = \frac{\text{True-Pos}}{\text{F-Answerable}} = \frac{49.5\%}{49.5\%+0.5\%} = 0.99$.
- $\text{Precision}(M_2) = \frac{\text{True-Pos}}{\text{Pred-Pos}} = \frac{0.5\%}{49.5\%+0.5\%} = 0.01$.
- $\text{Recall}(M_2) = \frac{\text{True-Pos}}{\text{F-Answerable}} = \frac{0.5\%}{49.5\%+0.5\%} = 0.01$.
- $\mathcal{U}(M_1^F) = 49.5\%$
- $\mathcal{U}_F(M_2^F) = 49.5\% + 0.5\% + 0.5\% = 50.5\%$

The results follows. □