Attribution in Practice: A Human-Grounded Evaluation of RAG Citation Methods

Anonymous ACL submission

Abstract

Retrieval-Augmented Generation (RAG) relies on accurate answer attribution to verify whether generated claims are grounded in retrieved evidence, yet a comprehensive evaluation of attribution methods remains missing. We present a large-scale evaluation of six context attribution methods across five large language models, four datasets, and three evidence selection strategies using a multi-axis framework that combines automated metrics, computational efficiency profiling, and human evaluation ($N = 300$). Results reveal stark differences: white-box methods (HSSAVG, TokenShapley) excel across all dimensions, with AttnUnionDep achieving the highest human agreement. Conversely, perturbation-based black-box methods (ContextCite, AttriBoT) show weaker performance and higher computational costs, while MIRAGE faces severe limitations in long-context settings. Furthermore, LLM-as-a-Judge (LaaJ) strongly correlates with human judgments, proving an effective automated proxy.

1 Introduction

Retrieval-Augmented Generation (RAG), which combines real-time knowledge retrieval with Large Language Models (LLMs), is increasingly deployed in high-stakes information extraction and reasoning tasks where the consequences of misinformation are severe, such as healthcare and law.

However, retrieved evidence alone does not guarantee correct grounding, as RAG systems may ignore, misinterpret, or hallucinate support from retrieved documents during generation (Magesh et al., 2025; Niu et al., 2024; Wan et al., 2024). Consequently, there is a critical need for mechanisms that enable humans to rapidly validate system outputs. In response, *context attribution* has emerged as the dominant explainability paradigm for RAG systems (Radevski et al., 2025), aiming to identify the exact retrieved evidence, such as source spans or passages, that directly supports or contributes to a generated claim (Hagström et al., 2025; Liu et al., 2023; Sun et al., 2026).

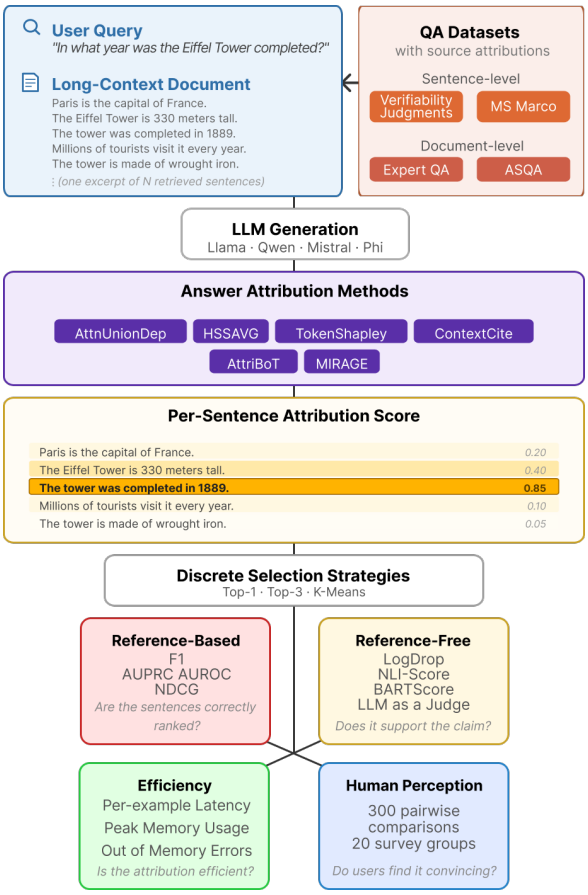

Figure 1: Overview of our multi-dimension evaluation study comparing answer attribution methods across (1) Reference-based, (2) Reference-free, (3) Efficiency, and (4) Human Perception. They are commonly used to explain a language model's decision-making process in RAG.

Recently, numerous context attribution methods have been developed to systematically trace generated statements back to their originating source documents (Cohen-Wang et al., 2024; Qi et al., 2024; Phukan et al., 2024; Ding et al., 2025; Xiao et al., 2025; Liu et al., 2025; Saxena et al., 2025; Li et al., 2025). Comparisons between newer methods

and earlier baselines are limited to a small scope, so a comprehensive evaluation of their practical applicability, combining automatic performance metrics, human evaluation, and analyses of computational time and resource efficiency, remains missing.

The latter is particularly relevant for long-context documents, which have shown strong potential to improve performance across many tasks by providing richer contextual information (Sun et al., 2026). At the same time, explainability methods often introduce substantial computational overhead, limiting their scalability in practical deployments (Das and Rad, 2020; Jiang et al., 2025). As document length increases, the computational cost of context attribution methods may grow linearly or even exponentially (Rawte et al., 2025; Mohammadi et al., 2026), making efficiency a critical consideration for real-world adoption.

To address these challenges in verifiable generation, we present a multi-dimensional benchmark of context attribution methods on long-context datasets, combining algorithmic profiling with human-grounded evaluation. Unlike prior surveys (Worledge et al., 2024; Sun et al., 2026) that taxonomize attribution methods or evaluate them on short inputs, our work provides the first large-scale empirical comparison that integrates ranking accuracy, corroborative quality, computational efficiency, and human judgment under realistic long-context conditions. Our code and data are publicly available.[1] Our main contributions are threefold:

- First, we systematically evaluate six state-of-the-art attribution methods (spanning white-box, perturbation-based, and surrogate paradigms) on QA datasets of varying document length, providing empirical insights into the impact of different passage selection strategies on faithfulness and citation correctness.

- Second, we map the Pareto frontier of RAG attribution by profiling the inference latency and memory constraints of each method alongside task performance. This analysis yields an actionable decision matrix for practitioners, balancing high-fidelity attribution against closed-model API constraints and computational overhead.

- Finally, we bridge the inherent disconnect between system-centric mathematical metrics and actual user trust. Through a rigorously controlled, pairwise human evaluation ($N = 300$), we measure how algorithmic extraction choices translate to human-perceived verifiability, ultimately revealing which automated metrics serve as the best proxies for real-world user preference.

[1] https://github.com/heaarag/heaarag

2 Background

2.1 Attribution Methods

Attribution methods can be differentiated based on the granularity at which evidence is associated with generated outputs.

Document-level attribution links responses to predefined text documents, which may consist of full texts, paragraphs, sentences, or smaller text spans. Methods such as AttnUnionDep (Ding et al., 2025) and ContextCite (Cohen-Wang et al., 2024) operate at this level by identifying or perturbing relevant supporting documents to estimate their influence. Token-level attribution provides granularity by assigning evidence or contribution signals to individual generated tokens or phrases. Token-Shapley (Xiao et al., 2025) and HSSAVG (Phukan et al., 2024), for example, leverage internal model states to estimate the influence of contextual information on specific generated tokens natively. Although these methods natively operate at varying granularities, we standardize our benchmark by aggregating their extracted signals into a unified sentence-level resolution, strictly following the aggregation methodologies proposed by the authors in their respective original papers (as detailed in Section 3).

Attribution approaches can additionally be categorized according to the level of access required to the underlying model. White-box methods require access to internal model representations, such as attention scores, hidden states, or gradients, to estimate evidence contribution. Approaches including MIRAGE (Qi et al., 2024), HSSAVG (Phukan et al., 2024), AttnUnionDep (Ding et al., 2025), and TokenShapley (Xiao et al., 2025) belong to this category. In contrast, black-box methods rely solely on model inputs and outputs, typically using perturbation or counterfactual strategies to estimate evidence importance without accessing model internals. ContextCite (Cohen-Wang et al., 2024) and AttriBoT (Liu et al., 2025) exemplify black-box approaches. Beyond these categories, some methods introduce attribution through architectural

modifications or task-specific adaptation, such as changes to the prompt design (Weller et al., 2024) or attribution-oriented fine-tuning (Huang et al., 2024).

2.2 Attribution Evaluation

Worledge et al. (2024) distinguish between **contributive** ("*Did the context physically cause the model to generate the text?*") and **corroborative** attribution ("*Does the context logically support the text (regardless of model mechanics)?*"). While their survey effectively taxonomizes existing methods, they highlight a critical gap: the lack of standardized, human-aligned evaluation protocols. Our work directly addresses this gap by providing a large-scale, human-grounded comparison of answer attribution methods.

Related to evaluation design, Sun et al. (2026) created an evaluation framework for assessing highlight-style explanations, finding that traditional feature attribution methods often misalign with human judgments of contextual relevance. In contrast, our study focuses on full answer attribution rather than localized highlights, and we benchmark a broader spectrum of techniques against human preferences.

Human evaluation remains central to assessing attribution reliability. Xing et al. (2025) conducted a human study on evidence attribution for fact-checking explanations, revealing that post-hoc selection methods frequently disagree with ground truth. However, their setup is restricted to a single domain and post-hoc selection paradigms. We extend this line of work by evaluating attribution across multiple QA domains and comparing a wider range of methods. Finally, Narayanan Venkit et al. (2025) evaluate LLM-based search systems and conclude that current citation mechanisms lack the transparency and reliability required for safe public deployment. Our benchmark directly responds to this concern by identifying which attribution methods consistently align with human judgment, thereby providing actionable guidelines for building trustworthy, citation-aware RAG systems.

3 Experimental Setup

For a full systematic evaluation of the previously listed context attribution methods, we run them across four long-document datasets using a variety of LLMs. We evaluate results using human evaluation, reference-based and reference-free automatic performance measures, as well as efficiency tracking.Comprehensive details regarding the global pipeline configurations, and method-specific hyperparameters are provided in Appendix J.

Approach	Based on	Granularity
MIRAGE (Qi et al., 2024)	Gradient	Token
ContextCite (Cohen-Wang et al., 2024)	Perturbation	Document
HSSAVG (Phukan et al., 2024)	Hidden-States	Token
AttnUnionDep (Ding et al., 2025)	Attention	Document
AttriBoT (Liu et al., 2025)	Perturbation	Document
TokenShapley (Xiao et al., 2025)	Hidden-States	Token

Table 1: Comparison of answer attribution methods for RAG based on method type and attribution granularity.

3.1 Datasets

We evaluate our methods on four question answering datasets with varying levels of attribution granularity: *Verifiability Judgments* (VJ) (Liu et al., 2023) and *MS MARCO* (MS)(Nguyen et al., 2016) provide sentence-level attribution scores, while *ExpertQA* (EQA) (Malaviya et al., 2024) and *ASQA* (Stelmakh et al., 2022) provide document-level attribution.

Throughout our experiments, we focus exclusively on **sentence-level attribution**. Document-level attribution datasets, which lack golden token or sentence attribution scores, are evaluated by reference-free metrics only.

Since these datasets rely on external web sources, we apply a unified data curation pipeline to ensure that all referenced documents are accessible, contain sufficient textual content, and remain consistent with their associated annotations. Details on the datasets and the pipeline can be found in Appendix B.

3.2 Model Selection

We selected a diverse suite of state-of-the-art open-weight models, ensuring sufficient context capabilities and robust coverage across different architectures and parameter scales. Specifically, our evaluation suite includes **Llama-3.1-8B** (Grattafiori et al., 2024), **Mistral-7B** (Jiang et al., 2023), **Qwen2.5** (both **7B** and **32B** variants) (Team, 2024), and **Phi-3-Medium** (Abdin et al., 2024).

3.3 Evaluation Metrics

3.3.1 Score Aggregation Protocol

To standardize attribution outputs to a sentence-level resolution, we uniformly distribute document-level scores to their constituent sentences. For token-level methods, we aggregate token scores

into a single sentence score by strictly following the algorithms proposed in their original papers.

3.4 Human Evaluation

Our human annotation study was conducted exclusively on Llama-3.1-8B generated outputs for these datasets.

We sampled 300 prompts from datasets VJ and MS. For each prompt, one pair of methods was drawn randomly and presented side by side to the annotator, allowing a direct comparison. The study was conducted via Prolific as a crowdsourcing platform. Attention check questions were embedded throughout the survey to ensure annotator engagement and response quality.

To evaluate the six explainability methods, we conducted a human evaluation study along four dimensions: Usefulness, Understandability, Trustfulness, and Performance defined by Lopes et al. (2022). The first three dimensions were assessed using 7-point Likert scales, while Performance was evaluated through pairwise comparisons.

Each annotation task presented the annotator with a chat interface showing a query and the corresponding model output generated by Llama 3.1 8B. Below the chat interface, two columns displayed the outputs of the two different explainability methods side by side. Each column showed a retrieved document in which the most relevant sentence was highlighted, following a top-1 selection strategy to avoid cognitive overload. Rather than displaying the full document, only the highlighted sentence along with 2 preceding and 2 following sentences were shown by default; annotators could optionally expand the view to read the full document via a dedicated button. Based on this interface, annotators first judged which of the two explainability outputs was more performant – option A, option B, or both equally (see Figure 2). In a subsequent step, each annotator was then shown either option A or option B in isolation and asked to rate it along the remaining three dimensions – Usefulness, Understandability, and Trustfulness – using Likert scales. The full experiment interface is provided in Appendix D.

3.4.1 Reference-Based Evaluation

Reference-based evaluation measures the method's ability to identify ground-truth evidence provided in the datasets. To evaluate the raw, continuous scoring power of the algorithms, we use *AUROC*, *AUPRC*, and *NDCG*. Specifically, AUROC measures the attribution method's overall ability to discriminate between gold evidence and irrelevant context across all decision thresholds (Hanley and McNeil, 1982). Since long-context retrieval inherently creates severe class imbalance (few gold sentences amidst massive noise), AUPRC serves as our primary metric by summarizing the precision-recall trade-off and providing a robust point estimate for ranking performance (Boyd et al., 2013). Finally, NDCG evaluates the strict ranking quality of the continuous attribution scores, penalizing models that place highly relevant supporting sentences lower in the output order (Järvelin and Kekäläinen, 2002).

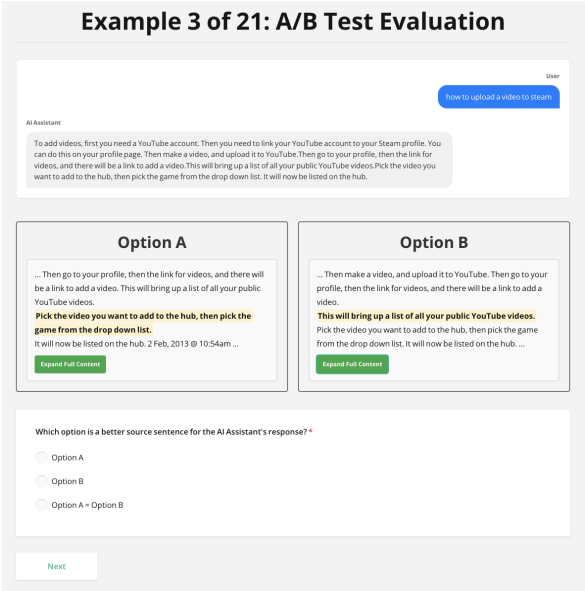

Figure 2: A/B Test Evaluation interface showing the side-by-side comparison of attribution methods.

To evaluate discrete subsets of extracted evidence, we employ the F_1 score. To binarize the continuous scores into a discrete subset, we apply three distinct selection strategies: Top-1, Top-3, and K-Means($K = 2$).

3.4.2 Reference-Free Evaluation

Reference-free evaluation assesses the corroborative quality and fluency of the attributions, focusing on whether the selected content logically supports the claim rather than merely matching ground-truth labels.

Following foundational citation benchmarks like ALCE (Gao et al., 2023), we utilize NLI Score (Honovich et al., 2022), BARTScore (Yuan et al., 2021), and LogDrop (Cohen-Wang et al., 2024). The LogDrop metric measures the marginal log-likelihood contribution of the retrieved evidence to

the generated response, calculated via the FLAN-T5 model (Chung et al., 2022). Furthermore, specifically for the MS and VJ datasets, we incorporate an LLM-as-a-judge (LaaJ) metric, leveraging Llama-3.3-70B-Instruct to perform pairwise comparative assessments across all models, providing a scalable proxy for attribution quality. The prompt instruction follows the human evaluation guidelines.

Our benchmark relies on datasets with varying levels of attribution granularity. For datasets without sentence-level ground truth (ASQA, EQA), evaluation is limited to reference-free metrics.

3.4.3 Efficiency

To thoroughly evaluate the operational viability of each attribution method, we systematically recorded execution latency, peak memory utilization, and the frequency of out-of-memory (OOM) failures. Execution latency is measured as the per-example processing time (seconds per query) as well as the absolute total execution time per dataset. For memory, we captured the peak VRAM allocation required by each method during its execution cycle. Finally, we explicitly tracked OOM events – instances where memory demands exceeded our hardware thresholds – as these represent critical bottlenecks for the scalability and production feasibility of attribution methods.

4 Results

4.1 Multi-dimensional Performance Overview

To understand the multi-dimensional strengths and weaknesses of each attribution method, Figure 3 illustrates their comprehensive performance profiles. Rather than relying on a single evaluation axis, this visualization maps each method across our expanded evaluation suite (15 distinct metrics). It simultaneously compares resource efficiency (Runtime, Peak Memory, OOM Rate), reference-based automatic evaluation (AUPRC, AUROC, NDCG, F_1), reference-free automatic evaluation (BARTScore, NLI-Score, LogDrop, LAAJ Score), and human-centric utility (Performance, Understandability, Usefulness, Trust). Given the varying applicability of certain metrics across different models and datasets, a comprehensive scope matrix detailing exactly which data subsets are utilized in each evaluation table is provided in Appendix Table 9.

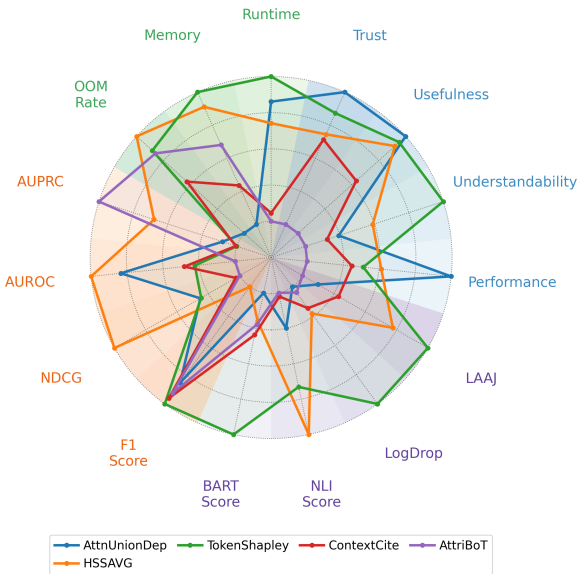

Figure 3: Radar chart comparing the multi-dimensional performance of top attribution methods across all automated, human, and resource metrics. Scores are mathematically normalized for radial plotting.

4.2 Reference-Based Evaluation

Table 2 presents aggregated performance across all datasets. At the aggregate level, the **Shared Metrics** (AUPRC, AUROC, NDCG), which are calculated independent of any selection strategie, as well as the **F1** scores across all selection strategies, remain highly competitive among the top-performing methods. It should be noted that the ASQA and EQA datasets are completely excluded from these evaluations, as they provide only document-level annotations rather than the sentence-level granularity required for reference-based evaluation.

4.3 Reference-Free Evaluation

The most substantial differentiators between these methods emerge in **BART, NLI, LogDrop** and the **LaaJ Score**. *TokenShapley* achieves the highest macro-average LaaJ ranking (0.1779) as shown in Table 3. Furthermore, it achieves the best BART and LogDrop results universally (e.g., Top-1 LogDrop of 1.393 and KMeans LogDrop of 1.652).

Observation of unaggregated cross-model data reveals differing levels of stability among the methods. While *TokenShapley*, *ContextCite*, and *AttriBoT* display stable scores across different LLMs, *AttnUnionDep* exhibits high cross-model variance (see Appendix Table 11).

The full unaggregated performance matrices for all datasets are provided in the Appendix (Tables 13, 12), 14) and 15)

Method	Shared Metrics			Top-1				Top-3				KMeans			
	AUPRC	AUROC	NDCG	F1	BART	NLI	LogDrop	F1	BART	NLI	LogDrop	F1	BART	NLI	LogDrop
AttnUnionDep	0.519	0.723	0.658	0.513	-2.839	0.262	1.308	0.465	-2.475	0.483	1.610	0.551	-2.505	0.441	1.569
AttriBoT	**0.546**	0.648	0.653	0.537	-2.784	0.266	1.339	0.443	-2.461	0.467	1.597	0.578	-2.500	0.405	1.561
ContextCite	0.516	0.680	0.654	**0.545**	-2.762	0.276	1.359	0.448	-2.445	0.469	1.611	**0.580**	-2.514	0.398	1.554
HSSAVG	0.534	**0.744**	**0.669**	0.526	-2.775	0.282	1.347	0.459	-2.512	0.465	1.579	0.285	-2.467	**0.587**	1.609
MIRAGE	0.382	0.643	0.559	0.371	-3.108	0.190	1.101	0.405	-2.546	0.407	1.558	0.393	-2.947	0.245	1.228
TokenShapley	0.516	0.674	0.658	0.545	**-2.709**	**0.283**	**1.393**	**0.466**	**-2.393**	**0.493**	**1.657**	0.578	**-2.374**	0.491	**1.652**

Table 2: Performance metrics averaged across all datasets and models.

#	Method	Wins	Ties	Losses	Avg Score
1	TokenShapley	22130	1244	15122	0.1779
2	HSSAVG	23290	1289	13917	0.1372
3	ContextCite	18599	1442	18455	0.0771
4	AttnUnionDep	18998	1174	18324	0.0550
5	AttriBoT	18034	1349	19113	0.0389
6	MIRAGE	11460	1058	25978	-0.4526

Table 3: Overall LLM-as-a-Judge rankings. The score represents the macro-average of win-rates across all dataset and model combinations, ensuring equal weight regardless of sample size.

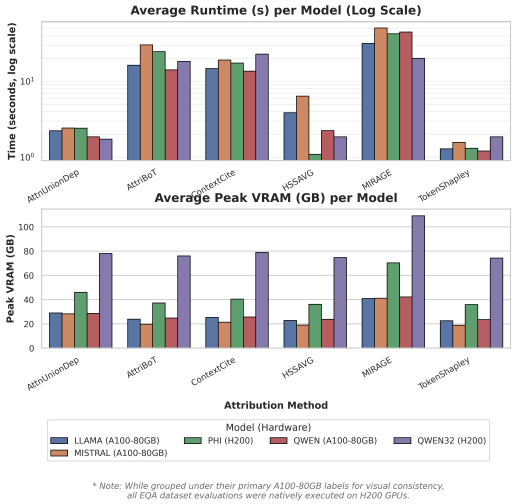

Figure 4: Average execution runtime and peak VRAM allocation, aggregated across all evaluated datasets.

4.4 Efficiency Evaluation

MIRAGE and *AttnUnionDep* exhibited significantly higher failure rates, predominantly due to OOM errors. A comprehensive reliability analysis and dataset-level failure matrices are provided in Appendix H (Figure 12).

Regarding execution speed (Figure 4, top), *TokenShapley* processes queries in $\sim$1 second. *AttnUnionDep* and *HSSAVG* also execute rapidly ($<$4 seconds), while *MIRAGE* requires 9–71 seconds depending on the model and dataset.

For peak VRAM (Figure 4, bottom), *MIRAGE* pushes the QWEN32 model from a $\sim$70 GB baseline to nearly 115 GB. Detailed dataset-level hardware tables are available in Appendix H.

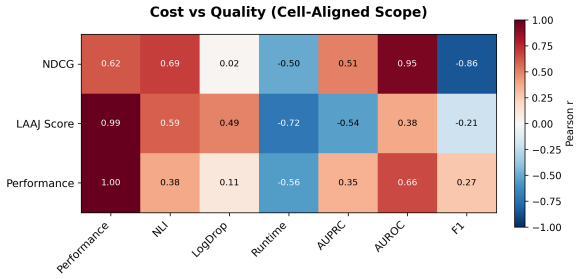

Figure 5: Pearson correlation matrix detailing Cost vs Quality relationships, calculated using a Cell-by-Cell Scope Alignment to eliminate sample mismatch.

4.5 Correlation Analysis

To isolate intrinsic metric relationships, we performed a Pearson correlation analysis using a Cell-by-Cell Scope Alignment (Figure 5), excluding resource-heavy outliers like *MIRAGE*. The matrix reveals that extended Runtime correlates negatively with qualitative outcomes like Usefulness (-0.71) and Performance (-0.56). Conversely, Peak Memory correlates positively (0.53) with human-evaluated Performance, an observation primarily driven by *AttnUnionDep*.

The correlation matrix exposes a near-perfect alignment ($r = 0.99$) between the automated *LAAJ Score* and human-annotated *Performance*. When examining continuous ranking metrics within the reference-based evaluation, *AUROC* (0.66) and *NDCG* (0.62) exhibit the strongest positive correlations with human *Performance*. Additionally, automated quality metrics exhibit internal alignments (e.g., NLI and NDCG at 0.69, and NLI and LogDrop at 0.50).

For the comprehensive square correlation matrix encompassing all automated and human evaluation metrics, we refer the reader to Appendix F (Figure 10).

#	Method	Wins	Ties	Losses	Score
1	AttnUnionDep	57	27	16	0.4100
2	HSSAVG	42	31	27	0.1500
3	TokenShapley	44	26	30	0.1400
4	ContextCite	41	30	29	0.1200
5	AttriBoT	32	31	37	-0.0500
6	MIRAGE	4	15	81	-0.7700

Table 4: Performance ranking of answer attribution methods by pairwise comparisons.

4.6 Scatter Analysis

Projecting multi-dimensional scatter plots (Figure 6) delineates the boundaries of method capabilities.

In the evaluation of human preference versus its automated proxy (Performance vs. LaaJ Score), *AttnUnionDep* is positioned in the upper-right quadrant across multiple models.

The metric space for Runtime vs. LaaJ Score highlights *TokenShapley* in the ultra-low-latency regime ($<$ 1s) with high LaaJ scores. *HSSAVG* yields competitive LaaJ scores with moderate speeds. *ContextCite* and *AttriBoT* cluster on the slower end of the spectrum ($>$ 5s).

Mapping ranking accuracy against perceived utility (AUROC vs. LAAJ Score) shows *ContextCite, AttriBoT* and *TokenShapley* maintaining a stable frontier. *AttnUnionDep* and *HSSAVG* scatters broadly along the AUROC axis depending on the underlying LLM architecture.

4.7 Human and LaaJ Alignment

We conducted pairwise comparisons using both human annotators (Table 4) and an LaaJ setup (Table 3), with human evaluation covering only a subset of the data points. *MIRAGE* ranks last in both environments. *HSSAVG* and *TokenShapley* secure top-tier positions, ranking 2nd and 3rd by human consensus, and 1st and 2nd by the LLM judge.

A point of divergence is *AttnUnionDep*, which ranks first in the constrained human trials but places fourth under the macro-averaged LLM judge. Detailed per-dataset and per-model matrices for both evaluation paradigms are provided in the Appendix (Tables 16, 8, and 7).

5 Discussion

White-box and lightweight methods dominate the cost–quality trade-off. Taking into consideration all evaluation metrics, we observe that *AttnUnionDep* performs best in human evaluation but suffers in automatic evaluation and incurs the highest memory consumption. *HSSAVG* and *TokenShapley* are highly efficient and yield strong evaluation results across all dimensions. Conversely, *AttriBoT* and *ContextCite* perform worse than the other methods across most dimensions, while *MIRAGE* is entirely unsuitable for long-document question answering, ranking last on every axis. With regard to the method category, perturbation-based models tend to perform worse than gradient or internals-based white-box methods. *TokenShapley* serves as a unique bridge: while it relies on Shapley values (often associated with perturbation), it operates natively within the hidden-state space, allowing it to bypass the massive latency penalties of pure black-box perturbation methods.

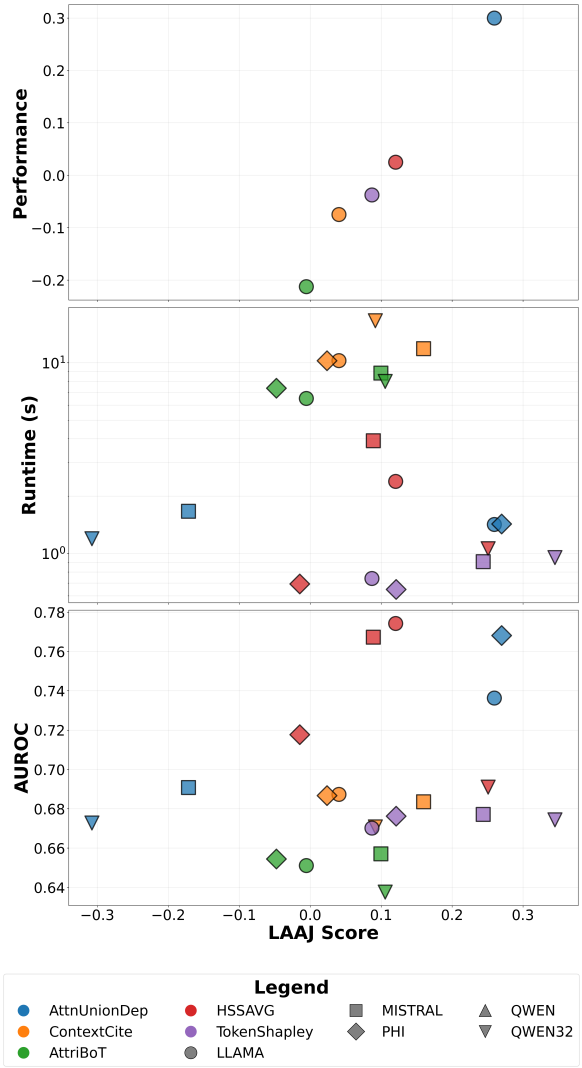

Figure 6: Multi-dimensional scatter analysis plotting Performance vs. LAAJ Score (Top), Runtime vs. LAAJ Score (Middle), and AUROC vs. LAAJ Score (Bottom).

Strong human preference can mask cross-model instability. Having a closer look at the correlation between human and automatic evaluation reveals a critical nuance. In our strict scope alignment (using only Llama generations on the MS and VJ datasets), *LaaJ Score* and human *Performance* exhibited a near-perfect correlation ($r = 0.99$). In this specific, constrained environment, *AttnUnionDep* emerged as the top-ranking method for both humans and the LLM judge. However, when examining the *macro-average* LaaJ rankings across *all* six evaluated LLMs (Table 3), *AttnUnionDep* drops to fourth place. This discrepancy is entirely driven by *AttnUnionDep*'s severe cross-model instability (as documented in Appendix Table 11). While it extracts highly convincing, human-preferred citations when applied to Llama, its performance fluctuates wildly when applied to different architectures (e.g., Mistral or Qwen). In contrast, methods like *Token-Shapley* and *HSSAVG* maintain robust, consistent performance regardless of the underlying LLM, making them safer choices for generalized deployment despite *AttnUnionDep*'s isolated peak performance.

Discrete selection masks differences between methods. Attribution methods produce continuous per-sentence scores, but downstream systems only see the output of the discrete selection step. Two methods that disagree sharply on raw scores can nonetheless select the same sentences once Top-K or thresholding is applied, masking quality gaps that are real at the continuous level. When comparing *Top-N* strategies against dynamic clustering (*K-Means*), performance diverges sharply depending on the attribution method's underlying score distribution. Most notably for *HSSAVG*, *K-Means* produces substantially lower F_1 than the threshold-based alternatives. This occurs because *HSSAVG* scores are densely distributed, causing *K-Means* to group an excessively large number of sentences into the top cluster. This over-selection severely penalizes precision (F_1) while artificially rewarding recall-oriented metrics like NLI-Score.

Top-K and cluster-based strategies trade precision for recall. Aggressive *Top-1* selection optimizes strict precision and yields the highest median F_1, but at the cost of evidence coverage. *K-Means* and *Top-3*, which surface multiple candidates, sacrifice F_1 in exchange for higher entailment-based scores. The appropriate operating point therefore depends on whether the downstream consumer is a strict citation interface (favoring Top-1 to minimize cognitive load) or a permissive entailment-checking reranker (favoring multi-sentence selection to ensure no context is missed).

6 Conclusion

We have presented a multi-axis benchmark of context attribution methods for RAG, combining automatic metrics, efficiency profiling, and a controlled pairwise human evaluation. Our study demonstrates that lightweight attribution methods consistently dominate the cost–quality Pareto frontier, delivering strong automatic and human evaluation scores at a fraction of the computational cost incurred by heavyweight perturbation-based approaches. Furthermore, through rigorous correlation analysis, we find that LaaJ emerges as a near-perfect proxy for human preference, while continuous ranking metrics (AUROC and NDCG) serve as the most reliable reference-based alternatives Finally, the choice of the discrete selection strategy interacts non-trivially with the attribution method itself, suggesting that the two should be co-optimized rather than selected independently.

These findings provide practitioners with actionable guidance for deploying attribution in production RAG systems: White-box methods are the default recommendation across most deployment scenarios, and LaaJ offers a highly reliable, scalable stand-in for costly human evaluation. Future work should investigate the interaction between attribution quality and retrieval depth, the impact of domain-specific fine-tuning on attribution stability, and richer human evaluation paradigms beyond forced pairwise comparisons.

Limitations

Coverage of evaluation regimes Both EQA and ASQA do not provide span-level supervision, restricting our analysis on these datasets strictly to reference-free metrics. A unified, ranking-annotated long-context benchmark covering multiple domains remains an open need.

Scale of human evaluation Our 300-pair human evaluation is sufficient to separate top from bottom methods but not to resolve mid-tier ties with statistical confidence. Larger-scale annotation, ideally across diverse annotator pools, would tighten the conclusions about user-perceived verifiability.

Beyond English QA All four datasets are English. Whether the relative ordering of attribution methods generalizes to multilingual settings or to domain-specialized RAG – legal, biomedical, or scientific – is an open question we leave to future work.

Coverage of attribution paradigms Our benchmark fixes a representative set of six methods spanning white-box, perturbation-based, and surrogate paradigms. Training-based and reward-driven attribution approaches form an emerging fourth category, and evaluating them against this same multi-dimensional protocol – ranking, corroborative, and human-grounded – is a natural next step.

Hardware caveat Efficiency numbers in this section are not directly comparable across all datasets: VJ, MS, and ASQA were run on NVIDIA A100-80GB GPUs, while EQA was run exclusively on NVIDIA H200 GPUs due to its longer context windows. Qwen-32B and Phi-3-Medium were additionally evaluated on H200 hardware. Consequently, cross-dataset efficiency comparisons (including the aggregate scatter plots in Figure 6) should be interpreted with this heterogeneity in mind; within-dataset comparisons remain valid.

Ethics Statement

Human participants. Our pairwise evaluation study was conducted via Prolific, a crowdsourcing platform that provides informed consent and fair compensation for research participants. A total of 60 annotators took part in the study, completing 300 pairwise comparisons across 15 method pairs. Each annotator received £5.01 (approximately £10.02 per hour) for their time, consistent with Prolific's minimum wage guidelines and regional standards for the recruited countries. Participants were recruited based on the requirement that their primary language is English. Attention check questions were embedded throughout the survey to ensure response quality; responses from participants who failed attention checks were excluded from analysis.

Informed consent. All participants provided informed consent prior to beginning the study. The survey instructions clearly described the study's purpose, the nature of the tasks, the estimated completion time of 30 minutes, and the compensation amount. Participants could withdraw at any time without penalty, in accordance with Prolific's participant protections.

Data handling and privacy. Annotator responses were collected anonymously through Prolific's API and stored securely on institution servers. No personally identifiable information was recorded beyond the anonymized Prolific worker ID, which was used solely for compensation and quality filtering. Raw annotation data will be shared publicly as part of our open-source release; with all worker identifiers permanently removed to ensure anonymity. The data collection protocol was approved by the institutional ethics committee. To ensure participant privacy, the final version will be shared in a subsequent phase.

Content exposure. The evaluation stimuli were drawn from the VJ and MS MARCO datasets, which contain question-answer pairs spanning diverse web-sourced topics. While we do not expect the content to pose significant risk to annotators, some passages may contain sensitive or controversial material given the open-domain nature of the source documents. No content warnings were issued, as the tasks required evaluating attribution quality rather than engaging deeply with the content.

Potential harms of the research. Our benchmark evaluates the reliability of answer attribution methods in RAG systems. By identifying which methods produce faithful, human-aligned citations, this work supports the development of more trustworthy AI systems. Conversely, deploying attribution methods with known weaknesses (e.g., high OOM failure rates, poor human alignment) in high-stakes domains such as healthcare or law could mislead users into over-trusting incorrect or unsupported claims. We hope our findings mitigate this risk by providing practitioners with evidence-based deployment guidance.

Statistical significance and reliability metrics While our study establishes a clear macro-level alignment between human preferences and the LLM judge, our current results rely primarily on point estimates and rank correlations. Expanding this analysis to include formal micro-level inter-rater reliability metrics and strict confidence bounds for the performance rankings remains an important next step. Incorporating these rigorous statistical significance tests in future iterations will

help to more definitively separate closely performing mid-tier methods and further validate the robustness of the automated judge.

References

Marah Abdin, Sam Ade Jacobs, Ammar Ahmad Awan, Jyoti Aneja, Ahmed Awadallah, Hany Awadalla, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Harkirat Behl, and 1 others. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*.

Kendrick Boyd, Kevin H. Eng, and C. David Page. 2013. Area under the precision-recall curve: Point estimates and confidence intervals. In *Machine Learning and Knowledge Discovery in Databases*, pages 451–466, Berlin, Heidelberg. Springer Berlin Heidelberg.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, and 12 others. 2022. Scaling instruction-finetuned language models. *arXiv preprint*.

Benjamin Cohen-Wang, Harshay Shah, Kristian Georgiev, and Aleksander Madry. 2024. Contextcite: Attributing model generation to context. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Arun Das and Paul Rad. 2020. Opportunities and challenges in explainable artificial intelligence (xai): A survey. *arXiv preprint arXiv:2006.11371*.

Qiang Ding, Lvzhou Luo, Yixuan Cao, and Ping Luo. 2025. Attention with dependency parsing augmentation for fine-grained attribution. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 372–387, Vienna, Austria. Association for Computational Linguistics.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. 2023. Enabling large language models to generate text with citations. *Preprint*, arXiv:2305.14627.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Lovisa Hagström, Sara Vera Marjanovic, Haeun Yu, Arnav Arora, Christina Lioma, Maria Maistro, Pepa Atanasova, and Isabelle Augenstein. 2025. A reality check on context utilisation for retrieval-augmented generation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 19691–19730, Vienna, Austria. Association for Computational Linguistics.

J A Hanley and B J McNeil. 1982. The meaning and use of the area under a receiver operating characteristic (roc) curve. *Radiology*, 143(1):29–36. PMID: 7063747.

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, Seattle, United States. Association for Computational Linguistics.

Chengyu Huang, Zeqiu Wu, Yushi Hu, and Wenya Wang. 2024. Training language models to generate text with citations via fine-grained rewards. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2926–2949.

Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of ir techniques. *ACM Trans. Inf. Syst.*, 20(4):422–446.

Albert Qiaochu Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *ArXiv*, abs/2310.06825.

Wenqi Jiang, Suvinay Subramanian, Cat Graves, Gustavo Alonso, Amir Yazdanbakhsh, and Vidushi Dadu. 2025. Rago: Systematic performance optimization for retrieval-augmented generation serving. In *Proceedings of the 52nd Annual International Symposium on Computer Architecture*, ISCA '25, page 974–989, New York, NY, USA. Association for Computing Machinery.

Ruizhe Li, Chen Chen, Yuchen Hu, Yanjun Gao, Xi Wang, and Emine Yilmaz. 2025. Attributing response to context: A jensen–shannon divergence driven mechanistic study of context attribution in retrieval-augmented generation. In *Mechanistic Interpretability Workshop at NeurIPS 2025*.

Fengyuan Liu, Nikhil Kandpal, and Colin Raffel. 2025. Attribot: A bag of tricks for efficiently approximating leave-one-out context attribution. In *The Thirteenth International Conference on Learning Representations*.

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

Pedro Lopes, Eduardo Silva, Cristiana Braga, Tiago Oliveira, and Luís Rosado. 2022. Xai systems evaluation: A review of human and computer-centred methods. *Applied Sciences*, 12(19):9423.

Varun Magesh, Faiz Surani, Matthew Dahl, Mirac Suzgun, Christopher D. Manning, and Daniel E. Ho. 2025. Hallucination-free? assessing the reliability of leading ai legal research tools. *Journal of Empirical Legal Studies*, 22(2):216–242.

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

Majid Mohammadi, Krikamol Muandet, Ilaria Tiddi, Annette Ten Teije, and Siu Lun Chau. 2026. Exact shapley attributions in quadratic-time for fanova gaussian processes. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(29):24414–24421.

Pranav Narayanan Venkit, Philippe Laban, Yilun Zhou, Yixin Mao, and Chien-Sheng Wu. 2025. Search engines in the ai era: A qualitative understanding to the false promise of factual and verifiable source-cited responses in llm-based search. In *Proceedings of the 2025 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '25, page 1325–1340, New York, NY, USA. Association for Computing Machinery.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. Ms marco: A human generated machine reading comprehension dataset. In *CoCo@NIPS*, volume 1773 of *CEUR Workshop Proceedings*. CEUR-WS.org.

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

Anirudh Phukan, Shwetha Somasundaram, Apoorv Saxena, Koustava Goswami, and Balaji Vasan Srinivasan. 2024. Peering into the mind of language models: An approach for attribution in contextual question answering. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 11481–11495, Bangkok, Thailand. Association for Computational Linguistics.

Jirui Qi, Gabriele Sarti, Raquel Fernández, and Arianna Bisazza. 2024. Model internals-based answer attribution for trustworthy retrieval-augmented generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 6037–6053, Miami, Florida, USA. Association for Computational Linguistics.

Gorjan Radevski, Kiril Gashteovski, Shahbaz Syed, Christopher Malon, Sebastien Nicolas, Chia-Chien Hung, Timo Sztyler, Verena Heußer, Wiem Ben Rim, Masafumi Enomoto, Kunihiro Takeoka, Masafumi Oyamada, Goran Glavaš, and Carolin Lawrence. 2025. On synthesizing data for context attribution in question answering. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16929–16950, Vienna, Austria. Association for Computational Linguistics.

Vipula Rawte, Ryan A. Rossi, Franck Dernoncourt, and Nedim Lipka. 2025. Document attribution: Examining citation relationships using large language models. In *Proceedings of the Fifth Workshop on Scholarly Document Processing (SDP 2025)*, pages 132–136, Vienna, Austria. Association for Computational Linguistics.

Yash Saxena, Raviteja Bommireddy, Ankur Padia, and Manas Gaur. 2025. Generation-time vs. post-hoc citation: A holistic evaluation of LLM attribution. In *NeurIPS 2025 Workshop on Evaluating the Evolving LLM Lifecycle: Benchmarks, Emergent Abilities, and Scaling*.

Ivan Stelmakh, Yi Luan, Bhuwan Dhingra, and Ming-Wei Chang. 2022. ASQA: Factoid questions meet long-form answers. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8273–8288, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jingyi Sun, Pepa Atanasova, Sagnik Ray Choudhury, Sekh Mainul Islam, and Isabelle Augenstein. 2026. Evaluation framework for highlight explanations of context utilisation in language models. *Computational Linguistics*, pages 1–33.

Qwen Team. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

Alexander Wan, Eric Wallace, and Dan Klein. 2024. What evidence do language models find convincing? In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7468–7484, Bangkok, Thailand. Association for Computational Linguistics.

Orion Weller, Marc Marone, Nathaniel Weir, Dawn Lawrie, Daniel Khashabi, and Benjamin Van Durme. 2024. "according to...": Prompting language models improves quoting from pre-training data. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2288–2301.

Theodora Worledge, Judy Hanwen Shen, Nicole Meister, Caleb Winston, and Carlos Guestrin. 2024. Sok:

Unifying corroborative and contributive attributions in large language models. In *2nd IEEE Conference on Secure and Trustworthy Machine Learning*.

Yingtai Xiao, Yuqing Zhu, Sirat Samyoun, Wanrong Zhang, Jiachen T. Wang, and Jian Du. 2025. Token-Shapley: Token level context attribution with shapley value. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 3882–3894, Vienna, Austria. Association for Computational Linguistics.

Rui Xing, Timothy Baldwin, and Jey Han Lau. 2025. Evaluating evidence attribution in generated fact checking explanations. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5475–5496, Albuquerque, New Mexico. Association for Computational Linguistics.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. Bartscore: Evaluating generated text as text generation. *Preprint*, arXiv:2106.11520.

A Methods

MIRAGE (2024). MIRAGE (*Model Internals-based RAG Explanations*) is a white-box, token-level attribution method that requires access to model internal activations and gradients to quantify how retrieved context affects answer generation. It proceeds in two stages: (i) it identifies context-sensitive generated tokens by measuring changes in token prediction probabilities when the retrieval context is removed, and (ii) it attributes these tokens to influential context tokens via gradient-based saliency. Although the underlying attribution is computed at token resolution, MIRAGE aggregates token-level signals into sentence-level document citations to improve interpretability in retrieval-augmented generation (RAG) systems. The resulting explanation is typically presented as grounded answer highlights with inline citations linking generated spans to supporting retrieved documents.

ContextCite (2024). ContextCite is a black-box (probability-access) context-attribution method that primarily targets sentence-level attributions, while permitting alternative source granularities (e.g., documents, paragraphs, or words). The method repeatedly ablates candidate context segments and measures the induced change in the probability of the generated statement. It then fits a sparse linear surrogate model to approximate the marginal effect of each segment on the model output, and interprets the surrogate weights as attribution scores. The output is a ranked list of context sources with real-valued importance scores, indicating which context segments causally contribute to a response.

AttriBoT (2025). AttriBoT is a black-box/gray-box context-attribution framework that approximates leave-one-out influence estimates over context spans, typically at the sentence or paragraph granularity. The method is perturbation-based: each candidate span is ablated from the context and its contribution is quantified by the induced change in the model likelihood of the original response. To reduce the computational cost of exact leave-one-out evaluation, AttriBoT incorporates several efficiency mechanisms, including key–value caching, hierarchical attribution, proxy modeling with smaller models, and proxy-based pruning. The output is a set of real-valued attribution scores over spans, commonly reported as a ranked list of sentences, paragraphs, or documents that most strongly affect the target model's generation.

HSSAVG (2024). HSSAVG is a white-box, model-internals-based method for token- and span-level attribution in contextual question answering. It leverages hidden-state similarity between generated answer tokens and document tokens, under the assumption that copied or context-dependent answer spans preserve representational proximity to their source text. The method first detects extractive answer tokens using cosine similarity between answer-token and document-token hidden states, and then attributes an answer span by averaging its hidden representations and matching them against sliding windows in the source document. The output is an evidence window (source passage) per attributed answer span, often visualized as highlighted answer tokens aligned to the corresponding supporting region in the document.

AttnUnionDep (2025). AttnUnionDep is a white-box attribution method that targets fine-grained token- and answer-span evidence, and can be aggregated to sentence-level attribution. It uses attention weights as a model-internal association signal between response tokens and prompt/document tokens. Its core component (AttnUnion) attributes each target token to high-attention document tokens and aggregates evidence via set-union operations, avoiding the coarse representations induced by hidden-state averaging. The

Dataset	Domain / Topic	Samples	Avg Doc Length (Words)	Avg Docs / Sample
Verifiability Judgments (Liu et al., 2023)	Q&A	624	1848.4	1.6
MS MARCO (Nguyen et al., 2016)	General	5,412	817.7	1.0
ExpertQA (Malaviya et al., 2024)	Specialized Domains	5,813	3410.2	1.1
ASQA (Stelmakh et al., 2022)	Wikipedia / Ambiguous Factoid	3,820	5461.6	1.5

Table 5: Statistics of our four question answering corpora used in the evaluation.

DEP extension incorporates dependency parsing to expand a target span with semantically related tokens from the same atomic fact (e.g., predicates, subjects, and objects). The output consists of evidence tokens, spans, or passages with attribution scores, typically visualized as fine-grained highlights aligned to specific answer spans.

TokenShapley (2025). TokenShapley is a white-box, token-level context-attribution method that assigns Shapley-value-based contributions from context tokens to generated response tokens. The approach builds on KNN-augmented language modeling by constructing a datastore of contextual token representations, where each entry maps a prefix representation to the subsequent token. For a given response token, TokenShapley retrieves nearest neighbors in representation space and computes exact weighted KNN-Shapley values to estimate each context token's marginal contribution. Relative to sentence-level attribution, the method is intended to provide fine-grained evidence for specific response elements (e.g., entities, dates, and numeric claims). The output is a token-to-token attribution matrix or a ranked list of influential context tokens, which can be aggregated into higher-level evidence spans when required.

B Datasets

Data Curation Pipeline Across all datasets, we follow a common procedure: (1) extract URLs from dataset annotations (e.g., passages, citations, or evidence fields), (2) deduplicate URLs at the dataset level using normalized URLs, (3) validate link accessibility (e.g., successful HTTP response without network or SSL failures), (4) validate content quality by ensuring the page contains readable text, is not blocked (e.g., CAPTCHA or interstitial pages), and meets minimum length thresholds, (5) extract the main textual content of the page (e.g., article body), and (6) map the processed documents back to their corresponding samples.

We focus on extracting the main textual content using tools such as `trafilatura` with heuristic fallbacks (e.g., `BeautifulSoup`), and discard pages that do not yield sufficient readable content.

In addition to this shared backbone, dataset-specific steps are applied. For *ASQA*, we perform relevance filtering by scoring the similarity between the annotated long answer and the extracted page text using TF-IDF cosine similarity, with lexical overlap as a fallback. For *MS MARCO*, we verify that the dataset-provided passage text is still present in the live webpage using normalized substring matching. For *Verifiability Judgments*, we additionally verify that annotated evidence spans are present in the source documents using normalized and token-overlap matching.

Verifiability Judgments The Verifiability Judgments dataset contains human-annotated responses with inline citations and associated evidence spans. After filtering invalid entries, we process 5,527 samples. We retain 624 samples where at least one cited source is reachable and contains the annotated evidence.

From 8,599 unique URLs, 6,942 are successfully retrieved and used in our experiments, while the remaining URLs are discarded due to accessibility failures or missing evidence in the retrieved content.

ExpertQA ExpertQA consists of expert-validated question-answer pairs with claim-level annotations, including attribution links generated by systems such as GPT-4 and Bing Chat. We process all samples without removing entire instances, but filter at the claim and document level.

Out of 21,395 URL references (8,771 unique URLs), 4,934 are successfully retrieved and pass content quality checks. The remaining URLs are discarded due to retrieval failures or insufficient textual content. In total, 10,745 claims are retained after filtering from 12,598 annotated claims, forming our final evaluation set of 5,813 samples.

MS MARCO We use a subset of 100,000 samples from the MS MARCO question answering datasets (v1.1 and v2.1 variants). Each sample contains a query, an answer, and associated passages

with source URLs.

From 89,604 unique URLs, 47,155 are reachable, and 42,093 pass content quality checks. We further ensure that the original passage text provided in the dataset is still present in the retrieved webpage using normalized substring matching. URLs failing this consistency check or content validation are discarded, resulting in a final evaluation set of 5,412 samples.

ASQA ASQA is a long-form question answering dataset designed for ambiguous questions, containing 4,353 training samples with annotated long answers and associated Wikipedia pages.

We retain 3,820 samples that contain at least one relevant source after filtering. From 7,275 referenced URLs, 4,783 are successfully retrieved and pass content validation. Relevance is determined by scoring the similarity between the annotated long answer and the extracted page text using TF-IDF cosine similarity with lexical overlap fallback.

URLs are discarded either due to low relevance or failure to meet content quality criteria.

C Hardware Infrastructure

To assess computational overhead and practical applicability, we analyze execution runtime, peak VRAM allocation, and operational reliability. To accommodate varying architectural demands, experiments were distributed across different hardware setups. Baseline models (LLAMA, MISTRAL, PHI, QWEN) utilized NVIDIA A100-80GB GPUs, while the 32-billion parameter QWEN32 utilized NVIDIA H200 GPUs. Notably, PHI and PHIH200 refer to the identical base model, differentiated solely to indicate runs executed on the A100 versus the H200. Furthermore, all model evaluations on the ExpertQA (EQA) dataset were strictly conducted on H200 GPUs to manage its extended context window demands.

Hardware Profiling and Execution Strategy A core objective of our benchmark is not only to measure ranking accuracy but also to comprehensively profile the operational efficiency of these attribution algorithms. Extracting hidden states and tracking gradients for long-context attribution is notoriously memory-intensive. Therefore, our hardware execution strategy was deliberately designed to analyze how different VRAM capacities influence the memory management, bottleneck behaviors, and execution latency of the methods under test.

- **Baseline A100 (80GB) Evaluations:** The evaluations for the *Verifiability Judgments*, *MS-MARCO*, and *ASQA* datasets were systematically executed using **NVIDIA A100 (80GB) GPUs**, establishing our baseline for high-performance hardware constraints.
- **H200 High-Capacity Profiling:** The entire *ExpertQA* dataset was exclusively processed using **NVIDIA H200 GPUs**. This strategic decision was made not merely due to the extreme length of the expert-level documents, but crucially to collect comparative hardware metrics. By deploying the algorithms in a vastly higher VRAM environment, we could rigorously profile how excess memory capacity alleviates internal fragmentation and impacts the scalability of each attribution method.
- **Hardware Exceptions and Consistency Validation:**
  - *Qwen-32B VRAM Threshold:* Due to the sheer parameter scale of the 32B model, computing full attribution passes exceeded the absolute memory limits of the A100s. Consequently, the evaluation of **Qwen-32B** on the *Verifiability Judgments* dataset was strategically routed to the **NVIDIA H200** cluster.
  - *Mid-Scale VRAM Sensitivity (Phi-3-Medium, 14B):* While our baseline 7B models (e.g., Mistral-7B, Qwen-7B) comfortably execute within the 80GB VRAM limit without encountering memory bottlenecks, the **Phi-3-Medium** model, at approximately 14B parameters, represents a critical intermediate memory threshold. At this scale, the massive memory overhead required for extracting hidden states and tracking gradients approaches the physical limits of an A100 GPU. To rigorously investigate whether an abundance of VRAM fundamentally alters memory management behaviors, execution efficiency, or the final attribution scores for mid-sized models, Phi-3-Medium was intentionally evaluated twice across the entire benchmark: once on the A100 (80GB) and once on the high-capacity H200.

Execution Optimizations and Preprocessing Heuristics To maximize throughput and prevent memory fragmentation on long-context documents, all base models were loaded in `bfloat16` precision

utilizing Flash Attention 2 alongside PyTorch's expandable memory segments. However, because optimized Flash Attention kernels do not materialize attention probability matrices in memory, our pipeline features a dynamic module swapper. For attribution methods requiring explicit attention weights (e.g., AttnUnionDep), the framework temporarily hot-swaps the execution module to PyTorch's native Scaled Dot-Product Attention (SDPA) during runtime. During the input preprocessing phase, retrieved documents were segmented into sentences using NLTK and subjected to a strict heuristic filter: any parsed segment containing fewer than 3 words or 15 characters was discarded to consistently eliminate web-scraping artifacts and navigational noise. Finally, to ensure strictly isolated hardware profiling, CUDA peak memory statistics were forcefully reset prior to each method's execution, and any OOM anomalies were gracefully caught, logged, and isolated without disrupting the comparative batch.

1. **Top-1:** Selecting only the highest-scoring sentence.
2. **Top-3:** Selecting the top three highest-scoring sentences.
3. **K-Means:** Applying 1D K-Means clustering (k=2) to separate high-confidence sentences from the background noise.

D Annotation Interface

This appendix provides detailed views of the human evaluation interface. Figure 2 displays the A/B comparison task, while Figure 9 illustrates the subsequent 7-point Likert-scale questionnaire.

E Data Scope and Aggregation Methodology

To ensure full transparency regarding the evaluation scope of each metric, Table 9 details the exact datasets and models used to generate every figure and table in this study. Because different metrics inherently operate on different subsets of data (e.g., human evaluations strictly use LLAMA on MS and VJ, whereas computational efficiency spans all models and datasets), the aggregation methodology varies per visualization to maintain strict "apples-to-apples" comparisons without introducing empty data biases.

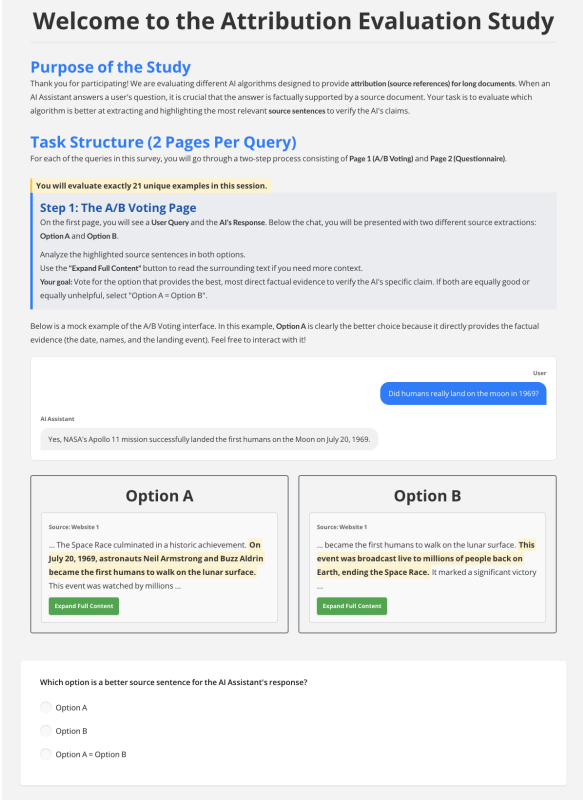

Figure 7: A/B Test Introduction interface showing step 1.

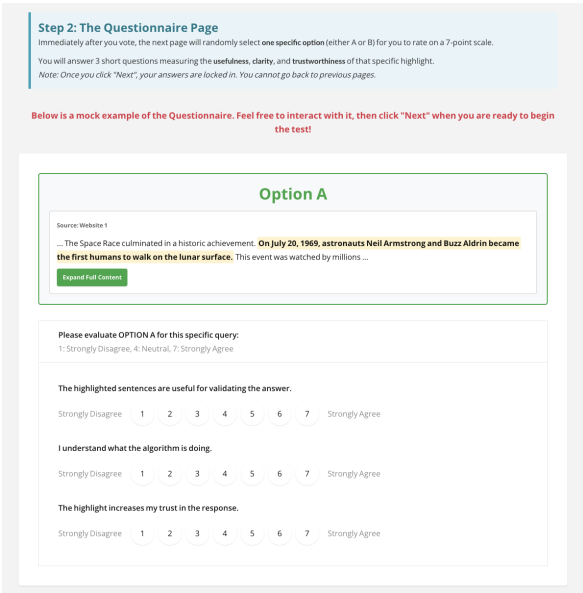

Figure 8: A/B Test Introduction interface showing step 2.

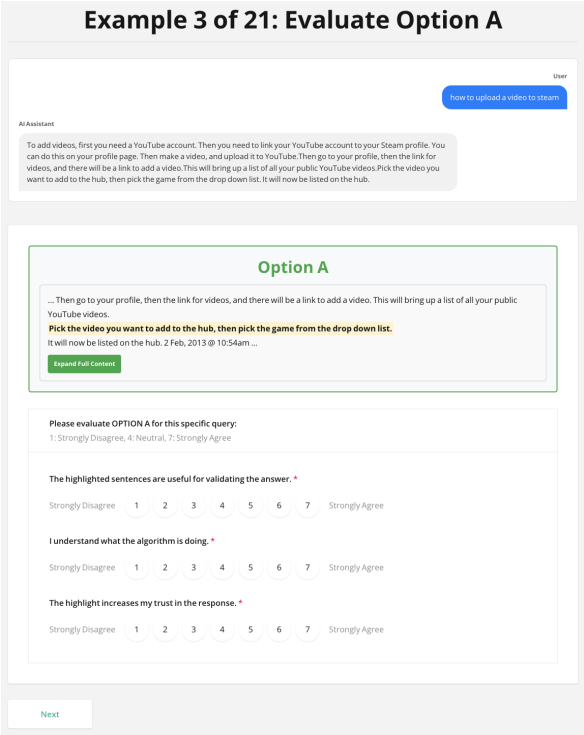

Figure 9: Questionnaire interface for rating a single attribution option using 7-point Likert scales.

Metric	Datasets	Models
Strict Scope (Human Evaluation)		
Performance	MS, VJ	LLAMA
Understandability	MS, VJ	LLAMA
Usefulness	MS, VJ	LLAMA
Trust	MS, VJ	LLAMA
Intermediate Scope (Probabilistic & LLM-as-a-Judge)		
AUPRC	MS, VJ	All Models
AUROC	MS, VJ	All Models
NDCG	MS, VJ	All Models
F1-Score	MS, VJ	All Models
LAAJ Score	MS, VJ	All Models
Universal Scope (Hardware & Text Generation)		
Runtime (s)	All Datasets	All Models
Peak Memory (GB)	All Datasets	All Models
NLI-Score	All Datasets	All Models
BARTScore	All Datasets	All Models
LogDrop	All Datasets	All Models
OOM Rate (%)	All Datasets	All Models

Table 6: Evaluation metrics and their corresponding data availability scopes (Datasets and Models). "All Datasets" refers to MS, VJ, EQA, and ASQA. "All Models" refers to LLAMA, QWEN, QWEN32, MISTRAL, PHI, and PHIH200.

F Correlation Methodology and Cell-by-Cell Scope Alignment

To prevent dataset bias and ensure pure "apples-to-apples" comparisons, we employ a *Cell-by-Cell Scope Alignment* strategy for the Pearson correlation matrix. Averaging metrics indiscriminately across all data would introduce severe noise, as our metrics naturally span different evaluation subsets.

We categorize all metrics into three scopes:

1. **Strict (Human Evaluation):** Evaluated only on `MS` and `VJ` datasets using the `LLAMA` model.
2. **Intermediate (Probabilistic & LAAJ):** Evaluated on `MS` and `VJ` datasets across **all models**.
3. **Universal (Hardware & Text Gen):** Evaluated globally across **all datasets** and **all models**.

For any given matrix cell correlating metric X and metric Y, the algorithm dynamically filters the dataset to their exact intersection (Lowest Common Denominator) before aggregating method-level means:

$$S_{X,Y} = S_X \cap S_Y$$

For example, when correlating *Performance* (Strict) with *Runtime* (Universal), the *Runtime* is strictly recalculated using only `LLAMA` data on `MS/VJ`. This dynamic masking guarantees that every correlation coefficient is derived from identically aligned sample spaces, providing a robust and unbiased reflection of trade-offs.

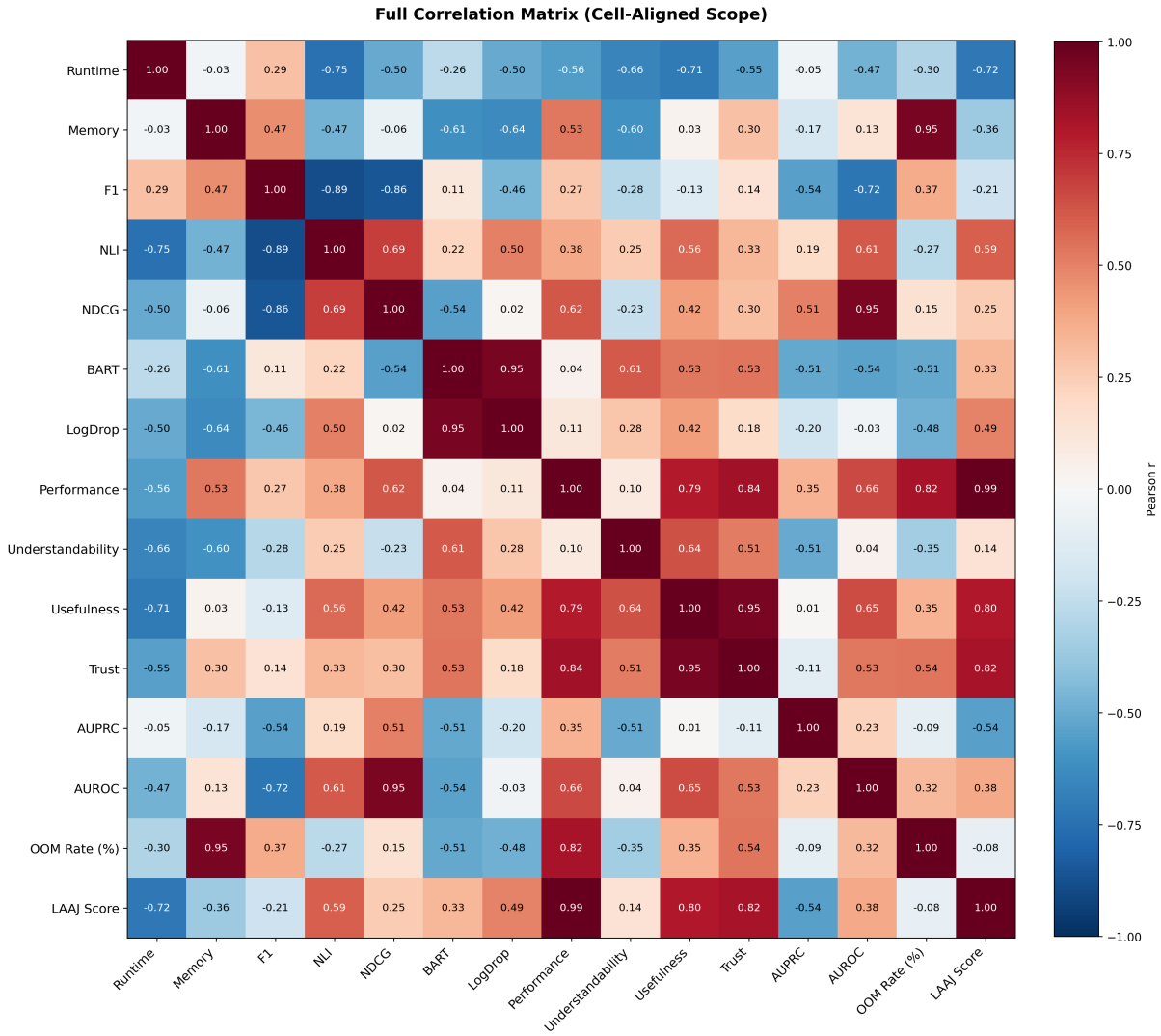

Figure 10: Comprehensive square Pearson correlation matrix across all automated and human evaluation metrics, calculated using a Cell-by-Cell Scope Alignment approach to eliminate dataset bias and sample mismatch. For a detailed explanation of the dynamic intersection methodology, see Appendix F.

G LLM-as-a-Judge Prompt Template

To ensure the reproducibility of our LLM-as-a-Judge evaluation, we provide the exact prompt template used for the pairwise comparison of attribution methods. The prompt was executed using the Llama-3.3-70B-Instruct model. (see Figure 11)

H Extended Efficiency and Reliability Analysis

This severe memory footprint directly impacts real-world reliability. Figure 12 reports the failure (unscored) rates across all 516,929 evaluations. It is imperative to note that these failed instances, which are predominantly caused by OOM errors, were completely excluded from the downstream performance metrics discussed in 4. Due to its high memory demands, *MIRAGE* exhibits the highest OOM failure rates across most models. A glaring anomaly is the 45.92% failure rate for the *MIRAGE-MISTRAL* pairing; this specific spike is not solely due to VRAM limitations, but rather a structural model-method incompatibility. *AttnUnionDep* also shows moderate instability, dropping 10-15% of examples.

I Programming Setup

All experiments and codebase developments were conducted using Python 3.10 within the Visual Studio Code. To ensure reproducibility and prevent dependency conflicts across different stages of our evaluation pipeline, we maintained strictly isolated virtual environments for distinct tasks. The full dependency trees are provided in our public repository, with the core frameworks highlighted below:

- **Method Execution Environment:** The base models and attribution methods were executed using `transformers` (v4.44.2), `accelerate` (v0.32.1), and `captum` (v0.8.0) to handle internal state access, gradients, and perturbations. Additional data handling was managed via `datasets` (v4.7.0) and `sentence-transformers` (v5.2.3).
- **Automatic Evaluation Environment:** To maintain compatibility with legacy evaluation scripts for corroborative metrics (e.g NLI-Score, BARTScore, F_1), these evaluations were isolated in an environment running `transformers` (v4.40.2), `accelerate` (v0.30.1), and `numpy` (v1.26.4).
- **LLM-as-a-Judge Environment:** For high-throughput automated judgments, we utilized `vllm` (v0.14.0) as the primary inference engine, interfaced via the `openai` (v2.37.0) client and running alongside `transformers` (v4.57.6) and `ray` (v2.55.1).

Finally, auxiliary data processing, script optimization, and formatting tasks during the development phase were assisted by Gemini Pro 3.1.

J Implementation Details and Hyperparameters

To ensure the reproducibility of our experiments, we provide the complete set of hyperparameters, hardware constraints, and pipeline configurations used for all evaluated context attribution methods.

All language models were executed in `torch.bfloat16` precision using Flash Attention 2 to optimize memory and throughput. Method-specific thresholds—such as the hierarchical chunk sizes for AttriBoT, the layer indices for HSSAVG and TokenShapley, and the gradient saliency parameters for MIRAGE—were strictly aligned with the optimal settings recommended in their respective original papers.

A comprehensive and granular breakdown of all global pipeline arguments and method-specific parameters is detailed in Tables 25 to 31.

```
You are an impartial and expert judge. Your task is to evaluate which
candidate 'Evidence' is more helpful and sufficient to verify the
provided 'Target Answer' based on the original 'Query'.

QUERY: {query}
TARGET ANSWER: {target_answer}

CANDIDATE A:
[Extracted Sentence]: {candidate_a}
[Surrounding Document Context]: {context_a}

CANDIDATE B:
[Extracted Sentence]: {candidate_b}
[Surrounding Document Context]: {context_b}

Evaluation Criteria:
- Which candidate's extracted sentence more directly and completely
  supports the claims made in the Target Answer in the context of the Query?
- Use the surrounding document context ONLY to understand the full meaning
  of the extracted sentence. The quality of the extraction itself is what matters.
- Penalize candidates that contain hallucinated or irrelevant information.

Output your evaluation STRICTLY in the following JSON format.
Do not provide any explanations or reasoning, just declare the winner.
{
  "winner": "A"
}
Note: For 'winner', strictly use "A", "B", or "Tie".
```

Figure 11: Prompt template used for the LLM-as-a-Judge pairwise evaluation.

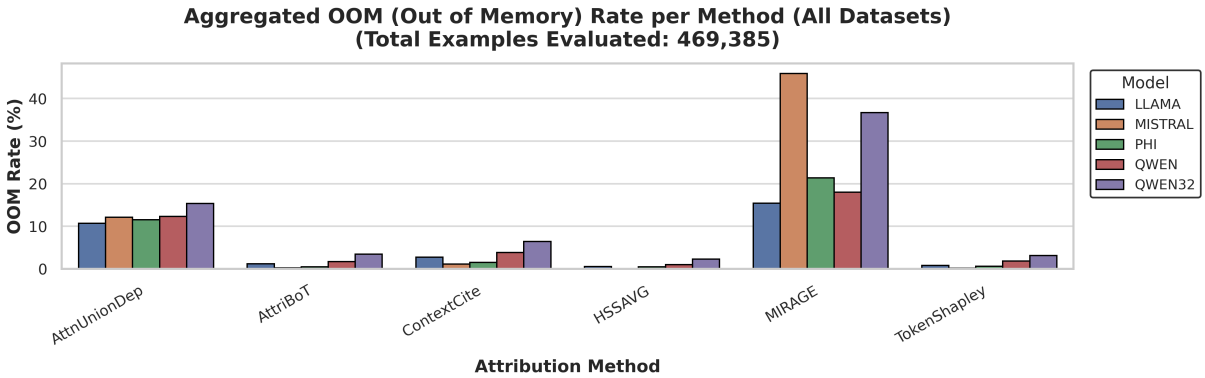

Figure 12: Aggregated percentage of unscored (failed) examples per attribution method across all four datasets (516,929 total evaluations). Failures are predominantly driven by OOM errors.

Model	#	Method	Wins	Ties	Losses	Total	Score
LLAMA	1	AttnUnionDep	281	28	147	456	0.2939
	2	AttriBoT	246	13	197	456	0.1075
	3	ContextCite	242	12	202	456	0.0877
	4	HSSAVG	232	20	204	456	0.0614
	5	TokenShapley	230	16	210	456	0.0439
	6	MIRAGE	119	6	331	456	-0.4649
MISTRAL	1	ContextCite	268	8	154	430	0.2651
	2	TokenShapley	254	10	166	430	0.2047
	3	AttriBoT	250	7	173	430	0.1791
	4	HSSAVG	203	13	214	430	-0.0256
	5	AttnUnionDep	197	10	223	430	-0.0605
	6	MIRAGE	92	4	334	430	-0.5628
PHI	1	AttnUnionDep	167	12	106	285	0.2140
	2	TokenShapley	159	15	111	285	0.1684
	3	ContextCite	155	13	117	285	0.1333
	4	AttriBoT	145	13	127	285	0.0632
	5	HSSAVG	120	10	155	285	-0.1228
	6	MIRAGE	75	5	205	285	-0.4561
QWEN	1	AttnUnionDep	264	21	155	440	0.2477
	2	ContextCite	252	14	174	440	0.1773
	3	TokenShapley	234	16	190	440	0.1000
	4	AttriBoT	227	11	202	440	0.0568
	5	HSSAVG	215	19	206	440	0.0205
	6	MIRAGE	85	5	350	440	-0.6023
QWEN32	1	TokenShapley	328	11	151	490	0.3612
	2	HSSAVG	283	14	193	490	0.1837
	3	AttriBoT	284	10	196	490	0.1796
	4	ContextCite	270	20	200	490	0.1429
	5	AttnUnionDep	181	7	302	490	-0.2469
	6	MIRAGE	93	0	397	490	-0.6204

Table 7: LLM-as-a-Judge detailed performance for the VJ dataset.

Model	#	Method	Wins	Ties	Losses	Total	Score
LLAMA	1	HSSAVG	4800	290	2440	7530	0.3134
	2	AttnUnionDep	4380	277	2873	7530	0.2001
	3	TokenShapley	3961	281	3288	7530	0.0894
	4	ContextCite	3756	316	3458	7530	0.0396
	5	AttriBoT	3598	313	3619	7530	-0.0028
	6	MIRAGE	2016	224	5290	7530	-0.4348
MISTRAL	1	TokenShapley	4594	192	2524	7310	0.2832
	2	HSSAVG	4295	210	2805	7310	0.2038
	3	ContextCite	3728	253	3329	7310	0.0546
	4	AttriBoT	3620	220	3470	7310	0.0205
	5	MIRAGE	2551	164	4595	7310	-0.2796
	6	AttnUnionDep	2542	161	4607	7310	-0.2825
PHI	1	AttnUnionDep	4244	222	2104	6570	0.3257
	2	HSSAVG	3480	223	2867	6570	0.0933
	3	TokenShapley	3415	225	2930	6570	0.0738
	4	ContextCite	2875	255	3440	6570	-0.0860
	5	AttriBoT	2646	238	3686	6570	-0.1583
	6	MIRAGE	2372	193	4005	6570	-0.2486
QWEN	1	HSSAVG	4840	256	2394	7490	0.3266
	2	AttnUnionDep	4463	259	2768	7490	0.2263
	3	TokenShapley	4091	246	3153	7490	0.1252
	4	ContextCite	3294	268	3928	7490	-0.0846
	5	AttriBoT	3284	258	3948	7490	-0.0887
	6	MIRAGE	1732	245	5513	7490	-0.5048
QWEN32	1	TokenShapley	4864	232	2399	7495	0.3289
	2	HSSAVG	4822	234	2439	7495	0.3179
	3	ContextCite	3759	283	3453	7495	0.0408
	4	AttriBoT	3734	266	3495	7495	0.0319
	5	MIRAGE	2325	212	4958	7495	-0.3513
	6	AttnUnionDep	2279	177	5039	7495	-0.3682

Table 8: LLM-as-a-Judge detailed performance for the MS dataset.

Element	Description	Dataset Scope	Model Scope	Aggregation Notes
Table 2	Automatic Evaluation Metrics	All	All	Macro-averaged across all available models and datasets.
Figure 3	Holistic Spider Chart	Metric-Specific	Metric-Specific	Averaged using the maximum available data per metric independently (e.g., Performance uses only LLAMA; NLI uses all models). *MIRAGE* is excluded as an extreme resource outlier.
Table 3	LLM-as-a-Judge Rankings	MS, VJ	All	Aggregated win-rates across all evaluated models for these two core datasets.
Figure 4	Hardware Efficiency	All	All	Averaged runtime and peak memory consumption across all evaluated scenarios.
Figure 5	Focus Correlation Matrix	Cell-Aligned	Cell-Aligned	Calculated via Cell-by-Cell Scope Alignment. *MIRAGE* is excluded as an extreme resource outlier.
Figure 6	Pareto Frontiers	Dynamic	Dynamic	Scope is dynamically restricted to the narrowest intersection of the two plotted metrics (e.g., NDCG vs. Performance strictly uses MS/VJ and LLAMA). *MIRAGE* is excluded as an extreme resource outlier.
Table 4	Human Evaluation Rankings	MS, VJ	LLAMA	Derived exclusively from the 300 pairwise human annotations.

Table 9: Summary of data scopes and aggregation strategies for all figures and tables presented in the study. "All" refers to the respective maximum combinations evaluated in this benchmark.

Dataset	Method	Shared Metrics			Top-1				Top-3				KMeans			
		AUPRC	AUROC	NDCG	F1	BART	NLI	LogDrop	F1	BART	NLI	LogDrop	F1	BART	NLI	LogDrop
ASQA	AttnUnionDep	—	—	—	—	-2.607	0.106	**1.007**	—	-2.088	**0.356**	**1.396**	—	-2.115	0.325	1.383
	AttriBoT	—	—	—	—	-2.570	0.100	0.986	—	-2.102	0.323	1.335	—	-2.188	0.248	1.282
	ContextCite	—	—	—	—	-2.560	0.105	1.006	—	-2.100	0.319	1.345	—	-2.245	0.233	1.248
	HSSAVG	—	—	—	—	-2.695	0.085	0.869	—	-2.324	0.256	1.143	—	-2.079	**0.520**	1.321
	MIRAGE	—	—	—	—	-2.782	0.084	0.889	—	-2.159	0.304	1.362	—	-2.561	0.148	1.055
	TokenShapley	—	—	—	—	**-2.533**	**0.109**	1.006	—	**-2.044**	0.343	1.368	—	**-1.985**	0.386	**1.421**
EQA	AttnUnionDep	—	—	—	—	-3.492	0.169	0.932	—	-3.291	**0.310**	1.077	—	-3.339	0.279	1.035
	AttriBoT	—	—	—	—	-3.542	0.150	0.890	—	-3.363	0.253	1.018	—	-3.378	0.227	0.998
	ContextCite	—	—	—	—	-3.541	0.155	0.889	—	-3.353	0.254	1.025	—	-3.390	0.225	0.994
	HSSAVG	—	—	—	—	**-3.418**	**0.178**	**0.985**	—	-3.275	0.310	1.090	—	-3.313	**0.376**	1.021
	MIRAGE	—	—	—	—	-3.750	0.102	0.742	—	-3.400	0.224	1.007	—	-3.656	0.133	0.813
	TokenShapley	—	—	—	—	-3.423	0.163	0.975	—	**-3.247**	0.291	**1.105**	—	**-3.247**	0.305	**1.089**
MS	AttnUnionDep	0.365	0.559	0.527	0.465	-3.052	0.444	1.653	0.416	-2.790	0.607	1.923	0.470	-2.785	0.588	1.868
	AttriBoT	0.382	0.492	0.518	0.494	-2.896	0.490	1.789	0.371	-2.720	0.625	1.970	0.496	-2.697	0.594	1.957
	ContextCite	0.370	0.531	0.528	0.499	-2.849	0.504	1.826	0.373	-2.690	0.632	1.989	0.498	-2.679	0.596	1.973
	HSSAVG	**0.434**	**0.613**	**0.578**	**0.520**	**-2.761**	**0.540**	**1.930**	**0.437**	-2.643	**0.676**	**2.108**	0.352	-2.690	**0.702**	**2.048**
	MIRAGE	0.299	0.512	0.488	0.394	-3.223	0.364	1.440	0.399	-2.712	0.565	1.918	0.401	-3.110	0.403	1.538
	TokenShapley	0.363	0.516	0.531	0.515	-2.777	0.521	1.887	0.410	**-2.640**	0.657	2.061	**0.511**	**-2.617**	0.637	2.020
VJ	AttnUnionDep	0.673	**0.887**	**0.789**	0.562	-2.206	0.328	1.640	0.513	-1.729	0.659	2.045	0.633	-1.779	0.573	1.990
	AttriBoT	**0.711**	0.803	0.788	0.580	-2.128	0.325	1.692	0.514	-1.661	0.669	2.066	0.661	-1.736	0.549	2.005
	ContextCite	0.663	0.830	0.780	**0.591**	**-2.098**	0.339	**1.717**	**0.522**	**-1.638**	0.672	2.086	**0.662**	-1.740	0.537	2.003
	HSSAVG	0.634	0.874	0.760	0.533	-2.227	0.326	1.604	0.482	-1.805	0.620	1.973	0.218	-1.784	**0.750**	2.046
	MIRAGE	0.464	0.773	0.630	0.348	-2.676	0.209	1.334	0.410	-1.915	0.534	1.945	0.384	-2.461	0.294	1.504
	TokenShapley	0.669	0.832	0.785	0.576	-2.102	**0.340**	1.705	0.521	-1.641	**0.679**	**2.094**	0.645	**-1.648**	0.637	**2.077**

Table 10: Performance metrics averaged across models per dataset.

Dataset	Method	Shared Metrics			Top-1				Top-3				KMeans			
		AUPRC	AUROC	NDCG	F1	BART	NLI	LogDrop	F1	BART	NLI	LogDrop	F1	BART	NLI	LogDrop
ASQA	AttnUnionDep	—	—	—	—	**0.0178**	**0.0007**	**0.0098**	—	**0.0287**	**0.0077**	**0.0172**	—	0.0199	0.0033	0.0133
	AttriBoT	—	—	—	—	0.0005	0.0000	0.0002	—	0.0011	0.0002	0.0005	—	0.0005	0.0001	0.0003
	ContextCite	—	—	—	—	0.0007	0.0000	0.0003	—	0.0014	0.0003	0.0007	—	0.0010	0.0002	0.0009
	HSSAVG	—	—	—	—	0.0029	0.0002	0.0015	—	0.0216	0.0037	0.0122	—	**0.0418**	**0.0260**	**0.0230**
	MIRAGE	—	—	—	—	0.0052	0.0000	0.0027	—	0.0042	0.0007	0.0025	—	0.0123	0.0006	0.0070
	TokenShapley	—	—	—	—	0.0008	0.0000	0.0003	—	0.0022	0.0003	0.0010	—	0.0023	0.0004	0.0011
EQA	AttnUnionDep	—	—	—	—	**0.0178**	**0.0018**	**0.0099**	—	**0.0094**	**0.0040**	**0.0051**	—	**0.0109**	**0.0021**	0.0064
	AttriBoT	—	—	—	—	0.0002	0.0000	0.0001	—	0.0002	0.0001	0.0001	—	0.0001	0.0000	0.0001
	ContextCite	—	—	—	—	0.0001	0.0000	0.0001	—	0.0001	0.0000	0.0000	—	0.0003	0.0001	0.0002
	HSSAVG	—	—	—	—	0.0008	0.0002	0.0005	—	0.0032	0.0011	0.0017	—	0.0020	0.0011	0.0020
	MIRAGE	—	—	—	—	0.0108	0.0006	0.0076	—	0.0040	0.0009	0.0027	—	0.0105	0.0008	**0.0072**
	TokenShapley	—	—	—	—	0.0008	0.0000	0.0004	—	0.0006	0.0001	0.0003	—	0.0005	0.0001	0.0003
MS	AttnUnionDep	**0.0045**	**0.0020**	**0.0029**	**0.0096**	**0.1100**	**0.0107**	**0.0811**	**0.0034**	0.0313	**0.0064**	**0.0243**	**0.0094**	0.0442	**0.0038**	**0.0352**
	AttriBoT	0.0000	0.0000	0.0000	0.0001	0.0016	0.0003	0.0013	0.0001	0.0011	0.0003	0.0007	0.0001	0.0012	0.0002	0.0010
	ContextCite	0.0001	0.0001	0.0000	0.0002	0.0020	0.0003	0.0019	0.0001	0.0011	0.0003	0.0008	0.0002	0.0013	0.0002	0.0013
	HSSAVG	0.0003	0.0004	0.0001	0.0002	0.0025	0.0002	0.0014	0.0004	0.0008	0.0002	0.0009	0.0010	0.0010	0.0001	0.0025
	MIRAGE	0.0006	0.0004	0.0008	0.0027	0.0573	0.0026	0.0312	0.0007	**0.0390**	0.0021	0.0131	0.0026	**0.0558**	0.0028	0.0311
	TokenShapley	0.0000	0.0000	0.0000	0.0000	0.0006	0.0001	0.0005	0.0000	0.0002	0.0001	0.0002	0.0000	0.0003	0.0000	0.0004
VJ	AttnUnionDep	**0.0124**	0.0014	**0.0069**	**0.0114**	**0.0429**	**0.0022**	**0.0251**	**0.0069**	**0.0381**	**0.0102**	**0.0240**	**0.0154**	0.0275	0.0035	**0.0145**
	AttriBoT	0.0003	0.0002	0.0002	0.0004	0.0018	0.0002	0.0013	0.0002	0.0010	0.0003	0.0006	0.0005	0.0007	0.0001	0.0005
	ContextCite	0.0001	0.0001	0.0001	0.0002	0.0010	0.0001	0.0005	0.0001	0.0009	0.0004	0.0003	0.0003	0.0007	0.0003	0.0002
	HSSAVG	0.0039	**0.0031**	0.0022	0.0006	0.0021	0.0001	0.0010	0.0028	0.0141	0.0041	0.0075	0.0022	0.0116	**0.0049**	0.0084
	MIRAGE	0.0030	0.0007	0.0022	0.0040	0.0286	0.0009	0.0110	0.0018	0.0132	0.0031	0.0040	0.0049	**0.0322**	0.0017	0.0125
	TokenShapley	0.0004	0.0000	0.0002	0.0003	0.0011	0.0002	0.0010	0.0003	0.0013	0.0003	0.0008	0.0002	0.0007	0.0001	0.0007

Table 11: Variance of performance metrics across models per dataset.

Dataset	Model	Method	Shared Metrics			Top-1				Top-3				KMeans			
			AUPRC	AUROC	NDCG	F1	BART	NLI	LogDrop	F1	BART	NLI	LogDrop	F1	BART	NLI	LogDrop
VJ	LLAMA	AttnUnionDep	**0.745**	0.906	**0.842**	**0.632**	**-2.076**	**0.360**	**1.741**	**0.576**	**-1.606**	**0.728**	**2.143**	**0.716**	-1.684	0.605	2.063
		AttriBoT	0.716	0.803	0.794	0.589	-2.089	0.332	1.722	0.519	-1.632	0.672	2.089	0.673	-1.717	0.545	2.013
		ContextCite	0.668	0.833	0.784	0.593	-2.082	0.335	1.725	0.529	-1.618	0.682	2.100	0.668	-1.745	0.530	1.998
		HSSAVG	0.700	**0.920**	0.807	0.554	-2.185	0.331	1.631	0.538	-1.683	0.685	2.056	0.208	-1.712	**0.799**	**2.115**
		MIRAGE	0.469	0.778	0.636	0.341	-2.644	0.206	1.320	0.428	-1.884	0.549	1.942	0.388	-2.420	0.295	1.505
		TokenShapley	0.651	0.826	0.772	0.556	-2.144	0.321	1.671	0.505	-1.673	0.655	2.073	0.626	**-1.673**	0.629	2.055
	MISTRAL	AttnUnionDep	0.618	0.865	0.750	0.510	-2.340	0.292	1.536	0.480	-1.815	0.605	1.971	0.573	-1.913	0.520	1.886
		AttriBoT	**0.719**	0.817	0.797	0.592	-2.121	0.326	1.699	0.521	-1.648	0.681	2.069	**0.672**	-1.715	0.558	2.024
		ContextCite	0.662	0.833	0.779	**0.593**	-2.087	**0.345**	1.729	0.520	-1.639	0.674	2.087	0.662	-1.728	0.543	2.012
		HSSAVG	0.647	**0.916**	0.770	0.509	-2.299	0.310	1.555	0.498	-1.748	0.653	2.005	0.207	-1.721	**0.805**	2.100
		MIRAGE	0.458	0.782	0.625	0.332	-2.697	0.212	1.354	0.405	-1.921	0.541	1.966	0.368	-2.470	0.304	1.526
		TokenShapley	0.687	0.839	**0.799**	0.587	**-2.080**	0.335	**1.733**	**0.540**	**-1.604**	**0.696**	**2.126**	0.658	**-1.612**	0.648	**2.110**
	PHI	AttnUnionDep	**0.766**	**0.924**	**0.858**	**0.643**	**-2.051**	**0.364**	**1.756**	**0.576**	**-1.598**	**0.734**	**2.150**	**0.732**	-1.637	0.637	2.090
		AttriBoT	0.731	0.817	0.805	0.600	-2.085	0.345	1.731	0.528	-1.633	0.691	2.092	0.685	-1.718	0.556	2.020
		ContextCite	0.678	0.839	0.793	0.612	-2.060	0.354	1.742	0.540	-1.602	0.698	2.107	0.689	-1.717	0.557	2.014
		HSSAVG	0.576	0.830	0.721	0.510	-2.239	0.332	1.594	0.433	-1.918	0.564	1.894	0.293	-1.842	**0.695**	2.001
		MIRAGE	0.539	0.800	0.695	0.438	-2.435	0.253	1.489	0.465	-1.754	0.611	2.036	0.483	-2.212	0.352	1.668
		TokenShapley	0.693	0.839	0.801	0.596	-2.060	0.358	1.743	0.536	-1.600	0.698	2.121	0.654	**-1.626**	0.640	**2.102**
	QWEN	AttnUnionDep	**0.734**	0.905	**0.835**	**0.628**	**-2.057**	**0.362**	**1.758**	**0.554**	**-1.592**	**0.720**	**2.156**	**0.700**	**-1.662**	0.602	2.077
		AttriBoT	0.695	0.792	0.775	0.559	-2.175	0.312	1.660	0.497	-1.693	0.649	2.045	0.643	-1.755	0.546	2.000
		ContextCite	0.658	0.827	0.776	0.586	-2.117	0.337	1.704	0.513	-1.655	0.662	2.070	0.652	-1.729	0.547	2.010
		HSSAVG	0.685	**0.907**	0.798	0.564	-2.195	0.340	1.629	0.521	-1.729	0.658	2.043	0.217	-1.698	**0.796**	**2.110**
		MIRAGE	0.470	0.776	0.635	0.367	-2.694	0.210	1.309	0.407	-1.940	0.510	1.912	0.395	-2.490	0.283	1.461
		TokenShapley	0.650	0.824	0.772	0.563	-2.114	0.341	1.690	0.509	-1.666	0.672	2.080	0.639	-1.666	0.627	2.057
	QWEN32	AttnUnionDep	0.503	**0.834**	0.661	0.397	-2.507	0.264	1.412	0.382	-2.036	0.505	1.805	0.441	-2.000	0.502	1.834
		AttriBoT	**0.693**	0.789	0.771	0.559	-2.169	0.310	1.651	0.506	-1.698	0.651	2.036	0.634	-1.773	0.540	1.967
		ContextCite	0.647	0.819	0.767	0.572	-2.142	0.322	1.687	0.510	-1.677	0.645	2.068	0.640	-1.783	0.509	1.982
		HSSAVG	0.563	0.798	0.702	0.528	-2.218	0.318	1.613	0.420	-1.946	0.539	1.868	0.163	-1.947	**0.656**	1.905
		MIRAGE	0.385	0.729	0.562	0.263	-2.910	0.167	1.197	0.347	-2.074	0.460	1.867	0.286	-2.714	0.237	1.360
		TokenShapley	0.664	0.831	**0.782**	**0.575**	**-2.111**	**0.344**	**1.690**	**0.517**	**-1.663**	**0.672**	**2.068**	**0.650**	**-1.661**	0.642	**2.063**

Table 12: Comprehensive Performance Metrics (VJ)

Dataset	Model	Method	Shared Metrics			Top-1				Top-3				KMeans			
			AUPRC	AUROC	NDCG	F1	BART	NLI	LogDrop	F1	BART	NLI	LogDrop	F1	BART	NLI	LogDrop
MS	LLAMA	AttnUnionDep	0.381	0.567	0.530	0.521	-2.872	0.502	1.802	0.445	-2.695	0.651	1.989	**0.521**	-2.670	0.625	1.959
		AttriBoT	0.391	0.500	0.522	0.510	-2.846	0.509	1.837	0.383	-2.681	0.645	2.007	0.510	-2.659	0.608	1.996
		ContextCite	0.381	0.542	0.530	0.513	-2.810	0.520	1.870	0.385	-2.662	0.648	2.017	0.510	-2.648	0.609	2.007
		HSSAVG	**0.451**	**0.629**	**0.584**	**0.536**	**-2.725**	**0.557**	**1.957**	**0.452**	**-2.604**	**0.695**	**2.147**	0.364	-2.669	**0.712**	**2.086**
		MIRAGE	0.293	0.511	0.459	0.384	-3.331	0.343	1.404	0.398	-2.807	0.550	1.875	0.391	-3.216	0.381	1.505
		TokenShapley	0.363	0.514	0.521	0.514	-2.796	0.514	1.871	0.410	-2.651	0.650	2.048	0.513	**-2.630**	0.634	2.015
	MISTRAL	AttnUnionDep	0.303	0.516	0.482	0.390	-3.357	0.347	1.401	0.373	-2.970	0.534	1.772	0.390	-3.005	0.530	1.680
		AttriBoT	0.387	0.497	0.524	0.496	-2.893	0.493	1.797	0.375	-2.711	0.629	1.970	0.501	-2.685	0.601	1.972
		ContextCite	0.374	0.534	0.532	0.504	-2.839	0.508	1.830	0.373	-2.687	0.635	1.999	0.504	-2.662	0.603	1.985
		HSSAVG	**0.433**	**0.619**	**0.577**	0.503	-2.844	0.517	1.869	**0.438**	-2.648	**0.679**	**2.113**	0.343	-2.672	**0.710**	**2.080**
		MIRAGE	0.325	0.538	0.521	0.425	-2.969	0.390	1.546	0.431	**-2.405**	0.617	2.072	0.432	-2.860	0.429	1.644
		TokenShapley	0.366	0.516	0.535	**0.519**	**-2.748**	**0.529**	**1.909**	0.413	-2.622	0.667	2.074	**0.514**	**-2.591**	0.643	2.050
	PHI	AttnUnionDep	**0.444**	**0.612**	**0.592**	**0.554**	-2.770	**0.536**	1.915	**0.477**	-2.628	**0.683**	2.082	**0.561**	-2.608	0.635	**2.053**
		AttriBoT	0.380	0.492	0.519	0.496	-2.871	0.501	1.806	0.370	-2.698	0.635	1.985	0.501	-2.674	0.607	1.968
		ContextCite	0.369	0.534	0.534	0.509	-2.806	0.521	1.865	0.386	-2.654	0.650	2.013	0.507	-2.652	0.609	2.002
		HSSAVG	0.427	0.605	0.575	0.511	-2.769	0.534	**1.922**	0.429	-2.659	0.666	**2.084**	0.366	-2.695	**0.694**	2.037
		MIRAGE	0.324	0.527	0.518	0.466	-2.961	0.441	1.688	0.422	-2.628	0.609	2.000	0.473	-2.851	0.483	1.785
		TokenShapley	0.365	0.514	0.534	0.515	**-2.751**	0.530	1.913	0.410	**-2.624**	0.664	2.074	0.510	**-2.605**	0.637	2.034
	QWEN	AttnUnionDep	0.409	0.589	0.566	0.527	-2.796	0.517	1.853	0.448	-2.663	0.658	2.031	**0.535**	**-2.618**	0.639	1.996
		AttriBoT	0.377	0.484	0.513	0.484	-2.928	0.473	1.754	0.368	-2.749	0.609	1.947	0.484	-2.728	0.577	1.926
		ContextCite	0.363	0.521	0.522	0.485	-2.884	0.490	1.784	0.361	-2.715	0.615	1.959	0.487	-2.703	0.583	1.945
		HSSAVG	**0.452**	**0.628**	**0.592**	**0.531**	**-2.741**	**0.545**	**1.946**	**0.457**	**-2.626**	**0.683**	**2.121**	0.386	-2.672	**0.703**	**2.073**
		MIRAGE	0.276	0.494	0.472	0.345	-3.401	0.330	1.295	0.372	-2.830	0.533	1.847	0.354	-3.279	0.372	1.400
		TokenShapley	0.362	0.516	0.533	0.514	-2.798	0.515	1.870	0.407	-2.652	0.651	2.054	0.509	-2.634	0.633	2.003
	QWEN32	AttnUnionDep	0.289	0.512	0.465	0.331	-3.463	0.316	1.293	0.338	-2.993	0.507	1.741	0.343	-3.022	0.512	1.653
		AttriBoT	0.376	0.486	0.513	0.484	-2.941	0.473	1.751	0.362	-2.760	0.606	1.941	0.482	-2.738	0.580	1.924
		ContextCite	0.363	0.523	0.523	0.485	-2.907	0.483	1.778	0.362	-2.730	0.612	1.958	0.481	-2.730	0.576	1.927
		HSSAVG	**0.408**	**0.584**	**0.560**	**0.517**	**-2.723**	**0.548**	**1.956**	0.407	-2.679	**0.658**	**2.072**	0.303	-2.744	**0.690**	1.966
		MIRAGE	0.275	0.492	0.471	0.348	-3.452	0.316	1.268	0.373	-2.890	0.517	1.795	0.355	-3.342	0.352	1.359
		TokenShapley	0.360	0.518	0.532	0.512	-2.793	0.517	1.870	**0.410**	**-2.649**	0.654	2.053	**0.511**	**-2.626**	0.639	**2.000**

Table 13: Comprehensive Performance Metrics (MS)

Dataset	Model	Method	Shared Metrics			Top-1				Top-3				KMeans			
			AUPRC	AUROC	NDCG	F1	BART	NLI	LogDrop	F1	BART	NLI	LogDrop	F1	BART	NLI	LogDrop
EQA	LLAMA	AttnUnionDep	—	—	—	—	-3.425	**0.198**	0.982	—	-3.244	**0.350**	1.110	—	-3.291	0.312	1.070
		AttriBoT	—	—	—	—	-3.525	0.157	0.901	—	-3.343	0.265	1.030	—	-3.367	0.231	1.008
		ContextCite	—	—	—	—	-3.534	0.159	0.894	—	-3.341	0.261	1.030	—	-3.394	0.221	0.992
		HSSAVG	—	—	—	—	**-3.379**	0.193	**1.013**	—	**-3.212**	0.350	**1.136**	—	-3.287	**0.410**	1.043
		MIRAGE	—	—	—	—	-3.764	0.093	0.731	—	-3.412	0.221	0.999	—	-3.676	0.122	0.795
		TokenShapley	—	—	—	—	-3.444	0.156	0.957	—	-3.263	0.280	1.093	—	**-3.262**	0.299	**1.075**
	MISTRAL	AttnUnionDep	—	—	—	—	-3.567	0.152	0.879	—	-3.359	0.279	1.029	—	-3.412	0.263	0.979
		AttriBoT	—	—	—	—	-3.537	0.150	0.895	—	-3.363	0.252	1.019	—	-3.382	0.225	1.001
		ContextCite	—	—	—	—	-3.538	0.153	0.891	—	-3.356	0.250	1.023	—	-3.371	0.225	1.008
		HSSAVG	—	—	—	—	-3.436	0.169	0.973	—	-3.242	**0.321**	1.112	—	-3.266	**0.396**	1.063
		MIRAGE	—	—	—	—	-3.691	0.121	0.809	—	-3.333	0.252	1.074	—	-3.603	0.149	0.875
		TokenShapley	—	—	—	—	**-3.394**	**0.169**	**0.995**	—	**-3.222**	0.304	**1.122**	—	**-3.225**	0.315	**1.106**
	PHI	AttnUnionDep	—	—	—	—	**-3.370**	**0.207**	**1.025**	—	**-3.196**	**0.375**	**1.149**	—	-3.231	0.321	**1.123**
		AttriBoT	—	—	—	—	-3.533	0.153	0.896	—	-3.360	0.258	1.021	—	-3.373	0.228	0.999
		ContextCite	—	—	—	—	-3.530	0.159	0.902	—	-3.349	0.262	1.032	—	-3.414	0.243	0.972
		HSSAVG	—	—	—	—	-3.429	0.178	0.979	—	-3.306	0.295	1.069	—	-3.313	**0.348**	1.032
		MIRAGE	—	—	—	—	-3.608	0.133	0.850	—	-3.337	0.258	1.046	—	-3.508	0.175	0.924
		TokenShapley	—	—	—	—	-3.391	0.167	0.998	—	-3.217	0.303	1.125	—	**-3.220**	0.317	1.111
	QWEN	AttnUnionDep	—	—	—	—	-3.406	0.185	0.990	—	**-3.231**	**0.329**	**1.118**	—	-3.277	0.290	**1.076**
		AttriBoT	—	—	—	—	-3.552	0.148	0.882	—	-3.371	0.248	1.010	—	-3.379	0.229	0.996
		ContextCite	—	—	—	—	-3.544	0.155	0.885	—	-3.356	0.250	1.020	—	-3.375	0.222	1.005
		HSSAVG	—	—	—	—	**-3.400**	**0.186**	**1.000**	—	-3.257	0.321	1.106	—	-3.313	**0.394**	1.023
		MIRAGE	—	—	—	—	-3.872	0.080	0.638	—	-3.449	0.196	0.963	—	-3.763	0.109	0.719
		TokenShapley	—	—	—	—	-3.448	0.158	0.959	—	-3.272	0.280	1.091	—	**-3.265**	0.294	1.075
	QWEN32	AttnUnionDep	—	—	—	—	-3.689	0.103	0.783	—	-3.426	0.216	0.977	—	-3.483	0.207	0.926
		AttriBoT	—	—	—	—	-3.564	0.143	0.876	—	-3.378	0.242	1.008	—	-3.392	0.224	0.988
		ContextCite	—	—	—	—	-3.558	0.150	0.875	—	-3.362	0.247	1.019	—	-3.399	0.216	0.992
		HSSAVG	—	—	—	—	-3.448	0.162	0.959	—	-3.356	0.261	1.030	—	-3.385	**0.333**	0.945
		MIRAGE	—	—	—	—	-3.817	0.083	0.684	—	-3.470	0.194	0.954	—	-3.729	0.109	0.753
		TokenShapley	—	—	—	—	**-3.440**	**0.164**	**0.964**	—	**-3.259**	**0.290**	**1.096**	—	**-3.263**	0.302	**1.077**

Table 14: Comprehensive Performance Metrics (EQA)

Dataset	Model	Method	Shared Metrics			Top-1				Top-3				KMeans			
			AUPRC	AUROC	NDCG	F1	BART	NLI	LogDrop	F1	BART	NLI	LogDrop	F1	BART	NLI	LogDrop
ASQA	LLAMA	AttnUnionDep	—	—	—	—	-2.534	**0.123**	**1.058**	—	**-1.988**	**0.409**	**1.472**	—	-2.063	0.341	1.420
		AttriBoT	—	—	—	—	-2.539	0.106	1.009	—	-2.051	0.344	1.373	—	-2.152	0.260	1.312
		ContextCite	—	—	—	—	**-2.527**	0.110	1.030	—	-2.046	0.342	1.387	—	-2.211	0.241	1.276
		HSSAVG	—	—	—	—	-2.644	0.101	0.909	—	-2.146	0.334	1.283	—	**-1.891**	**0.674**	**1.458**
		MIRAGE	—	—	—	—	-2.742	0.089	0.918	—	-2.119	0.325	1.389	—	-2.478	0.167	1.117
		TokenShapley	—	—	—	—	-2.567	0.103	0.979	—	-2.100	0.317	1.324	—	-2.040	0.366	1.376
	MISTRAL	AttnUnionDep	—	—	—	—	-2.681	0.097	0.948	—	-2.212	0.298	1.298	—	-2.276	0.261	1.249
		AttriBoT	—	—	—	—	-2.561	0.098	0.984	—	-2.090	0.329	1.334	—	-2.189	0.247	1.273
		ContextCite	—	—	—	—	-2.563	0.105	0.992	—	-2.100	0.318	1.328	—	-2.245	0.228	1.232
		HSSAVG	—	—	—	—	-2.729	0.085	0.839	—	-2.303	0.266	1.153	—	-1.948	**0.685**	1.395
		MIRAGE	—	—	—	—	-2.786	0.081	0.896	—	-2.161	0.292	1.379	—	-2.566	0.140	1.063
		TokenShapley	—	—	—	—	**-2.505**	**0.112**	**1.018**	—	**-1.994**	**0.362**	**1.400**	—	**-1.938**	0.403	**1.452**
	PHI	AttnUnionDep	—	—	—	—	**-2.502**	**0.126**	**1.082**	—	**-1.952**	**0.429**	**1.498**	—	-1.971	0.391	**1.497**
		AttriBoT	—	—	—	—	-2.573	0.098	0.977	—	-2.111	0.319	1.322	—	-2.203	0.240	1.264
		ContextCite	—	—	—	—	-2.549	0.107	1.004	—	-2.098	0.323	1.340	—	-2.287	0.250	1.206
		HSSAVG	—	—	—	—	-2.658	0.085	0.892	—	-2.333	0.247	1.128	—	-2.154	0.392	1.284
		MIRAGE	—	—	—	—	-2.687	0.092	0.954	—	-2.093	0.335	1.404	—	-2.433	0.180	1.148
		TokenShapley	—	—	—	—	-2.504	0.114	1.021	—	-1.998	0.355	1.397	—	**-1.931**	**0.410**	1.455
	QWEN	AttnUnionDep	—	—	—	—	**-2.511**	**0.121**	**1.085**	—	**-1.966**	**0.414**	**1.498**	—	-2.012	0.364	**1.475**
		AttriBoT	—	—	—	—	-2.576	0.100	0.987	—	-2.119	0.317	1.328	—	-2.191	0.251	1.286
		ContextCite	—	—	—	—	-2.564	0.103	1.012	—	-2.104	0.316	1.349	—	-2.221	0.230	1.275
		HSSAVG	—	—	—	—	-2.673	0.088	0.887	—	-2.286	0.271	1.175	—	**-2.003**	**0.517**	1.396
		MIRAGE	—	—	—	—	-2.823	0.081	0.859	—	-2.159	0.299	1.359	—	-2.620	0.131	1.009
		TokenShapley	—	—	—	—	-2.552	0.109	0.998	—	-2.075	0.337	1.349	—	-2.012	0.375	1.406
	QWEN32	AttnUnionDep	—	—	—	—	-2.807	0.066	0.861	—	-2.325	0.230	1.214	—	-2.255	0.270	1.272
		AttriBoT	—	—	—	—	-2.599	0.095	0.974	—	-2.138	0.304	1.319	—	-2.206	0.243	1.275
		ContextCite	—	—	—	—	-2.598	0.099	0.991	—	-2.152	0.294	1.320	—	-2.262	0.216	1.250
		HSSAVG	—	—	—	—	-2.771	0.064	0.818	—	-2.553	0.165	0.977	—	-2.400	0.330	1.074
		MIRAGE	—	—	—	—	-2.873	0.079	0.819	—	-2.263	0.270	1.278	—	-2.710	0.123	0.939
		TokenShapley	—	—	—	—	**-2.537**	**0.107**	**1.016**	—	**-2.051**	**0.344**	**1.372**	—	**-2.004**	**0.375**	**1.419**

Table 15: Comprehensive Performance Metrics (ASQA)

		Pairwise Performance				Human Evaluation (1-7)		
Rank	Method	Wins	Ties	Losses	Score	Trust	Useful	Understand
1	AttnUnionDep	57	27	16	0.4100	4.74	4.86	4.86
2	HSSAVG	42	31	27	0.1500	4.60	4.81	4.92
3	TokenShapley	44	26	30	0.1400	4.67	4.83	5.07
4	ContextCite	41	30	29	0.1200	4.59	4.65	4.84
5	AttriBoT	32	31	37	-0.0500	4.35	4.44	4.80
6	MIRAGE	4	15	81	-0.7700	3.69	3.59	4.32

Table 16: Comprehensive evaluation: Pairwise performance and average Likert scores.

Method	LLAMA	MISTRAL	PHI	PHIH200	QWEN	QWEN32
AttnUnionDep	31 / 624 (4.97%)	34 / 624 (5.45%)	55 / 624 (8.81%)	28 / 624 (4.49%)	30 / 624 (4.81%)	36 / 624 (5.77%)
AttriBoT	2 / 624 (0.32%)	0 / 624 (0.00%)	9 / 624 (1.44%)	0 / 624 (0.00%)	4 / 624 (0.64%)	8 / 624 (1.28%)
ContextCite	9 / 624 (1.44%)	2 / 624 (0.32%)	19 / 624 (3.04%)	2 / 624 (0.32%)	11 / 624 (1.76%)	15 / 624 (2.40%)
HSSAVG	0 / 624 (0.00%)	0 / 624 (0.00%)	8 / 624 (1.28%)	0 / 624 (0.00%)	0 / 624 (0.00%)	2 / 624 (0.32%)
MIRAGE	61 / 624 (9.78%)	227 / 622 (36.50%)	209 / 623 (33.55%)	75 / 622 (12.06%)	90 / 622 (14.47%)	239 / 624 (38.30%)
TokenShapley	0 / 624 (0.00%)	0 / 624 (0.00%)	8 / 624 (1.28%)	0 / 624 (0.00%)	2 / 624 (0.32%)	5 / 624 (0.80%)

Table 17: Unscored (failed) examples vs. total evaluated examples per attribution method for the **VJ** dataset. (Total Evaluated: 22,457)

Method	LLAMA	MISTRAL	PHI	PHIH200	QWEN	QWEN32
AttnUnionDep	31 / 5,412 (0.57%)	38 / 5,412 (0.70%)	61 / 5,412 (1.13%)	29 / 5,412 (0.54%)	46 / 5,412 (0.85%)	43 / 5,412 (0.79%)
AttriBoT	5 / 5,412 (0.09%)	1 / 5,412 (0.02%)	8 / 5,412 (0.15%)	1 / 5,412 (0.02%)	8 / 5,412 (0.15%)	7 / 5,412 (0.13%)
ContextCite	8 / 5,412 (0.15%)	5 / 5,412 (0.09%)	18 / 5,412 (0.33%)	5 / 5,412 (0.09%)	11 / 5,412 (0.20%)	13 / 5,412 (0.24%)
HSSAVG	1 / 5,412 (0.02%)	1 / 5,412 (0.02%)	7 / 5,412 (0.13%)	1 / 5,412 (0.02%)	3 / 5,412 (0.06%)	5 / 5,412 (0.09%)
MIRAGE	216 / 5,412 (3.99%)	2,052 / 5,411 (37.92%)	541 / 5,406 (10.01%)	456 / 5,405 (8.44%)	256 / 5,410 (4.73%)	512 / 5,400 (9.48%)
TokenShapley	3 / 5,412 (0.06%)	1 / 5,412 (0.02%)	8 / 5,412 (0.15%)	1 / 5,412 (0.02%)	7 / 5,412 (0.13%)	6 / 5,412 (0.11%)

Table 18: Unscored (failed) examples vs. total evaluated examples per attribution method for the **MS** dataset. (Total Evaluated: 194,804)

Method	LLAMA	MISTRAL	PHI	QWEN	QWEN32
AttnUnionDep	318 / 5,792 (5.49%)	384 / 5,792 (6.63%)	499 / 5,791 (8.62%)	343 / 5,792 (5.92%)	684 / 5,792 (11.81%)
AttriBoT	36 / 5,792 (0.62%)	18 / 5,792 (0.31%)	46 / 5,791 (0.79%)	40 / 5,792 (0.69%)	144 / 5,792 (2.49%)
ContextCite	47 / 5,792 (0.81%)	30 / 5,792 (0.52%)	78 / 5,791 (1.35%)	54 / 5,792 (0.93%)	257 / 5,792 (4.44%)
HSSAVG	22 / 5,792 (0.38%)	6 / 5,792 (0.10%)	44 / 5,791 (0.76%)	24 / 5,792 (0.41%)	86 / 5,792 (1.48%)
MIRAGE	374 / 5,792 (6.46%)	2,396 / 5,790 (41.38%)	948 / 5,786 (16.38%)	458 / 5,790 (7.91%)	1,996 / 5,791 (34.47%)
TokenShapley	28 / 5,792 (0.48%)	16 / 5,792 (0.28%)	47 / 5,791 (0.81%)	33 / 5,792 (0.57%)	124 / 5,792 (2.14%)

Table 19: Unscored (failed) examples vs. total evaluated examples per attribution method for the **EQA** dataset. (Total Evaluated: 173,744)

Method	LLAMA	MISTRAL	PHI	PHIH200	QWEN	QWEN32
AttnUnionDep	1,301 / 3,820 (34.06%)	1,450 / 3,820 (37.96%)	1,746 / 3,820 (45.71%)	1,251 / 3,820 (32.75%)	1,510 / 3,820 (39.53%)	1,641 / 3,820 (42.96%)
AttriBoT	149 / 3,820 (3.90%)	24 / 3,820 (0.63%)	404 / 3,820 (10.58%)	34 / 3,820 (0.89%)	225 / 3,820 (5.89%)	390 / 3,820 (10.21%)
ContextCite	371 / 3,820 (9.71%)	142 / 3,820 (3.72%)	871 / 3,820 (22.80%)	159 / 3,820 (4.16%)	530 / 3,820 (13.87%)	722 / 3,820 (18.90%)
HSSAVG	64 / 3,820 (1.68%)	6 / 3,820 (0.16%)	390 / 3,820 (10.21%)	38 / 3,820 (0.99%)	136 / 3,820 (3.56%)	267 / 3,820 (6.99%)
MIRAGE	1,768 / 3,820 (46.28%)	2,508 / 3,820 (65.65%)	2,773 / 3,819 (72.61%)	1,866 / 3,816 (48.90%)	2,020 / 3,819 (52.89%)	2,993 / 3,814 (78.47%)
TokenShapley	104 / 3,820 (2.72%)	13 / 3,820 (0.34%)	458 / 3,820 (11.99%)	54 / 3,820 (1.41%)	250 / 3,820 (6.54%)	362 / 3,820 (9.48%)

Table 20: Unscored (failed) examples vs. total evaluated examples per attribution method for the **ASQA** dataset. (Total Evaluated: 137,508)

Method	LLAMA (A100-80GB)	MISTRAL (A100-80GB)	PHI (H200)	QWEN (A100-80GB)	QWEN32 (H200)
AttnUnionDep	2.08s / 25.37GB	2.16s / 24.69GB	2.18s / 41.32GB	1.82s / 25.10GB	1.77s / 75.48GB
AttriBoT	10.00s / 22.14GB	13.48s / 18.48GB	11.29s / 34.97GB	8.84s / 22.24GB	11.78s / 72.63GB
ContextCite	14.07s / 23.57GB	16.90s / 19.98GB	13.93s / 37.70GB	13.31s / 23.07GB	21.41s / 75.70GB
HSSAVG	3.55s / 21.25GB	5.84s / 17.93GB	0.88s / 33.95GB	2.25s / 21.33GB	1.47s / 71.51GB
MIRAGE	40.20s / 38.52GB	69.46s / 38.84GB	61.20s / 68.79GB	70.90s / 39.93GB	24.22s / 114.73GB
TokenShapley	1.01s / 21.07GB	1.17s / 17.86GB	0.89s / 33.77GB	0.95s / 21.15GB	1.38s / 71.22GB

Table 21: Average execution runtime and peak VRAM allocation per attribution method for the **VJ** dataset. Format: [Runtime in seconds] / [Peak VRAM in GB].

Method	LLAMA (A100-80GB)	MISTRAL (A100-80GB)	PHI (H200)	QWEN (A100-80GB)	QWEN32 (H200)
AttnUnionDep	0.76s / 18.49GB	1.17s / 17.00GB	0.68s / 30.33GB	0.70s / 18.91GB	0.63s / 67.15GB
AttriBoT	3.01s / 18.31GB	4.18s / 16.21GB	3.42s / 30.09GB	2.91s / 18.82GB	4.21s / 67.29GB
ContextCite	6.48s / 19.30GB	6.83s / 16.75GB	6.57s / 30.93GB	6.53s / 19.63GB	11.83s / 69.09GB
HSSAVG	1.23s / 18.04GB	1.97s / 16.04GB	0.50s / 29.63GB	0.82s / 18.56GB	0.65s / 66.90GB
MIRAGE	9.24s / 22.28GB	14.11s / 21.37GB	14.69s / 39.27GB	12.40s / 22.67GB	15.01s / 82.68GB
TokenShapley	0.47s / 17.95GB	0.65s / 16.00GB	0.41s / 29.52GB	0.46s / 18.46GB	0.53s / 66.76GB

Table 22: Average execution runtime and peak VRAM allocation per attribution method for the **MS** dataset. Format: [Runtime in seconds] / [Peak VRAM in GB].

Method	LLAMA (A100-80GB)	MISTRAL (A100-80GB)	PHI (H200)	QWEN (A100-80GB)	QWEN32 (H200)
AttnUnionDep	2.61s / 30.83GB	2.12s / 30.38GB	2.33s / 43.80GB	2.00s / 31.81GB	1.64s / 77.49GB
AttriBoT	10.48s / 23.50GB	19.67s / 19.46GB	17.04s / 36.15GB	8.96s / 25.56GB	15.76s / 75.39GB
ContextCite	8.12s / 24.64GB	9.26s / 20.68GB	14.98s / 38.58GB	7.41s / 26.10GB	22.00s / 77.82GB
HSSAVG	1.59s / 22.36GB	1.58s / 18.70GB	0.98s / 35.09GB	1.18s / 24.44GB	1.59s / 74.03GB
MIRAGE	13.14s / 43.58GB	23.54s / 43.89GB	26.66s / 67.24GB	17.44s / 45.34GB	21.02s / 108.84GB
TokenShapley	0.92s / 22.13GB	0.93s / 18.60GB	1.15s / 34.87GB	0.78s / 24.22GB	1.59s / 73.69GB

Table 23: Average execution runtime and peak VRAM allocation per attribution method for the **EQA** dataset. Format: [Runtime in seconds] / [Peak VRAM in GB]. **Note:** All evaluations for the EQA dataset were executed on H200 GPUs, despite being grouped under their primary A100-80GB labels for cross-dataset consistency.

Method	LLAMA (A100-80GB)	MISTRAL (A100-80GB)	PHI (H200)	QWEN (A100-80GB)	QWEN32 (H200)
AttnUnionDep	3.51s / 41.52GB	4.29s / 41.32GB	4.48s / 68.95GB	2.94s / 38.97GB	2.91s / 92.56GB
AttriBoT	42.06s / 31.95GB	84.11s / 25.05GB	67.27s / 47.89GB	36.14s / 32.77GB	41.89s / 88.84GB
ContextCite	30.53s / 33.96GB	43.65s / 28.26GB	34.49s / 54.77GB	27.43s / 34.18GB	36.41s / 93.74GB
HSSAVG	9.07s / 29.94GB	16.27s / 23.59GB	2.02s / 46.30GB	4.74s / 31.04GB	3.72s / 86.50GB
MIRAGE	64.52s / 59.44GB	96.11s / 60.64GB	68.13s / 106.09GB	79.05s / 61.27GB	20.60s / 130.41GB
TokenShapley	2.77s / 29.33GB	3.56s / 23.36GB	2.82s / 45.71GB	2.65s / 30.49GB	3.93s / 85.74GB

Table 24: Average execution runtime and peak VRAM allocation per attribution method for the **ASQA** dataset. Format: [Runtime in seconds] / [Peak VRAM in GB].

GLOBAL PIPELINE & HARDWARE			
Scope	**Parameter**	**Type / Default**	**Functional Description / Role**
Pipeline	`-model_id`	String (Req.)	HuggingFace repo ID or absolute local path to the target LLM.
	`-dataset_jsonl`	Path (Req.)	Path to the evaluation file containing queries and responses.
	`-dataset_txt_dir`	Path (Req.)	Directory containing the scraped, raw source text files.
	`-output_dir`	Path (Req.)	Root directory where scores and hardware metrics are saved.
	`-methods`	List (Req.)	List of attributors to run (`cc`, `hssavg`, `attribot`, etc.).
	`-local_files_only`	Boolean (`False`)	Forces loading models strictly from local cache to avoid Hub queries.
	`-start_idx`	Integer (`0`)	Starting data index for distributed processing and chunking.
	`-end_idx`	Integer (`None`)	Ending data index (exclusive) for the processing block.
Hardware	`BATCH_SIZE`	Integer (`20`)	Samples processed before flushing memory buffers and saving `.json`.
	`torch_dtype`	`torch.bfloat16`	Forced 16-bit precision to halve parametric VRAM footprint.
	`attn_impl`	`flash_attention_2`	Enforced as the global default layout; dynamically hot-swapped to SDPA/Eager mode when specific mathematical matrix extractions are required.
	`expandable_seg`	Boolean (`True`)	Prevents PyTorch memory block splitting, lowering OOM crash rates.
	`Filters`	≥ 3 words, > 15 chars	Rejects overly short text segments to reduce web-scraping noise.

Table 25: Centralized evaluation pipeline parameters, enclosing global command-line arguments (CLI), resource configurations, and hardware optimization boundaries.

ATTNUNIONDEP			
Scope	**Parameter**	**Type / Default**	**Functional Description / Role**
Parsing	`lal_parser`	`"./best_parser.pt"`	Local path to pre-trained LAL-Parser weights for dependency tree modeling.
	`layer_idx`	`(total_layers // 2) + 1`	Dynamically targets the mid-upper attention layer to extract raw probability matrices.
	`k`	Integer (`10`)	Number of top source tokens collected per response token (AttnUnion step).
	`tau` (τ)	Integer (`2`)	Distance window threshold restricting isolated noise tokens without neighbors.
Engine	`torch_dtype`	`torch.bfloat16`	Forced 16-bit precision to maintain structural gradient integrity.
	`attn_impl`	`sdpa` (Swapped)	Temporarily hot-swapped via Swapper to extract unmasked self-attention.
	`model_type`	Template Strings	Dynamically configures `system_prefix` and `q_suffix` based on core LLM setup.
	`sep_ids`	`["\n"]`	Injects newline IDs to prevent attention smoothing across physical sentence bounds.
	`is_junk`	Digits, Punct., Brackets	Eliminates non-semantic characters from the active score mask.

Table 26: Algorithmic thresholds and configuration parameters for the AttnUnionDep engine.

ATTRIBOT			
Scope	**Parameter**	**Type / Default**	**Functional Description / Role**
Hierarchy	`keep_groups`	Integer (`4`)	Pruning threshold: Retains only the top 4 macro chunks after the first pass.
	`chunk_size`	Integer (`10`)	Sentences packed into a single macro paragraph node (per paper Appx D.2).
	`use_cache`	Boolean (`True`)	Enables KV-caching to bypass redundant prefix processing during LOO masking.
	`Len filter`	> 5 chars	Culls ultra-short strings to avoid executing zero-sum LOO operations.
Engine	`torch_dtype`	`torch.bfloat16`	Crucial for reducing memory costs during sequential loops over ablated inputs.
	`model_type`	Template Strings	Dynamically guides internal format placeholders (`{context}`, `{question}`).

Table 27: Hierarchical pruning thresholds and parameter setup for the AttriBoT evaluation engine.

CONTEXTCITE			
Scope	**Parameter**	**Type / Default**	**Functional Description / Role**
Regression	num_ablations	Integer (64)	Number of random masks executed to collect log-probability drops for Lasso.
	max_tokens	Integer (128000)	Rigid context window bound enforcing document slicing to avoid length crashes.
	provided_resp	answer String	Passes gold answer directly to surrogate model, bypassing generative steps.
Engine	autocast_fn	torch.bfloat16	Injected proxy function mapping Gemma 3 text configs to bfloat16 directly.
	expandable_seg	Boolean (True)	Protects the runtime block against VRAM inflation from stacked linear evaluations.
	Len filter	> 5 chars	Ignores mapped scores on micro-segments to filter scraper junk out of results.

Table 28: Surrogate model limits and linear regression parameters for ContextCite.

HSSAVG			
Scope	**Parameter**	**Type / Default**	**Functional Description / Role**
Extraction	layer_idx	Integer (16)	Target hidden layer for cosine representations where verbatim copy traces peak.
	theta (θ)	Float (0.5)	Proximity cutoff threshold used to classify natively extractive token spans.
	max_win_len	Integer (80)	Upper bound constraint (L) defining the maximum checked sliding window width.
	top_k_anchors	Integer (15)	Restricts computationally heavy sliding window similarity scans to top 15 anchors.
Engine	torch.float32	torch.float32	Forced precision upcasting prior to cosine math to stabilize matrix underflows.
	model_type	Template Strings	Dynamically selects prompt prefixes and instruction suffixes to map embeddings.

Table 29: Representational thresholds and sliding window search parameters for the HSSAVG method.

MIRAGE			
Scope	**Parameter**	**Type / Default**	**Functional Description / Role**
Saliency	`cti_thresh`	Float (`1.0`)	SD factor ($\mu + 1.0\sigma$) strictly identifying context-sensitive token variants.
	`cci_thresh`	Integer (`-5`)	Top-K parameter tracking specific source tokens that drive high-gradient emissions.
	`attributed_fn`	`"kl_divergence"`	Base metric evaluating prediction distribution drift with/without background info.
	`attr_method`	`"saliency"`	Sets the exact core Inseq engine strategy to analytical gradient backpropagation.
Engine	`model_type`	Template Strings	Passes architecture-specific chat templates and reserves exact `special_tokens`.
	`separator`	String (`" ^ "`)	Special delimiter enabling boolean token mask mapping to capture array shifts.
	`tokenizer_kw`	`{"use_fast": True}`	Enforces high-performance tokenizer engines for pixel-perfect offset records.
	`expandable_seg`	Boolean (`True`)	Critical baseline defense as backpropagation dramatically spikes peak VRAM.

Table 30: Saliency metrics and gradient threshold configurations for the MIRAGE engine.

TOKENSHAPLEY			
Scope	**Parameter**	**Type / Default**	**Functional Description / Role**
Shapley	`M`	Integer (`10`)	Maximum datastore neighbor capacity updated via Game Theory scoring (k-NN).
	`gamma` (γ)	Float (`1.0`)	Distance scalar governing RBF kernel density projections across high-D tokens.
	`alphas` (α)	$\frac{1}{N}\sum_{i=j}^{N}\frac{1}{i}$	Fractional distribution array calculating marginal utility changes at rank index i.
	`mid_layer_idx`	`num_layers // 2`	Intercepts layers precisely at the mid-point axis to capture balanced semantic vectors.
Engine	`torch_dtype`	`torch.bfloat16`	Maintains VRAM consistency during continuous prefix datastore build operations.
	`model_type`	Template Strings	Dynamically directs sequence construction inside the consolidated prompt assembly.
	`score calc`	`np.sum()` Array	Segment boundaries collecting individual token Shapley values across active sentences.

Table 31: k-NN datastore properties and Game Theory distribution parameters for TokenShapley.