The RAT: A Unified Bayesian Model for RAG Evaluation

Pius von Däniken* **Felix Matthias Saaro***
Mark Cieliebak **Jan Milan Deriu**
Centre for Artificial Intelligence
ZHAW School of Engineering
{vode,saaf,ciel,deri}@zhaw.ch

Abstract

Evaluating Retrieval-Augmented Generation (RAG) systems requires assessing not only end-to-end correctness but also how individual components interact and how errors propagate through the pipeline. We introduce a Bayesian evaluation framework that jointly models retrieval success, abstention behavior, and answer correctness, factorized according to the pipeline's information flow. The model distinguishes *task success*. whether the user received a correct answer, from *generator success*, whether the generator behaved appropriately given the retrieval outcome. We apply the framework to 27 RAG configurations across three datasets, three retrievers, and three generators, and show that the conditional decomposition reveals substantial behavioral differences between systems that appear equivalent under marginal metrics. We further analyze the annotation allocation problem, demonstrating that retrieval-success annotations are more informative than task-success annotations for estimating policy adherence, and provide an information-theoretic explanation for this asymmetry. Finally, we extend the model to incorporate LLM-as-a-judge annotations as calibrated noisy observations, enabling practitioners to combine limited human judgments with cheaper automated assessments within a unified probabilistic model.

1 The Introduction

Evaluating Retrieval-Augmented Generation (RAG) systems is challenging and often tedious. It is not sufficient to evaluate only the system's end-to-end behavior; each component must be evaluated both in isolation and within the pipeline. Since RAG systems are built in a pipelined approach, errors tend to propagate through the pipeline. If retrieval fails, the generator has no basis for producing a correct answer and should abstain from answering the question. Most benchmarks developed for RAG evaluation consider each component in isolation (Es et al., 2024; Saad-Falcon et al., 2024; Rau et al., 2024), or evaluate generators using adversarial retrieval results (Wang et al., 2024). However, a holistic view of the evaluation that accounts for the dependencies between components is still missing.

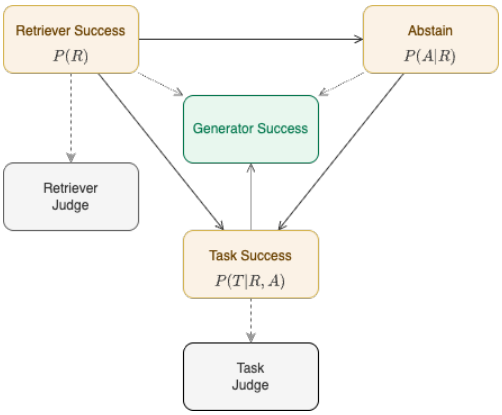

Figure 1: The dependency structure of the evaluation variables: generator success depends on retrieval success (which governs whether the generator should abstain or answer), abstention behavior, and task success.

A key shortcoming of end-to-end evaluation is that it conflates two distinct notions of success. *Task success* measures whether the user received a correct answer, whereas *generator success* measures whether the generator behaved appropriately given the retrieval outcome, i.e., answering when retrieval succeeded and abstaining when it failed. These two quantities can diverge substantially: a generator that never abstains may achieve reasonable task success while systematically violating the desired policy, and two systems with identical task success can exhibit very different behaviors under retrieval failure. To make such distinctions explicit, we need a model that captures the dependencies between retrieval, abstention, and answer correctness.

*Equal contribution

We propose a Bayesian evaluation framework that models these dependencies explicitly. Figure 1 illustrates the dependency structure, which shows how these variables interact: retrieval success governs both the abstention decision and the conditions under which task success is meaningful, and generator success is a deterministic function of all three. Casting this as a Bayesian model allows us to propagate uncertainty through the full dependency structure and to handle partially observed data by marginalizing over unobserved variables, which is a practical advantage when some annotations are expensive while others are cheap. The framework is further extended by incorporating LLM-as-a-judge annotations alongside human gold-standard labels, using a calibration model that treats automated judgments as noisy observations of the underlying ground-truth variables (von Däniken et al., 2022, 2024). This allows practitioners to combine small amounts of expensive human annotation with larger volumes of automated judgments in a principled way. We make the following contributions [1]:

- We introduce a Bayesian evaluation model for RAG systems that factorizes the joint distribution over retrieval success, abstention, and task success according to the pipeline's information flow, and derives generator success as a deterministic variable over these random variables (Section 3).
- We apply the model to 27 RAG configurations (3 retrievers $\times$ 3 generators $\times$ 3 datasets) and show that the conditional decomposition reveals behavioral differences that marginal metrics conceal, in particular, systems with near-identical task success can differ sharply in policy adherence.
- We analyze the annotation allocation problem: given a fixed budget, we derive and empirically validate which combination of retrieval and task-success annotations minimizes estimation error, and provide an information-theoretic explanation for the observed asymmetry.
- We extend the model to incorporate calibrated LLM-as-a-judge annotations as noisy observations, enabling practitioners to combine human and automated judgments within the same probabilistic framework.

[1] Our code can be found at: https://github.com/vodezhaw/rat.

2 The Related Work

Retrieval-Augmented Generation (RAG) is the process of incorporating the output of an information retrieval (IR) system into the context of a large language model (LLM), so that the LLM's answer is informed by relevant documents or passages (Lewis et al., 2020; Borgeaud et al., 2022). The process is composed of two components: retrieval and generation. The user's question is transformed into a query for an IR engine, which retrieves a set of relevant documents or passages, which are then given to the LLM to generate an answer to the user's query. This setting, while straightforward to describe, is highly challenging to evaluate. In general, the two components are evaluated separately. For an extensive overview of RAG evaluation, we refer the reader to (Yu et al., 2025).

Retrieval Evaluation. There are two main settings: either there is a gold standard available, that is, for a set of queries, all relevant documents, or there is no gold standard available. In the case of an existing gold standard, which is highly costly to create, one applies the standard scores used in the IR literature (Tang and Yang, 2024), such as Mean Reciprocal Rank, or Mean Average Precision (Schütze et al., 2008). In cases where no gold standard is available, two approaches are commonly used: either silver-standard generation or an unreferenced LLM-as-a-judge (Zheng et al., 2023). The generation of silver standards consists of an inverted process by providing an LLM with a document or passage, and letting the LLM generate questions (Es et al., 2024). Then the provided document serves as the relevant document (while disregarding other potentially relevant documents). For LLM-as-a-judge, one provides the LLM with the result of the retrieval and the user question, and asks to rate whether the question can be answered with the provided retrieval (Saad-Falcon et al., 2024).

Generator Evaluation. The evaluation of the generator often entangles two concepts: the end-to-end success, and the evaluation of the generator under different retrieval settings. For instance, Wang et al. (2024) evaluates the behavior of the generator under different adversarial retrieval settings. Chen et al. (2024) also highlights the importance of evaluating the impact of the retrieval on the generation in isolation. They also state the need to handle cases where the retrieval does not find relevant sources in the generator by allowing abstention from answer-

ing, and, in turn, evaluating the LLM's behavior in these cases.

Pipeline Evaluation. Rau et al. (2024) introduces a retrieval benchmarking library that focuses on evaluating pipeline components. ARES (Saad-Falcon et al., 2024) introduces a LLM-as-a-judge-based evaluation framework to evaluate each component of the pipeline separately using various dimensions. RAGAs (Es et al., 2024) introduces a reference-free framework for evaluating multiple dimensions of RAG pipelines without relying on ground-truth human annotations, covering context relevance, faithfulness, and answer relevance. Both RAGAs and ARES follow the RAG Triad framing popularized by *TruLens* (TruLens, 2024). CRUD-RAG (Lyu et al., 2025) is a benchmark oriented towards real-world scenarios rather than the QA tasks popular in academic settings. They include evaluation of multiple components, such as chunk size selection and reranking. While these frameworks evaluate multiple dimensions, they treat each dimension independently and do not model the statistical dependencies between retrieval and generation outcomes.

Our Work builds on these efforts and integrates them into a single probabilistic framework that jointly models the dependencies between retrieval, abstention, and answer correctness. This unified view enables distinguishing policy adherence from end-task success, propagating uncertainty across the pipeline, reasoning about annotation allocation, and combining human and automated judgments via calibration.

3 The Model

Evaluating a RAG system only by whether its final output matches a reference answer does not tell the full story. End-task correctness quantifies the system's overall output, but it does not distinguish among the generator's underlying behaviors. In particular, the common intuition that better retrieval should lead to better answers is only valid if the generator can use the retrieved information appropriately. Likewise, abstention can be desirable when retrieval fails, even though it may reduce overall answer rates and, in turn, affect aggregate performance metrics.

The Variables. To make these distinctions explicit, we model RAG behavior using four binary variables. Let R denote retrieval success, where $R = 1$ means that the retrieved context contains all information required to answer the question. Let A denote abstention, where $A = 1$ means that the generator declines to answer by producing an explicit "I don't know." response. Let T denote task success, where $T = 1$ means that the final answer is correct. Finally, let G denote generator success, or policy adherence, where $G = 1$ if the generator behaves as desired. We use binary variables as a deliberate simplification.

The Conditionals. Studying only the marginals of these variables is insufficient to characterize the system's behavior. For example, a system with slightly higher task success may still be less desirable if it achieves that gain by answering questions without appropriate support, rather than abstaining appropriately. To understand such trade-offs, we study conditional probabilities that separate retrieval quality, abstention behavior, and answer correctness.

We therefore propose a factorization of the joint distribution that follows the flow of information through the RAG pipeline:

$$P(R, A, T) = P(R)\, P(A \mid R)\, P(T \mid A, R).$$

In our data (see Section 4), task success is only possible for non-abstained responses, so effectively $T = 0$ whenever $A = 1$. This yields conditional probabilities with direct behavioral interpretations: $P(R)$ captures retrieval quality, $P(A \mid R)$ captures how abstention depends on retrieval success or failure, and $P(T \mid A = 0, R)$ captures answer correctness conditional on both the retrieval state and the decision to answer.

The Generator Success. Finally, we define generator success G deterministically from (R, A, T) to encode the desired policy: $G = \big((R = 0) \wedge (A = 1)\big) \vee \big((R = 1) \wedge (A = 0) \wedge (T = 1)\big)$. That is, the generator is successful if it abstains when retrieval fails, or if it answers correctly when retrieval succeeds. This construction makes it possible to distinguish correct behavior from correct output and to analyze how different RAG configurations trade off retrieval, abstention, and answer quality.

The Unified Model. We cast this model in a Bayesian framework. This provides two practical advantages for our setting. First, it propagates uncertainty throughout the full model, allowing us to quantify uncertainty not only for the primitive conditional probabilities but also for derived quantities such as policy adherence. Second, it naturally accommodates partially observed data by marginalizing over unobserved variables, which

is particularly useful when some annotations are expensive while others are cheap.

Under the factorization above, the model is parameterized by five probabilities: $\theta_R = P(R = 1)$, $\theta_{A^-} = P(A = 1 \mid R = 0)$, $\theta_{A^+} = P(A = 1 \mid R = 1)$, $\theta_{T^-} = P(T = 1 \mid R = 0, A = 0)$, and $\theta_{T^+} = P(T = 1 \mid R = 1, A = 0)$. These correspond directly to interpretable aspects of RAG behavior: retrieval success, abstention on retrieval failure, abstention despite successful retrieval, unsupported-answer success, and supported-answer success.

The Automated Judge Extension. To add the LLM-as-a-judge to the model, we extend the factorization by an additional term to represent the binary R_J and T_J variables: $P(R, A, T, R_J, T_J) = P(R)P(A|R)P(T|R, A)P(R_J, T_J|R, A, T)$. This adds 18 degrees of freedom to the model, corresponding to the 6 feasible (R, A, T) states and a 4-category conditional distribution for each state. The resulting judgment model is more complex than simpler factorizations such as $P(R_J|R)P(T_J|T)$, but avoids imposing conditional independence assumptions that were not supported by preliminary analyses.

We place independent uniform priors on all five basic model parameters and a Dirichlet prior on the conditional probability of automated judge observations. The resulting posterior can then be queried for both the primitive parameters and any deterministic function of them, including model-implied marginals and the derived policy-adherence quantity. We implement the model in Stan (Stan Development Team, 2026) and perform posterior inference using Hamiltonian Monte Carlo with the No-U-Turn Sampler (NUTS) (Hoffman and Gelman, 2014). For each experiment, we run five chains with 2,000 warmup iterations and collect 10,000 posterior samples per chain.

4 The Retrievers, The Generators, and The Tasks

The Data. This work employs the KILT benchmark (Petroni et al., 2021), a unified library for knowledge intensive languag tasks comprising 11 datasets across 5 task categories, including fact-checking and open-domain question answering. All datasets are grounded in a shared, pre-processed Wikipedia snapshot, thereby ensuring consistent evaluation conditions and facilitating interoperability across tasks with minimal preprocessing overhead.

- **FEVER (FEV)** (Thorne et al., 2018) is a fact-checking dataset consisting of approximately 126,000 human-annotated claims, each assigned one of three labels: *Supported*, *Refuted*, or *NotEnoughInfo*. The majority of queries have 1 relevant paragraph.
- **HotpotQA (HQA)** (Yang et al., 2018) is an open-domain question answering dataset comprising approximately 113,000 questions that necessitate multi-hop reasoning across multiple Wikipedia articles. The question-answer pairs were collected using crowd workers who were shown pairs of Wikipedia paragraphs and asked to write multi-hop questions along with their answer. All queries have 2 relevant paragraphs.
- **Natural Questions (NQ)** (Kwiatkowski et al., 2019) consists of approximately 92,000 naturally occurring queries submitted to the Google search engine, each paired with a long and short answer annotation curated by crowd workers. In this work, only the short answer annotations are used. All queries have exactly 1 relevant paragraph.

For each task, a subset of 10,000 queries was randomly selected. A single shared knowledge base of 1,720,160 paragraphs was subsequently constructed from all documents relevant to the selected queries across all tasks[2].

The Retrieval Strategies. Three strategies were evaluated:

- **Sparse (S)**, a term-based retrieval approach employing the BM25 ranking function based on lexical matching.
- **Dense (D)**, semantic retrieval approach leveraging multi-task text embeddings (Sturua et al., 2024), indexed using a Hierarchical Navigable Small World (HNSW) graph via the FAISS library (Douze et al., 2024) for efficient approximate nearest-neighbour search.
- **Hybrid (H)**, a combination of sparse and dense retrieval, with result fusion performed using Reciprocal Rank Fusion (RRF) (Cormack et al., 2009).

[2]This knowledge base represents a more controlled retrieval environment than a fully open-domain corpus. Consequently, the absolute retrieval-success rates (see Section 5) may be interpreted as optimistic relative to fully open-domain deployment.

DS	Gen.	Ret.	$P(R{=}1)$	$P(A{=}1)$	$P(T{=}1)$	$P(G{=}1)$
FEV	Apt	D	0.603	0.135	0.753	0.608
	Gem			0.180	0.784	0.736
	Qwn			0.199	0.751	**0.744**
	Apt	H	**0.655**	*0.089*	0.801	0.614
	Gem			0.096	**0.864**	0.701
	Qwn			0.112	0.838	0.703
	Apt	S	0.456	0.124	0.759	*0.504*
	Gem			0.221	0.744	0.644
	Qwn			**0.246**	*0.708*	0.657
HQA	Apt	D	0.151	0.038	0.260	*0.110*
	Gem			0.311	*0.253*	0.391
	Qwn			**0.489**	0.260	**0.574**
	Apt	H	0.207	*0.032*	0.304	0.138
	Gem			0.218	0.315	0.337
	Qwn			0.375	**0.329**	0.503
	Apt	S	**0.218**	0.033	0.318	0.152
	Gem			0.226	0.318	0.360
	Qwn			0.391	0.326	0.535
NQ	Apt	D	0.347	0.028	0.239	0.164
	Gem			0.256	0.242	0.414
	Qwn			0.355	0.243	0.496
	Apt	H	**0.348**	*0.024*	**0.267**	0.163
	Gem			0.198	0.267	0.359
	Qwn			0.307	0.266	0.451
	Apt	S	0.244	0.039	0.233	*0.136*
	Gem			0.300	*0.220*	0.412
	Qwn			**0.434**	0.215	**0.531**

Table 1: Marginal probabilities across all 27 RAG configurations. $P(R{=}1)$ is shared across generators within each dataset–retriever pair. Highest per-dataset values in **bold**, lowest in *italics*.

The Generators. Three large language models were employed as generators:

- **Apertus 8B** (Hernández-Cano et al., 2025), a fully open-source, decoder-only transformer model with 8 billion parameters, developed with an emphasis on full transparency of both model weights and training data.
- **Gemma3 12B** (Team et al., 2025), a lightweight open multimodal model developed by Google, comprising 12 billion parameters and trained on 12 trillion tokens spanning more than 140 languages.
- **Qwen3.5 9B** (Qwen Team, 2026), a multimodal model developed by Alibaba Cloud with 9 billion parameters, supporting multilingual inference across 201 languages. Although the model includes a dedicated reasoning mode, this functionality was not employed in the present work

5 The Experiments

5.1 The Marginals

Table 1 reports the marginal probabilities across all 27 configurations. We count retrieval as successful only when *all* relevant documents are present in the retrieved context[3]. Hybrid retrieval achieves the highest retrieval success on *FEV* and *NQ*, whereas sparse retrieval performs best on *HQA*. Retrieval success is substantially higher on *FEV* than on the other two datasets because each sample has exactly one relevant document, making the retrieval task considerably easier. Turning to the generator-level metrics, *Apertus* exhibits consistently low abstention rates across all datasets, with a maximum of 0.135 on *FEV* with dense retrieval. In contrast, *Gemma3* and *Qwen3.5* abstain substantially more often, reaching rates as high as 0.489 for *Qwen3.5* on *HQA* with dense retrieval.

Task success is highest on *FEV*, where each question has only two answer options, and lower on *HQA* and *NQ*, which require an exact match to an open-form reference answer. Across datasets, the three generation models achieve broadly similar task success rates. Policy adherence, however, differs much more strongly across models: *Qwen3.5* achieves the highest policy adherence on all datasets, whereas *Apertus* consistently lags behind.

5.2 The Conditionals

We fit the model introduced in Section 3 separately to the data from each configuration. The model-implied marginals closely match the corresponding count-based marginals up to Monte Carlo error. Table 2 therefore focuses on the conditional probabilities, which provide a more interpretable decomposition of system behavior.

A first pattern is that *Gemma3* and *Qwen3.5* distinguish between retrieval success and failure much more clearly than *Apertus*. Across datasets, both models assign substantially higher abstention probabilities when retrieval fails than when retrieval succeeds. For example, on *HQA* with hybrid retrieval, *Apertus* has $P(A{=}1 \mid R{=}0) = 0.039$ and $P(A{=}1 \mid R{=}1) = 0.005$, whereas *Gemma3* reaches 0.270 and 0.021, and *Qwen3.5* 0.463 and 0.036. Thus, under the same retrieval conditions, the models exhibit markedly different abstention policies. At the same time, abstention conditional

[3]This binary definition is a modeling simplification (see Section 3). We discuss partial retrieval in Appendix C.

DS	Gen.	Ret.	θ_{A-}	θ_{A+}	θ_{T-}	θ_{T+}
FEV	Apt	D	0.245	**0.063**	*0.810*	0.903
	Gem	D	0.422	0.021	0.941	0.962
	Qwn	D	**0.462**	0.027	0.894	0.954
	Apt	H	*0.158*	0.053	0.831	*0.901*
	Gem	H	0.242	0.019	0.940	0.961
	Qwn	H	0.274	0.027	0.913	0.955
	Apt	S	0.197	0.038	0.828	0.906
	Gem	S	0.391	*0.018*	**0.942**	**0.963**
	Qwn	S	0.428	0.028	0.916	0.956
HQA	Apt	D	0.044	0.010	*0.229*	*0.492*
	Gem	D	0.361	0.026	0.313	0.570
	Qwn	D	**0.568**	**0.044**	**0.458**	0.638
	Apt	H	*0.039*	*0.005*	0.258	0.519
	Gem	H	0.270	0.021	0.331	0.608
	Qwn	H	0.463	0.036	0.456	0.678
	Apt	S	0.041	0.006	0.264	0.552
	Gem	S	0.286	0.011	0.326	0.632
	Qwn	S	0.493	0.028	0.446	**0.704**
NQ	Apt	D	0.039	0.007	*0.161*	*0.402*
	Gem	D	0.382	0.021	0.192	0.485
	Qwn	D	0.513	0.060	0.256	0.494
	Apt	H	*0.034*	*0.006*	0.179	0.408
	Gem	H	0.290	0.024	0.210	0.499
	Qwn	H	0.439	0.061	0.277	0.503
	Apt	S	0.050	0.009	0.187	0.407
	Gem	S	0.387	0.032	0.216	0.505
	Qwn	S	**0.550**	**0.077**	**0.293**	**0.513**

Table 2: Model-implied conditional probabilities θ_{A-}, θ_{A+}, θ_{T-}, and θ_{T+} across all 27 RAG configurations, reported as posterior mean estimates. We show the highest value for each dataset in **bold** and the lowest in *italics.*

on retrieval success remains low across all models, indicating that unnecessary abstention is uncommon.

A second pattern is that retrieval success consistently improves answer success among non-abstained responses, but the magnitude of this improvement depends strongly on the dataset. On *FEV*, unsupported task success remains high even when retrieval fails, reflecting the relative ease of the task and the fact that each question has only two answer options. By contrast, on *HQA* and *NQ*, where retrieval success requires recovering all necessary documents and answers must exactly match an open-form reference, the gap between $P(T{=}1 \mid R{=}0, A{=}0)$ and $P(T{=}1 \mid R{=}1, A{=}0)$ is much larger.

The conditional decomposition also clarifies why similar marginal task success can hide substantial behavioral differences. On *NQ* with dense retrieval, the three generators achieve nearly identical task success rates (*Apertus*: 0.239, *Gemma3*: 0.242, *Qwen3.5*: 0.243), yet their policy adherence differs sharply (*Apertus*: 0.164, *Gemma3*: 0.414, *Qwen3.5*: 0.496). In other words, comparable end-task accuracy does not imply comparable behavior under retrieval failure. The conditional probabilities reveal that these differences are largely driven by the abstention policy rather than by answer accuracy alone.

Finally, improved retrieval does not automatically translate into proportional gains in either task success or policy adherence. Better retrieval increases the opportunity to answer correctly, but the realized benefit depends on whether the generator both recognizes retrieval failure and uses retrieved support effectively when it is available.

5.3 The Sample Allocation Problem

Here, we investigate a setting closer to what one might encounter in real-world applications, where annotation scarcity is common. Thus, we assume access to 100 annotations (*Base* samples) for retrieval and task success, and that abstention is always observed (via simple string matching). Then, we assume that we are given an additional budget for more annotations (*Add.* samples). The question is where to allocate the additional samples: to measuring retrieval success, task success, or a mix of the two. We apply the model to 100 Base samples with full annotations for retrieval and task success, assuming that abstention is always observed via simple string matching. Given an additional annotation budget of *Add.* $\in 60, 100, 200, 500$ samples, we investigate five allocation strategies: (1) all additional samples receive full annotations (i.e., both task-success and retrieval-success annotations), (2) half receive full annotations and half only retrieval-success annotations, (3) half receive full and half only task-success annotations, (4) all additional samples receive only retrieval-success annotations, and (5) all additional samples receive only task-success annotations. To ensure robust estimates, we subsample 500 times from the 10,000 available samples and report the Mean Absolute Error (MAE) relative to the full-data point estimate, as well as the 95% credible interval width. We run experiments on the HQA dataset for Apertus and Qwen using the Hybrid retriever.

The Observation. Figure 2 reveals a consistent asymmetry across both models: retrieval-focused strategies reduce estimation error more effectively for policy adherence (G), while task-focused strate-

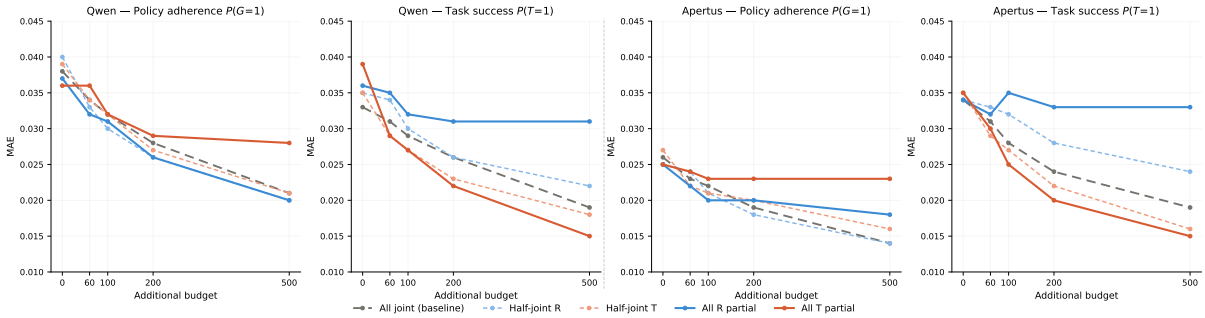

Figure 2: MAE for policy adherence $P(G{=}1)$ and task success $P(T{=}1)$ under five annotation allocation strategies, for Qwen (left) and Apertus (right). All configurations start with 100 fully annotated base samples. Abstention is always observed. Results on HotpotQA with Hybrid retrieval, averaged over 500 subsamples.

gies are more effective for task success (T). For $P(G{=}1)$, the all-R strategy matches or approaches the all-joint baseline for Qwen, whereas all-T shows markedly slower improvement and plateaus at a higher MAE. This pattern holds for Apertus, though the gap between all-R and all-joint widens at larger budgets. For $P(T{=}1)$, the pattern reverses: all-T closely tracks the all-joint baseline for both models, while all-R provides negligible improvement; its MAE and CI width remain nearly flat regardless of budget, particularly for Apertus. The half-joint strategies consistently fall between the corresponding extremes, with half-joint-R closer to all-joint for $P(G{=}1)$ and half-joint-T closer to all-joint for $P(T{=}1)$. Notably, the all-joint strategy never underperforms the best partial strategy by a large margin, making it a robust default when the estimation target is not known in advance.

The Explanation. To understand this asymmetry, we analyze the information gain of each strategy for estimating $P(G{=}1)$ (the case for $P(T{=}1)$ is straightforward, since task-success annotations directly observe T). Since the abstention A is always observed, we measure the conditional information gain $I(G; \cdot \mid A)$ of additionally observing R, T, or both. The key insight follows from the definition of G: of the four (R, A) cells, three resolve G deterministically, only $(R{=}1, A{=}0)$ requires knowing T (see Table 3a). Thus, observing R resolves G in three of four cases, while observing T resolves G only in one of three non-zero (A, T) cells. This structural asymmetry explains why retrieval annotations are consistently more informative for G than task annotations. Table 3b quantifies this using the conditional probabilities from Table 2 (full derivations in Appendix F). For both models, the All-R strategy captures the majority of the information provided by the All-Joint strategy (65.6% for Qwen, 58.0% for Apertus), whereas All-T captures far less (29.3% for Qwen, 40.8% for Apertus). The difference between models reflects their abstention behavior: Qwen's strong retrieval, abstention dependency ($P(A{=}1 \mid R{=}0) = 0.463$ vs. $P(A{=}1 \mid R{=}1) = 0.036$) means that R observations yield highly variable G labels, whereas Apertus rarely abstains regardless of retrieval ($P(A{=}1 \mid R{=}0) = 0.039$), reducing the discriminative value of R. The information-theoretic ranking correctly predicts the empirical ordering of partial strategies. The only discrepancy is that for Qwen, All-R slightly outperforms All-Joint at large budgets, despite lower per-sample information gain. This is because the 100 base joint samples already constrain $\theta_{T^+} = P(T{=}1 \mid R{=}1, A{=}0)$ in the single ambiguous cell, after which additional joint annotations provide diminishing returns (see Appendix F for details).

5.4 The Automated Judge

Since human annotations are highly time and cost-intensive, it has become common practice to use LLM-as-a-judge to automate parts of the evaluation. We investigate the impact of using automated judgments on the evaluation pipeline. The difficulty stems from the need to calibrate automated judgments to match human judgments (von Däniken et al., 2022), which introduces uncertainty that depends on the automated judge's performance.

The Judge. We use gpt-4o-mini (OpenAI et al., 2024) as our judge for both the task and retrieval success rates following the evaluation framework proposed by (Saad-Falcon et al., 2024) for context and answer relevance. We compare the retrieval success rates according to the judge $P(R_J = 1), P(T_J = 1)$ to those according to humans

R	A	G	Qwen	Apertus
0	1	1	0.367	0.031
0	0	0	0.426	0.762
1	1	0	0.007	0.001
1	0	T	0.200	0.206

(a) Joint probabilities $P(R, A)$ with the resulting value of G. In three of four cells, G is determined by (R, A) alone; only $(R{=}1, A{=}0)$ requires T.

Strategy	Qwen	Apertus
All-Joint	0.526	0.490
Half-Joint-R	0.436	0.387
All-R	0.345	0.284
Half-Joint-T	0.340	0.345
All-T	0.154	0.200

(b) Information gain $I(G; \cdot \mid A)$ per sample for each annotation strategy. Higher values indicate more informative observations for estimating $P(G{=}1)$.

Table 3: Information-theoretic analysis of annotation strategies on HotpotQA with Hybrid retrieval.

DS	Ret.	TPR	FPR	$P(R)$	$P(R_J)$
HQA	D	0.77	0.17	0.15	0.26
HQA	H	0.77	0.19	0.21	0.31
HQA	S	0.77	0.18	0.22	0.31

(a) Retrieval judge.

DS	Gen.	TPR	FPR	$P(T)$	$P(T_J)$
HQA	Qwn	0.94	0.32	0.21	0.52
HQA	Apt	0.88	0.43	0.21	0.57
HQA	Gem	0.93	0.38	0.21	0.55

(b) Task judge.

Table 4: LLM-as-a-judge calibration on HotpotQA. TPR and FPR denote the judge's true and false positive rates, e.g., TPR $= P(R_J{=}1 \mid R{=}1)$ and FPR $= P(R_J{=}1 \mid R{=}0)$ for retrieval. The judge exhibits high sensitivity but a substantial false-positive rate, leading to an overestimation of success rates.

$P(R = 1), P(T = 1)$. The judge's performance is measured in terms of true-and-false positive rates ($TPR = (R_J = 1 \mid R = 1)$, $FPR = P(R_J = 1 \mid R = 0)$). An analogous procedure is applied to task success, comparing T_J with T to derive the TPR and FPR. Tables 4a and 4b report calibration results for the retrieval and task judges on HotpotQA, showing that while both judges achieve high sensitivity, they exhibit substantial FPR, leading to overestimated success rates $P(R_J)$ and $P(T_J)$.

The Judge's Impact. We investigate whether supplementing the 200 human-annotated base samples with automated judge annotations improves estimation quality. We add $n_{\text{add}} \in \{0, 500, 5000\}$ judge-annotated samples to the base set, treating R_J and T_J as noisy observations of R and T with the calibrated TPR and FPR from Table 4. Table 5 reports results for Apertus on HotpotQA with Hybrid retrieval. Adding automated judgments yields only marginal improvements: the CI width for $P(G{=}1)$ decreases from 0.094 to 0.080 even with 5,000 additional samples, while MAE remains essentially unchanged. The pattern for $P(T{=}1)$ is similar. This is consistent with the judge's high false-positive rates (Table 4): when the FPR is substantial, each automated annotation carries limited information, and large volumes of noisy labels cannot substitute for even modest amounts of human annotation. The result highlights that the value of LLM-as-a-judge annotations depends critically on calibration quality. While our empirical findings are specific to the judges and calibration setup considered here, they illustrate that a poorly calibrated judge may add annotation volume without meaningfully reducing uncertainty. This is consistent with the findings of prior literature (von Däniken et al., 2022).

	P(G=1)		P(T=1)	
n_{add}	MAE	CI W.	MAE	CI W.
0	0.0174	0.0941	0.0232	0.1246
500	0.0166	0.0838	0.0243	0.1176
5000	0.0170	0.0802	0.0230	0.1150

Table 5: Mean absolute error and 95% credible interval width for P(G=1) and P(T=1) based on 200 fully annotated observations and additional observations where we observe R_J instead of R and T_J instead of T.

6 The Conclusion

We presented a Bayesian evaluation framework for RAG systems that factorizes the joint distribution over retrieval success, abstention, and task success according to the pipeline's information flow. The key distinction is between task success, i.e., whether the user received a correct answer, and generator success, i.e., whether the generator behaved appropriately given the retrieval outcome. Applying the model to 27 configurations, we showed that systems with near-identical task success can differ sharply in policy adherence, a distinction that marginal metrics conceal but the conditional decomposition makes explicit.

On the practical side, we analyzed the annotation allocation problem and demonstrated that

retrieval-success annotations are more informative than task-success annotations for estimating policy adherence. This asymmetry has a structural explanation: observing retrieval resolves generator success in three of four joint cells, whereas observing task success resolves only one. We further extended the model to incorporate LLM-as-a-judge annotations as calibrated noisy observations. Our results show that when the judge exhibits high false-positive rates, even thousands of automated annotations yield only marginal gains over a small set of human judgments, underscoring the importance of calibration quality.

More broadly, the framework illustrates how casting evaluation as probabilistic inference enables reasoning about uncertainty, partial observability, and annotation efficiency within a unified model. Natural extensions include continuous quality scales and more complex pipelines incorporating reranking, query reformulation, or iterative retrieval. Future work could also consider a hierarchical model that shares statistical strength across related datasets, retrievers, and generators, rather than treating each RAG configuration independently.

Limitations

Binary Model. The core limitation of this work is the simplified assumption that all judgments are provided on a binary scale (fail vs. success). Especially in Information Retrieval, the use of continuous metrics such as MAP and MRR is common and yields a more fine-grained model.

Deterministic Policy Definition. Generator success is defined by a single fixed policy (abstain iff retrieval fails, answer correctly otherwise). In practice, reasonable policies may differ; for instance, a system might legitimately attempt a partial answer when retrieval is incomplete, or abstention thresholds may be application-dependent. So far, the framework does not accommodate soft or alternative policy definitions.

Limited Task and Model Diversity. Due to budget limitations, the experiments cover three datasets (one fact-checking, two QA) and three relatively small open-weight generators (8–12B parameters).

Simple RAG Pipeline. The framework models a minimal retrieve-then-generate pipeline. Current production RAG systems often include additional components such as query reformulation, reranking, chunk filtering, or multi-turn retrieval. Each additional component introduces its own failure modes and dependencies that the current model does not capture. Extending the framework to deeper pipelines would require additional variables and a more complex dependency structure.

Single-Turn Evaluation. The framework evaluates each query in isolation as a single-turn interaction. It does not account for multi-turn conversational RAG settings, where retrieval and generation decisions depend on dialogue history, and where errors in earlier turns can compound across the conversation.

Abstention Detection. Abstention is detected through exact matching against the prescribed abstention response. This is appropriate in our constrained generation setting, where answers are restricted to a single entity, number, or fixed label, but would not capture hedged or indirect abstentions in free-form generation. Such settings could instead incorporate a calibrated abstention classifier as a noisy observation model.

Acknowledgments

This work was supported by the Swiss National Science Foundation (SNF) within the project "Unified Model for Evaluation of Text Generation Systems (UniVal)" [200020_219819].

References

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George Bm Van Den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego De Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, and 9 others. 2022. Improving language models by retrieving from trillions of tokens. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 2206–2240. PMLR.

Jiawei Chen, Hongyu Lin, Xianpei Han, and Le Sun. 2024. Benchmarking large language models in retrieval-augmented generation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(16):17754–17762.

Gordon V. Cormack, Charles L A Clarke, and Stefan Buettcher. 2009. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd International ACM SIGIR*

Conference on Research and Development in Information Retrieval, SIGIR '09, page 758–759, New York, NY, USA. Association for Computing Machinery.

Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. 2024. The faiss library.

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, St. Julians, Malta. Association for Computational Linguistics.

Alejandro Hernández-Cano, Alexander Hägele, Allen Hao Huang, Angelika Romanou, Antoni-Joan Solergibert, Barna Pasztor, Bettina Messmer, Dhia Garbaya, Eduard Frank Ďurech, Ido Hakimi, Juan García Giraldo, Mete Ismayilzada, Negar Foroutan, Skander Moalla, Tiancheng Chen, Vinko Sabolčec, Yixuan Xu, Michael Aerni, Badr AlKhamissi, and 82 others. 2025. Apertus: Democratizing Open and Compliant LLMs for Global Language Environments. https://arxiv.org/abs/2509.14233.

Matthew D. Hoffman and Andrew Gelman. 2014. The no-u-turn sampler: Adaptively setting path lengths in hamiltonian monte carlo. *Journal of Machine Learning Research*, 15(47):1593–1623.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

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.

Yuanjie Lyu, Zhiyu Li, Simin Niu, Feiyu Xiong, Bo Tang, Wenjin Wang, Hao Wu, Huanyong Liu, Tong Xu, and Enhong Chen. 2025. CRUD-RAG: A comprehensive chinese benchmark for retrieval-augmented generation of large language models. *ACM Trans. Inf. Syst.*, 43(2).

OpenAI, :, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mądry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, and 401 others. 2024. GPT-4o system card. *Preprint*, arXiv:2410.21276.

Fabio Petroni, Aleksandra Piktus, Angela Fan, Patrick Lewis, Majid Yazdani, Nicola De Cao, James Thorne, Yacine Jernite, Vladimir Karpukhin, Jean Maillard, Vassilis Plachouras, Tim Rocktäschel, and Sebastian Riedel. 2021. KILT: a benchmark for knowledge intensive language tasks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2523–2544, Online. Association for Computational Linguistics.

Qwen Team. 2026. Qwen3.5: Towards native multimodal agents.

David Rau, Hervé Déjean, Nadezhda Chirkova, Thibault Formal, Shuai Wang, Stéphane Clinchant, and Vassilina Nikoulina. 2024. BERGEN: A benchmarking library for retrieval-augmented generation. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 7640–7663, Miami, Florida, USA. Association for Computational Linguistics.

Jon Saad-Falcon, Omar Khattab, Christopher Potts, and Matei Zaharia. 2024. ARES: An automated evaluation framework for retrieval-augmented generation systems. 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 338–354, Mexico City, Mexico. Association for Computational Linguistics.

Hinrich Schütze, Christopher D Manning, and Prabhakar Raghavan. 2008. *Introduction to Information Retrieval*, volume 39. Cambridge University Press Cambridge.

Stan Development Team. 2026. Stan Reference Manual 2.38. https://mc-stan.org.

Saba Sturua, Isabelle Mohr, Mohammad Kalim Akram, Michael Günther, Bo Wang, Markus Krimmel, Feng Wang, Georgios Mastrapas, Andreas Koukounas, Andreas Koukounas, Nan Wang, and Han Xiao. 2024. jina-embeddings-v3: Multilingual embeddings with task LoRA. *Preprint*, arXiv:2409.10173.

Yixuan Tang and Yi Yang. 2024. Multihop-RAG: Benchmarking retrieval-augmented generation for multi-hop queries. *arXiv preprint arXiv:2401.15391*.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, and 197 others. 2025. Gemma 3 technical report. *Preprint*, arXiv:2503.19786.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018.

FEVER: a large-scale dataset for fact extraction and VERification. In *NAACL-HLT*.

TruLens. 2024. The RAG triad. https://www.trulens.org/getting_started/core_concepts/rag_triad/. Accessed: 2026-07-30.

Pius von Däniken, Jan Deriu, Don Tuggener, and Mark Cieliebak. 2022. On the effectiveness of automated metrics for text generation systems. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 1503–1522, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Pius von Däniken, Jan Milan Deriu, Alvaro Rodrigo, and Mark Cieliebak. 2024. Improving quantification with minimal in-domain annotations: Beyond classify and count. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 18, pages 1585–1598.

Shuai Wang, Ekaterina Khramtsova, Shengyao Zhuang, and Guido Zuccon. 2024. Feb4rag: Evaluating federated search in the context of retrieval augmented generation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 763–773.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Hao Yu, Aoran Gan, Kai Zhang, Shiwei Tong, Qi Liu, and Zhaofeng Liu. 2025. Evaluation of retrieval-augmented generation: A survey. In *Big Data*, pages 102–120, Singapore. Springer Nature Singapore.

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 Experimental Setup

A.1 Retrieval

Table 6 summarizes the retrieval configuration. All retrievers use top-$K = 5$.

A.2 Generation

A.3 Evaluation

No normalization, article stripping, or case adjustment is applied to generated answers prior to matching against the ground truth. Abstention is detected via exact string matching: for FEVER, the label `NOT_ENOUGH_INFO` is matched; for HotpotQA and NQ, the output `I DO NOT KNOW` is matched.

Sparse (BM25)	
Tokenizer	Whitespace
Stemmer	English
Stopwords	English
Min. DF	1
Lowercase	True
Ampersand norm.	True
Special char norm.	True
Acronym norm.	True
Punctuation removal	True
Dense	
Embedding model	`jinaai/jina-embeddings-v3`
Embedding task	`text-matching`
Retrieval task	`retrieval.query`
HNSW dim.	1024
HNSW M	32
HNSW metric	Inner product
Hybrid	
RRF k	1.0

Table 6: Retrieval hyperparameters.

Generator	
Temperature	0.7
Max tokens	512
Max concurrent requests	15
Thinking	None
LLM-as-a-Judge	
Model	`gpt-4o-mini-2024-07-18`
Max tokens	1
Temperature	1
Logprobs	True
Top logprobs	20

Table 7: Generation and judge hyperparameters.

B Generation Prompts

B.1 System Prompts

Depending on the task, a different system prompt is used.

B.1.1 Fact Checking

Used for the Fever task.

> You are a master fact checker. From given passages and a claim you can say whether the claim is: SUPPORTS, REFUTES.
>
> If the passages do not provide a clear answer you need to say: "NOT ENOUGH INFO".

Only answer with one of the given labels: SUPPORTS, REFUTES, NOT ENOUGH INFO.

B.1.2 QA

Used for both the HotpotQA and Natural Question task.

You are a precise question-answering assistant. You will be given a question and a list of retrieved paragraphs containing the answer. Your response must be ONLY the answer itself. A single word, name, entity, or number. No explanation, no punctuation, no full sentences, no preamble. Only use the retrieved paragraphs to answer the question! If the answer is not in the retrieved paragraphs you must say: I DO NOT KNOW

Examples:
Q: What nationality is Friedrich Merz?
A: German

Q: How many siblings did Marie Curie have?
A: Four

B.1.3 User Prompts

For all tasks the same user prompt is used. It consists of a list of paragraphs and the original query.

```
{% if retrieved|length > 0 %}
<retrieved>
    {% for p in retrieved %}
    <paragraph
      document_id="{{  p.document_id }}"
      index="{{  p.index }}"
    >
       {{  p.text }}
    </paragraph>
    {% endfor %}
</retrieved>
{% endif %}

<question>
    {{ input }}
</question>
```

C Partial Retrieval Success

In Section 3, we define retrieval success R as binary and measure downstream abstention and task behavior conditional on this binary outcome. In practice, however, a generation model may be able to produce a correct answer from only partial context [4]. Here, we add an experiment studying this setting. We define a ternary retrieval outcome $R \in$ fail, partial, success, where failure means that no relevant documents were retrieved, partial retrieval means that some but not all relevant documents were retrieved, and success means that all relevant documents were retrieved.

Dataset	Retriever	Partial retrievals, n (%)
FEVER	Dense	419 (4.19%)
	Hybrid	453 (4.53%)
	Sparse	300 (3.00%)
HotpotQA	Dense	4,724 (47.24%)
	Hybrid	5,647 (56.47%)
	Sparse	5,233 (52.33%)
NQ	Dense	0 (0.00%)
	Hybrid	0 (0.00%)
	Sparse	0 (0.00%)

Table 8: Number of partial retrievals for each dataset and retriever.

Table 8 shows the number and proportion of partial retrievals for each dataset and retriever. For *NQ*, the rate of partial retrieval is 0%, since each query in this dataset has exactly one relevant document. Similarly, the majority of *FEV* queries have exactly one relevant document, while a minority have more than one, leading to a maximum of 453 partial retrievals with the *Hybrid* retriever. For *HQA*, on the other hand, every query has two relevant documents by construction, since *HQA* is designed for multi-hop reasoning. Consequently, its partial retrieval rate reaches up to 56.47

We calculate the conditional decomposition for this ternary setup in Table 9. We define the conditional probability parameters analogously to Section 3: $\theta_{A,r} = P(A = 1 \mid R = r)$ and $\theta_{T,r} = P(T = 1 \mid A = 0, R = r)$, where $r \in f, p, s$ corresponds to failure, partial retrieval, and success, respectively. The results are overall consistent with the discussion in Section 5. In particular, the abstention rate decreases monotonically as retrieval quality increases, while the task success rate increases.

[4] This is in part due to relevance annotations not distinguishing which documents are individually sufficient or jointly necessary.

DS	Gen.	Ret.	$\theta_{A,f}$	$\theta_{A,p}$	$\theta_{A,s}$	$\theta_{T,f}$	$\theta_{T,p}$	$\theta_{T,s}$
HQA	Apt	D	0.064	*0.027*	0.009	*0.120*	0.313	*0.492*
	Gem		0.592	0.177	0.025	0.143	0.379	0.570
	Qwn		**0.812**	**0.373**	**0.044**	**0.272**	**0.502**	0.638
	Apt	H	*0.058*	0.031	*0.005*	0.127	*0.310*	0.519
	Gem		0.518	0.169	0.020	0.158	0.372	0.608
	Qwn		0.723	0.358	0.036	0.242	0.493	0.679
	Apt	S	0.068	0.028	*0.005*	0.146	0.321	0.553
	Gem		0.525	0.168	0.011	0.155	0.374	0.633
	Qwn		0.744	0.369	0.027	0.208	0.494	**0.705**

Table 9: Count-based conditional probabilities for HotpotQA treating retrieval outcome as ternary. We show the highest value in **bold** and the lowest in *italics.*

D LLM-as-a-Judge Prompts

The following prompts are used for the LLM-as-a-judge results.

D.1 System Prompts

System prompts differ between the two judges.

D.1.1 Retrieval Judge

> Given the following question and documents, you must analyze the provided documents and determine whether they are sufficient for answering the question. In your evaluation, you should consider the content of the documents and how they relate to the provided question. Output your final verdict by strictly following this format: "Yes" if the documents are sufficient and "No" if the documents provided are not sufficient. Do not provide any additional explanation for your decision.

D.1.2 Task Judge

> Given the following question, documents, and answer, you must analyze the provided answer and documents before determining whether the answer is relevant for the provided question. In your evaluation, you should consider whether the answer addresses all aspects of the question and provides only correct information from the documents for answering the question. Output your final verdict by strictly following this format: "Yes" if the answer is relevant for the given question and "No" if the answer is not relevant for the given question. Do not provide any additional explanation for your decision.

D.2 User Prompts

Both the retrieval and task judge use the same user prompt template. However the retrieval judge does not see the generated answer.

```
<context>
    {% for document in context %}
        <document>
            {{ document }}
        </document>
    {%  endfor %}
</context>

<question>{{ query }}</question>

{% if response|length > 0 %}
    <answer>{{ response }}</answer>
{% endif %}
```

E Extended Sample Allocation Plot

Figure 3 reports both MAE and 95% credible interval width for policy adherence $P(G{=}1)$ and task success $P(T{=}1)$ across all five annotation strategies, complementing the MAE-only summary in the main text (Figure 2). The credible interval width closely tracks the MAE ordering: for $P(G{=}1)$, retrieval-focused strategies yield narrower intervals than task-focused ones at every budget level, while for $P(T{=}1)$ the pattern reverses. For instance, at budget 500, the All-R strategy achieves a CI width of 0.105 for Qwen's $P(G{=}1)$ compared to 0.132 for All-T, whereas for $P(T{=}1)$ the All-T strategy reaches 0.075 versus 0.150 for All-R. Coverage remains close to the nominal 95% level across all strategies and budget levels, indicating that the posterior intervals are well-calibrated regardless of the allocation strategy. The all-joint strategy consistently yields among the narrowest intervals for both quantities, confirming its robust-

R	A	G	Qwen	Apertus
0	1	1	0.367	0.031
0	0	0	0.426	0.762
1	1	0	0.007	0.001
1	0	T	0.200	0.206

Table 10: Joint probabilities $P(R, A)$ for each cell of the (R, A) contingency table, with the resulting value of G. In three of four cells, G is fully determined by (R, A) alone. Only the cell $(R{=}1, A{=}0)$ leaves G unresolved, where $G = T$. Values shown for Qwen and Apertus on HotpotQA with Hybrid retrieval.

ness as a default when the estimation target is unknown in advance.

F Information Gain Calculations

We now analyze why certain annotation strategies are more effective than others for estimating policy adherence $P(G{=}1)$. Since abstention is always observed via string matching, we condition on A throughout and ask: how much does additionally observing R, T, or both reduce our uncertainty about G?

Joint Probability Structure. Recall that G is defined as:

$$G = (R{=}0 \wedge A{=}1) \vee (R{=}1 \wedge A{=}0 \wedge T{=}1).$$

Table 10 shows the four (R, A) cells and whether G is determined. In three cells, G follows directly from R and A—these correspond to cases where either retrieval or abstention went wrong. Only when both retrieval and abstention are correct, i.e., $(R{=}1, A{=}0)$, does G depend on T. This asymmetry is the structural basis for the annotation efficiency differences we observe.

Base Uncertainty. The uncertainty about G given only A is:

$$H(G \mid A) = \sum_a P(A{=}a) \cdot H(G \mid A{=}a). \quad (1)$$

When $A{=}1$ (the generator abstained), G depends only on whether the abstention was justified, i.e., whether $R{=}0$. Since R is unobserved:

$$H(G \mid A{=}1) = h\left(\frac{P(A{=}1, R{=}0)}{P(A{=}1)}\right), \quad (2)$$

where $h(\cdot)$ denotes the binary entropy function. For Qwen, $P(R{=}0 \mid A{=}1) = 0.367/0.374 = 0.981$, yielding $h(0.981) = 0.134$; for Apertus, $0.031/0.032 = 0.969$, yielding $h(0.969) = 0.196$.

When $A{=}0$ (the generator answered), $G{=}1$ requires both $R{=}1$ and $T{=}1$. With neither observed:

$$H(G \mid A{=}0) = h\big(P(G{=}1 \mid A{=}0)\big) \quad (3)$$

where

$$P(G{=}1 \mid A{=}0) = \frac{P(R{=}1)\, P(A{=}0 \mid R{=}1)\, \theta_{T^+}}{P(A{=}0)} \quad (4)$$

Remember $\theta_{T^+} = P(T = 1 \mid R = 1, A = 0)$.

For Qwen, $P(G{=}1 \mid A{=}0) = 0.216$, giving $h(0.216) = 0.760$; for Apertus, $P(G{=}1 \mid A{=}0) = 0.110$, giving $h(0.110) = 0.500$.

Combining both cases:

$$H(G \mid A) = P(A{=}1) \cdot h_{A=1} + P(A{=}0) \cdot h_{A=0}, \quad (5)$$

yielding 0.526 for Qwen and 0.490 for Apertus. Despite very different abstention behaviors, both models exhibit similar base uncertainty—though from different sources: for Qwen, a balanced split between $G{=}1$ and $G{=}0$ in the $A{=}0$ cell; for Apertus, the sheer mass of the $A{=}0$ cell (96.8% of samples) compensates for its lower per-sample entropy.

All-Joint Strategy. Observing both R and T resolves G completely, since G is a deterministic function of (R, A, T):

$$I(G; R, T \mid A) = H(G \mid A) - \underbrace{H(G \mid R, A, T)}_{=0} = H(G \mid A). \quad (6)$$

This represents the maximum achievable information gain per sample.

All-R Strategy. Observing only R yields:

$$I(G; R \mid A) = H(G \mid A) - H(G \mid R, A). \quad (7)$$

From Table 10, three of four (R, A) cells resolve G deterministically. Only $(R{=}1, A{=}0)$ leaves G unresolved, where $G = T$. The residual uncertainty is therefore:

$$H(G \mid R, A) = P(R{=}1, A{=}0) \cdot h(\theta_{T^+}). \quad (8)$$

For Qwen: $0.200 \times h(0.678) = 0.181$; for Apertus: $0.206 \times h(0.519) = 0.206$. The resulting gains are $I(G; R \mid A) = 0.345$ (Qwen) and 0.284 (Apertus). Observing R eliminates the majority of the uncertainty about G.

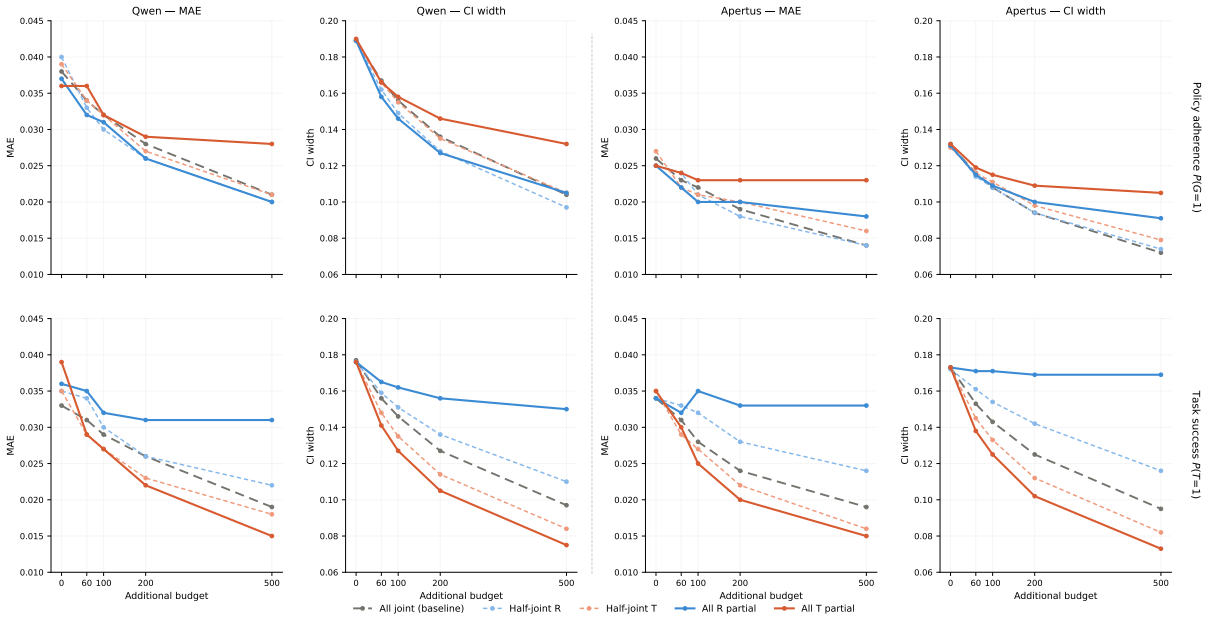

Figure 3: Estimation quality (MAE and 95% credible interval width) for policy adherence $P(G = 1)$ (top) and task success $P(T = 1)$ (bottom) under five annotation allocation strategies, for Qwen (left) and Apertus (right). All configurations start with 100 fully annotated base samples and increase the additional budget. Abstention is always observed. Results on HotpotQA with Hybrid retrieval, averaged over 500 subsamples.

All-T Strategy. Observing only T yields:

$$I(G;T \mid A) = H(G \mid A) - H(G \mid A, T). \quad (9)$$

Of the three non-zero (A, T) cells, only $(A{=}0, T{=}0)$ resolves G deterministically ($G{=}0$, since both branches of the definition fail). The remaining two cells require R:

- $(A{=}1, T{=}0)$: the generator abstained, but G depends on whether the abstention was justified, i.e., $H = h(P(R{=}0 \mid A{=}1))$.
- $(A{=}0, T{=}1)$: the generator answered correctly, but G depends on whether the answer was supported, i.e., $H = h(P(R{=}1 \mid A{=}0, T{=}1))$.

The residual uncertainty is:

$$\begin{aligned} H(G \mid A, T) = {} & P(A{=}1) \cdot h(P(R{=}0 \mid A{=}1)) \\ & + P(A{=}0, T{=}1) \\ & \cdot h(P(R{=}1 \mid A{=}0, T{=}1)), \end{aligned} \quad (10)$$

where $P(R{=}1 \mid A{=}0, T{=}1)$ is obtained via Bayes' rule:

$$\frac{P(R{=}1)\, P(A{=}0 \mid R{=}1)\, \theta_{T^+}}{P(A{=}0, T{=}1)}. \quad (11)$$

For Qwen: $P(R{=}1 \mid A{=}0, T{=}1) = 0.411$, yielding $H(G \mid A, T) = 0.372$ and $I(G;T \mid A) = 0.154$. For Apertus: $P(R{=}1 \mid A{=}0, T{=}1) = 0.352$, yielding $H(G \mid A, T) = 0.290$ and $I(G;T \mid A) = 0.200$.

Half-Joint-R Strategy. Here, half the additional budget is allocated to joint samples and the other half to retrieval-only samples. Each joint sample contributes $I(G; R, T \mid A) = H(G \mid A)$, while each R-partial sample contributes $I(G; R \mid A)$. The expected per-sample gain is the average:

$$\tfrac{1}{2}\, H(G \mid A) + \tfrac{1}{2}\, I(G; R \mid A), \quad (12)$$

yielding 0.436 for Qwen and 0.387 for Apertus. Since the R-partial component already captures most of the uncertainty (three of four cells resolved), the loss relative to the all-joint strategy is modest.

Half-Joint-T Strategy. Analogously, half the budget goes to joint samples and half to task-only samples. The expected per-sample gain is:

$$\tfrac{1}{2}\, H(G \mid A) + \tfrac{1}{2}\, I(G; T \mid A), \quad (13)$$

yielding 0.340 for Qwen and 0.345 for Apertus. Despite receiving the same number of joint samples as Half-Joint-R, this strategy is less effective because the T-partial component contributes less information about G—it leaves two cells unresolved rather than one.

Summary. Table 11 summarizes the information gains. The structural asymmetry is clear: observing R leaves one ambiguous cell for G, while

observing T leaves two. This directly explains why retrieval-focused strategies outperform task-focused strategies for estimating policy adherence. The mixed strategies interpolate linearly between these extremes.

Discussion. The information-theoretic ranking in Table 11 correctly predicts the relative ordering of partial strategies: All-R consistently outperforms All-T for estimating $P(G{=}1)$, and the half-joint strategies fall between the corresponding extremes. However, the predicted ranking does not perfectly match the empirical results at all budget levels. For Apertus, the empirical MAE ordering at budget 500 follows the theoretical ranking closely: All-Joint $\approx$ Half-Joint-R $<$ Half-Joint-T $<$ All-R $<$ All-T. For Qwen, however, All-R and Half-Joint-R slightly outperform All-Joint despite having lower per-sample information gain.

This discrepancy arises because the information-theoretic analysis treats each sample independently, whereas the Bayesian model shares information across cells through its parameterization. The only cell left unresolved by (R, A) observations is $(R{=}1, A{=}0)$, where $G = T$. The 100 base joint samples provide direct observations of T in this cell, allowing the model to estimate θ_{T^+}. Once this parameter is sufficiently well-estimated, additional joint samples yield diminishing returns—they annotate T in cells where G is already resolved by (R, A) alone. R-partial samples, by contrast, contribute exclusively to the three resolved cells, where each observation provides a clean binary label for G.

For Apertus, $\theta_{T^+} = 0.519$ with near-maximal entropy ($h(0.519) = 0.999$), making it intrinsically harder to estimate. The 100 base samples are insufficient to pin it down, and additional joint samples that directly observe T in the ambiguous cell remain valuable. For Qwen, $\theta_{T^+} = 0.678$ with lower entropy ($h(0.678) = 0.904$), allowing the base samples to estimate it more effectively and reducing the marginal value of additional joint annotations.

In summary, the information-theoretic analysis reliably predicts *which* partial observation is more informative for a given target, while the question of whether partial annotations can fully *substitute* for joint annotations depends on how well the base samples constrain the parameters in the ambiguous cells.

Strategy	Qwen	Apertus
All-Joint	0.526	0.490
Half-Joint-R	0.436	0.387
All-R	0.345	0.284
Half-Joint-T	0.340	0.345
All-T	0.154	0.200

Table 11: Information gain $I(G; \cdot \mid A)$ per sample for each annotation strategy, computed from the conditional probabilities in Table 2. Higher values indicate more informative observations for estimating $P(G{=}1)$. The ranking is consistent with the empirical MAE ordering in Figure 2.

G AI Assistants

AI assistants (Claude, Anthropic) were used for code development, data analysis, and iterative drafting of manuscript text. All outputs were reviewed and validated by the authors.