Replacement-Aware Source Valuation for Retrieval-Augmented Systems

Anonymous ACL submission

Abstract

Retrieval-augmented question answering and evidence retrieval systems depend on external evidence sources, but retrieval frequency, retrieval score, and fixed-context deletion do not capture how source unavailability changes utility after reretrieval. When a source is removed from the corpus, the retriever may select replacement evidence, which can recover or alter downstream utility. We formulate replacement-aware source valuation as the leave-one-source-out change in aggregate task utility under reretrieval. Exact reretrieval LOO provides a direct validation target, but is too expensive to run for every source. We propose R-ReLOO, a cached candidate-pool estimator that simulates source removal by allowing lower-ranked candidates to enter the retrieved context as replacement evidence. R-ReLOO occupies the middle ground between exact reretrieval LOO, which is faithful but costly, and fixed top-k deletion, which is efficient but replacement-blind. Across three evidence-oriented NLP datasets, R-ReLOO consistently improves ranking fidelity to exact reretrieval LOO over fixed top-k-only deletion, with larger gains when replacement evidence plays a stronger role, and outperforms retrieval/source baselines. Selection and pruning experiments show useful source-management signals, while top-k-only deletion remains a competitive operational baseline. R-ReLOO is a utility-relative estimator for evaluated retrieval/evidence workloads, not a production RAG quality measure, data-pricing method, or Shapley-style allocation rule.

1 Introduction

Retrieval-augmented NLP systems answer questions, verify claims, and support knowledge-intensive prediction by retrieving evidence from an external corpus before making downstream decisions (Lewis et al., 2020; Guu et al., 2020; Petroni et al., 2021). This paradigm underlies multi-hop question answering (Yang et al., 2018; Ho et al., 2020), scientific claim verification (Wadden et al., 2020), and citation-oriented generation (Gao et al., 2023). In these systems, performance depends not only on the retriever or reader, but also on which evidence sources are available to retrieve. This makes source utility a practical question: for a fixed workload, retriever, and utility function, which sources would reduce downstream utility if they became unavailable?

Answering this question is important for source auditing, evidence retention, corpus pruning, and retrieval-infrastructure maintenance. A system designer may want to identify sources that are critical for a specific workload, remove redundant or noisy sources under storage and latency constraints, or understand which parts of an evidence corpus support task utility. These goals are related to data valuation (Ghorbani and Zou, 2019; Wang and Jia, 2023; Yoon et al., 2020; Sim et al., 2022) and resource selection in information retrieval (Wu and Crestani, 2003). Yet retrieval-augmented source valuation has a distinctive complication: the retrieved context is not fixed. If a source is removed from the corpus, the retriever can rerank the remaining sources and select replacement evidence.

Common retrieval-side signals do not measure this replacement-aware contribution. Retrieval frequency measures how often a source is exposed to the downstream system, and average retrieval score measures the retriever's preference for that source. These are useful descriptive statistics, but they are not interventional utility estimates. A frequently retrieved source may be redundant if other sources provide the same support, while an infrequently retrieved source may be valuable if it uniquely supports a difficult subset of queries. Similarly, a high retrieval score does not necessarily imply high marginal utility for the workload.

A closer baseline is fixed-context deletion: retrieve the top-k context once, remove a source from that fixed context, and measure the utility

drop. This baseline is efficient and often strong because many useful sources are already present in the retrieved context. However, it does not model what the retrieval system would do after corpus removal. When a source is unavailable, retrieval becomes a new post-intervention problem: a lower-ranked source may enter the final context as replacement evidence. This replacement can recover lost support, degrade the retrieved context, or even improve measured utility by promoting better evidence. Thus, source value under corpus removal should be defined with respect to reretrieval, not only deletion from an already selected top-k list.

We study this issue through leave-one-source-out (LOO) source valuation for retrieval-augmented question answering and evidence retrieval. For a source s_i, exact reretrieval LOO removes s_i from the corpus, reruns retrieval for the workload, and measures the resulting change in aggregate utility. This provides a faithful utility-relative validation target for source value under corpus removal. However, exact reretrieval LOO is expensive: scoring every source can require repeated retrieval and utility evaluation over a large workload and source collection. This cost makes exact reretrieval LOO unsuitable as a routine source-ranking, source-selection, or pruning tool.

We propose R-ReLOO, a replacement-aware approximate LOO estimator for retrieval-augmented source valuation. R-ReLOO retrieves, for each query, a candidate pool larger than the final top-k context. It then simulates source removal inside this cached pool, allowing lower-ranked candidates to enter the context as replacement evidence. This yields a computationally lightweight middle ground between exact reretrieval LOO, which is faithful but costly, and fixed top-k-only deletion, which is efficient but replacement-blind. R-ReLOO is non-parametric and training-free: it does not require learning a new model, estimating full Shapley-style coalition values, or rerunning retrieval once per source.

We evaluate R-ReLOO on three evidence-oriented NLP datasets: HotpotQA (Yang et al., 2018), SciFact (Wadden et al., 2020), and 2WikiMultiHopQA (Ho et al., 2020). We compare against fixed top-k deletion, retrieval/source proxies, and bounded sampled Shapley/Banzhaf-style valuation baselines. Across repeated bounded exact-validation subsets, R-ReLOO improves Spearman correlation with exact reretrieval LOO over top-k-only deletion on all three datasets. The gains are smaller on HotpotQA and larger on SciFact and 2WikiMultiHopQA, indicating that the importance of replacement modeling depends on the workload and retrieval setting. Source selection and pruning results are more nuanced: R-ReLOO provides useful decision support, while top-k-only deletion remains a strong operational baseline and can match or exceed it on some decision metrics.

Contributions. Our contributions are summarized as follows.

- We formulate evidence-source utility as a replacement-aware reretrieval LOO contrast, distinguishing corpus removal from deletion within a fixed retrieved context.
- We introduce R-ReLOO, a cached candidate-pool estimator that approximates post-removal reretrieval without source-by-source retrieval over the full corpus.
- We evaluate R-ReLOO against exact reretrieval LOO, fixed top-k-only deletion, retrieval/source proxies, and bounded sampled Shapley/Banzhaf-style baselines on three evidence-oriented NLP datasets.
- We analyze ranking fidelity, selection/pruning behavior, candidate-pool sensitivity, and efficiency while keeping the claim bounded to utility-relative source valuation.

2 Background and Related Work

Retrieval-augmented NLP and evidence grounding. Retrieval-augmented generation and retrieval-integrated language models use external documents, passages, or evidence units to support downstream prediction (Lewis et al., 2020; Guu et al., 2020). Benchmarks such as KILT, HotpotQA, SciFact, and 2WikiMultiHopQA connect task performance with evidence retrieval, provenance, or supporting facts (Petroni et al., 2021; Yang et al., 2018; Wadden et al., 2020; Ho et al., 2020). Citation-oriented generation and RAG attribution further study whether generated statements or predictions are grounded in retrieved or cited evidence (Gao et al., 2023; Qi et al., 2024; Nematov et al., 2025). Our question is complementary: rather than attributing an individual output to evidence, we estimate workload-level source value under source removal and reretrieval.

Resource selection and corpus management. Resource selection in distributed and federated retrieval treats collections or resources as selectable

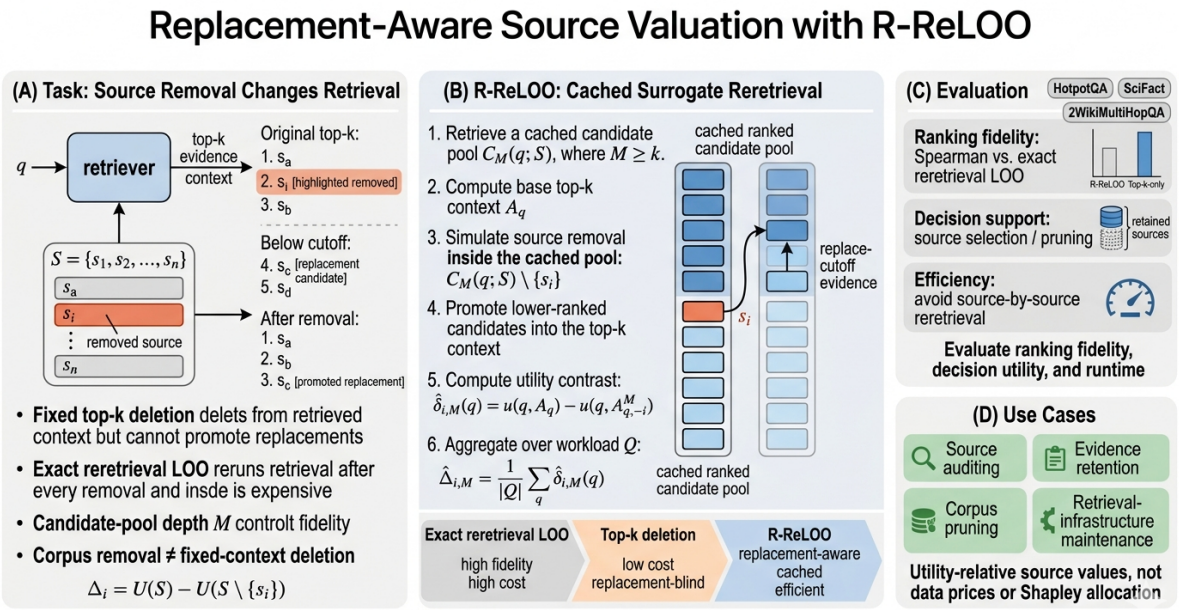

Figure 1: Overview of R-ReLOO. Exact reretrieval LOO removes a source and reretrieves, but is expensive. Fixed top-k-only deletion is efficient but replacement-blind. R-ReLOO caches a larger candidate pool and simulates source removal within it, allowing lower-ranked evidence to enter as replacement and producing utility-relative source values.

units under relevance, coverage, diversity, cost, and efficiency constraints (Wu and Crestani, 2003; Si and Callan, 2003; Shokouhi, 2007; Hong and Si, 2013). This provides a natural background for source ranking, retention, and pruning in retrieval-augmented systems. However, collection-level relevance, retrieval frequency, retrieval score, and source diversity do not directly measure the utility change caused by removing an individual source and reretrieving from the remaining corpus. Selection frequency and marginal utility can diverge when replacement evidence remains available. R-ReLOO targets this source-level removal-and-replacement contrast.

Data valuation and leave-one-out utility. Data valuation methods estimate how data units contribute to downstream utility. Leave-one-out gives a direct removal contrast, while Data Shapley, Data Banzhaf, and DVRL estimate coalition-based or learned data contribution (Ghorbani and Zou, 2019; Wang and Jia, 2023; Yoon et al., 2020; Sim et al., 2022). We include bounded sampled Shapley and Banzhaf variants as related baselines, but our target is narrower and retrieval-specific: R-ReLOO estimates a leave-one-source-out utility contrast for a fixed workload, retriever, source granularity, retrieval depth, candidate-pool depth, and utility function. It does not average marginal contributions over all coalitions and does not provide pricing, fairness, or allocation guarantees.

Positioning. R-ReLOO is positioned between fixed-context deletion and exact reretrieval LOO. Fixed-context deletion is efficient but replacement-blind, because it removes a source only from the already retrieved top-k context. Exact reretrieval LOO captures source removal and replacement by rerunning retrieval, but is expensive at scale. R-ReLOO uses cached candidate pools to approximate this replacement behavior without source-by-source reretrieval over the full corpus.

3 Problem Formulation

We study source valuation for retrieval-augmented QA and evidence-retrieval workloads. Let $S = \{s_1, \ldots, s_n\}$ be a collection of retrievable evidence sources and $Q = \{q_1, \ldots, q_m\}$ be a fixed workload. For a query q, a retriever $R_k(q; S)$ returns an ordered top-k context from S. A bounded utility function $u(q, A)$ scores a retrieved context A, for example by evidence recall, answer-string support, claim-support overlap, or a weighted combination. The aggregate workload utility is

$$U(S) = \frac{1}{|Q|} \sum_{q \in Q} u(q, R_k(q; S)). \qquad (1)$$

Source value is not intrinsic to a source. It is relative to the workload, retriever, utility function, retrieval depth, and source granularity.

Exact reretrieval LOO. For source s_i, the exact reretrieval leave-one-source-out value is

$$\Delta_i = U(S) - U(S \setminus \{s_i\}). \tag{2}$$

Equivalently, at the query level,

$$\delta_i^{\text{exact}}(q) = u(q, R_k(q; S)) - u(q, R_k(q; S \setminus \{s_i\})), \tag{3}$$

with $\Delta_i = |Q|^{-1} \sum_{q \in Q} \delta_i^{\text{exact}}(q)$. This contrast corresponds to the operational intervention of making s_i unavailable and rerunning retrieval over the remaining corpus. It captures replacement evidence because the post-removal context is selected by the retriever rather than obtained by editing the original top-k context.

Fixed-context deletion and replacement. A cheaper baseline retrieves $A_q = R_k(q; S)$ once and removes s_i only from this fixed context:

$$\delta_i^{\text{top}k}(q) = u(q, A_q) - u(q, A_q \setminus \{s_i\}). \tag{4}$$

This measures sensitivity of the already selected context, but cannot promote replacement evidence from below the retrieval cutoff.

To make the missing term explicit, define

$$\begin{aligned} A_{q,-i}^{\text{del}} &= A_q \setminus \{s_i\}, \\ A_{q,-i}^{\text{rep}} &= R_k(q; S \setminus \{s_i\}). \end{aligned} \tag{5}$$

Then

$$\delta_i^{\text{exact}}(q) = \delta_i^{\text{top}k}(q) + \rho_i(q), \tag{6}$$

where

$$\rho_i(q) = u(q, A_{q,-i}^{\text{del}}) - u(q, A_{q,-i}^{\text{rep}}) \tag{7}$$

is the replacement effect. If reretrieval finds evidence that recovers the removed source's contribution, $\rho_i(q)$ can be negative and fixed-context deletion overestimates source value. If the replacement context is worse or distracting, $\rho_i(q)$ can be positive. This replacement effect is the central reason source valuation under corpus removal differs from deletion from a fixed evidence list.

Evaluation views. We evaluate source values through ranking fidelity to exact reretrieval LOO, source selection under a keep budget, and pruning of low-valued sources followed by reretrieval. These decision tasks use source values as signals; they are not assumed to solve a globally optimal subset-selection problem. The formulation is deliberately narrower than general data valuation: it estimates one specified LOO utility contrast for a fixed retrieval-augmented workload, not a Shapley, Banzhaf, pricing, fairness, or causal allocation framework.

4 Method

We introduce R-ReLOO, a replacement-aware approximate leave-one-source-out estimator for retrieval-augmented evidence-source valuation. Exact reretrieval LOO measures the utility change caused by making a source unavailable and reretrieving from the remaining corpus, but it requires a post-removal retrieval pass for each source removal. R-ReLOO replaces this expensive post-removal retriever with a cached candidate-pool surrogate.

The method has three technical ingredients. First, it defines a surrogate reretrieval operator that allows below-cutoff candidates to enter the retrieved context after source removal. Second, it estimates the replacement effect that fixed top-k deletion omits. Third, it uses candidate-pool depth M as an explicit fidelity-cost control.

4.1 Cached Surrogate Reretrieval

For each query q, R-ReLOO retrieves an ordered candidate pool $C_M(q; S)$ with $M \geq k$. It approximates the exact post-removal retriever $R_k(q; S \setminus \{s_i\})$ by

$$\widetilde{R}_{k,M}(q; S \setminus \{s_i\}) = \text{top}_k(C_M(q; S) \setminus \{s_i\}). \tag{8}$$

This surrogate is the central object in R-ReLOO. It is not merely a longer retrieval list: it defines the post-removal context used in place of exact reretrieval. When $M = k$, the surrogate has no below-cutoff replacement capacity and reduces to fixed-context deletion. When $M > k$, lower-ranked sources can enter the final context after a source is removed.

The query-level R-ReLOO estimate is

$$\begin{aligned} \hat{\delta}_{i,M}(q) = u(q, R_k(q; S)) \\ - u(q, \widetilde{R}_{k,M}(q; S \setminus \{s_i\})). \end{aligned} \tag{9}$$

The workload-level estimate is

$$\hat{\Delta}_{i,M} = \frac{1}{|Q|} \sum_{q \in Q} \hat{\delta}_{i,M}(q). \tag{10}$$

If $s_i \notin C_M(q; S)$, R-ReLOO assigns zero query-level contribution under depth M, because the cached pool contains no observable path through which s_i can affect the query. The values are signed: negative values indicate that removing a source improves the evaluated context under the chosen utility.

Table 1: Comparison of exact reretrieval leave-one-source-out, top-k-only deletion, and R-ReLOO.

Method	Replacement?	Retrieval cost	Target
Exact reretrieval LOO	Yes	Reretrieve after each source removal	Corpus-removal utility contrast
Top-k-only deletion	No	Retrieve once, delete from fixed context	Fixed-context deletion sensitivity
R-ReLOO	Approximate	Retrieve candidate pools once, simulate replacement	Candidate-pool approximation to reretrieval LOO

Algorithm 1 R-ReLOO: Replacement-Aware Approximate LOO Valuation

Input: Source corpus $S = \{s_i\}_{i=1}^n$, workload $Q = \{q_j\}_{j=1}^m$, retriever R, retrieval depth k, candidate-pool depth $M \geq k$, utility function u.

Output: Source-value estimates $\{\hat{\Delta}_{i,M}\}_{i=1}^n$.

Initialize $\hat{\Delta}_{i,M} \leftarrow 0$ for all $s_i \in S$.
for each query $q \in Q$ **do**
  Retrieve candidate pool $C_M(q; S) \leftarrow R_M(q; S)$.
  Set base context $A_q \leftarrow \mathrm{top}_k(C_M(q; S))$.
  Compute base utility $b_q \leftarrow u(q, A_q)$.
  for each source $s_i \in C_M(q; S)$ **do**
    Construct replacement-aware context $A_{q,-i}^M \leftarrow \mathrm{top}_k(C_M(q; S) \setminus \{s_i\})$.
    Compute query contribution $\hat{\delta}_{i,M}(q) \leftarrow b_q - u(q, A_{q,-i}^M)$.
    Update $\hat{\Delta}_{i,M} \leftarrow \hat{\Delta}_{i,M} + \hat{\delta}_{i,M}(q)/|Q|$.
  end for
end for
return $\{\hat{\Delta}_{i,M}\}_{i=1}^n$.

4.2 R-ReLOO Algorithm

Algorithm 1 summarizes the estimator. R-ReLOO retrieves one candidate pool per query, computes the base utility, and simulates each in-pool source removal by promoting lower-ranked candidates when needed.

4.3 Replacement-Effect Approximation

R-ReLOO can be interpreted as estimating the replacement term omitted by fixed-context deletion. Recall that

$$\delta_i^{\mathrm{exact}}(q) = \delta_i^{\mathrm{top}k}(q) + \rho_i(q). \tag{11}$$

R-ReLOO replaces the exact post-removal context with

$$A_{q,-i}^M = \mathrm{top}_k(C_M(q; S) \setminus \{s_i\}), \tag{12}$$

and uses

$$\hat{\rho}_{i,M}(q) = u(q, A_{q,-i}^{\mathrm{del}}) - u(q, A_{q,-i}^M) \tag{13}$$

as a cached-pool approximation to the exact replacement effect. Thus,

$$\hat{\delta}_{i,M}(q) = \delta_i^{\mathrm{top}k}(q) + \hat{\rho}_{i,M}(q). \tag{14}$$

This decomposition clarifies what R-ReLOO adds over fixed top-k deletion: it estimates how utility changes when replacement evidence is allowed to enter from the cached pool. If replacement evidence recovers the deleted source's support, R-ReLOO reduces the estimated value relative to fixed-context deletion; if replacement evidence is worse, R-ReLOO increases it.

4.4 Candidate-Pool Adequacy

The candidate-pool depth M controls the fidelity-cost tradeoff. R-ReLOO is most accurate when the cached pool contains the sources that exact reretrieval would select after source removal and when the retriever ranking is locally stable. For a fixed query-source pair (q, s_i), let

$$\begin{aligned} B_{q,-i} &= R_k(q; S \setminus \{s_i\}), \\ A_{q,-i}^M &= \mathrm{top}_k(C_M(q; S) \setminus \{s_i\}). \end{aligned} \tag{15}$$

If all sources in $B_{q,-i}$ are contained in $C_M(q; S)$, the relative ordering of relevant remaining candidates is stable under removal, and u is deterministic, then

$$A_{q,-i}^M = B_{q,-i} \quad \text{and} \quad \hat{\delta}_{i,M}(q) = \delta_i^{\mathrm{exact}}(q). \tag{16}$$

This is a sufficient condition for query-level exactness, not a global guarantee. Additional proof details, replacement-coverage diagnostics, and utility-sensitive error analysis are provided in Appendix A.

4.5 Decision Use, Complexity, and Boundaries

R-ReLOO values can rank sources, retain top-valued sources under a budget, or prune low-valued sources before reretrieval. These uses are decision support, not optimal subset selection: one-at-a-time LOO values do not model all interactions among

Table 2: Dataset-scale inputs used for source valuation. Task abbreviations: MH-QA denotes multi-hop question answering, and SCV denotes scientific claim verification. 2Wiki denotes 2WikiMultiHopQA. Queries are examples with gold evidence/source annotations. Sources are retrievable evidence units in the local corpus. Retrieval records are cached top-M retriever outputs used by the valuation pipeline.

Dataset	Task	Queries	Sources	Retr. rec.
HotpotQA	MH-QA	2,000	19,305	148,100
SciFact	SCV	1,109	5,183	22,166
2Wiki	MH-QA	500	3,439	10,000

retained or removed sources. We therefore evaluate both ranking fidelity against exact LOO and reretrieval-based selection/pruning utility.

Let $n = |S|$, $m = |Q|$, C_R be one retrieval cost, and C_u be one utility-evaluation cost. Exact reretrieval LOO requires $O(mnC_R)$ retrieval work plus $O(mnC_u)$ utility evaluations. Top-k-only deletion requires $O(mC_R)$ retrieval and up to $O(mkC_u)$ utility evaluations. R-ReLOO retrieves once per query at candidate depth M, requiring $O(mC_R)$ retrieval work and up to $O(mMC_u)$ in-pool utility evaluations. Thus, R-ReLOO occupies the fidelity-cost region between fixed-context deletion and exact reretrieval LOO, with cost controlled by M rather than by n.

R-ReLOO estimates the full-corpus removal contrast $F(S) - F(S \setminus \{s_i\})$, whereas Shapley and Banzhaf values average marginal contributions over coalitions. We include bounded sampled Shapley and Banzhaf as related baselines, but do not claim that R-ReLOO satisfies coalition-value axioms or provides pricing, fairness, or allocation guarantees. Extended complexity and coalition-baseline details are in Appendix A.9.

5 Experiments

We evaluate whether R-ReLOO efficiently approximates exact reretrieval leave-one-source-out (LOO) source valuation. We ask five questions: **RQ1** tests ranking fidelity to exact reretrieval LOO; **RQ2** compares against bounded sampled Shapley and Banzhaf baselines; **RQ3** analyzes candidate-pool sensitivity; **RQ4** evaluates source selection and pruning; and **RQ5** measures runtime efficiency.

5.1 Setup

We evaluate on three evidence-oriented NLP datasets: HotpotQA and 2WikiMultiHopQA for multi-hop QA (Yang et al., 2018; Ho et al., 2020), and SciFact for scientific evidence retrieval (Wadden et al., 2020). Table 2 summarizes the evaluated query and source scales. The default retriever in the main experiments is a BM25 lexical retriever (Kamphuis et al., 2020) over source titles and texts. Appendix F reports bounded diagnostics with dense and hybrid retrievers. The utility $u(q, A)$ is a bounded evidence-oriented task proxy, combining dataset-specific evidence recall, answer support, claim overlap, or retrieval precision signals; it is not a retrieval score. All experiments are conducted on a single server with four NVIDIA A40 GPUs and an Intel Xeon Gold 6248R CPU. [1]

We compare R-ReLOO with retrieval/source proxies, fixed top-k-only deletion, and bounded sampled Shapley- and Banzhaf-style valuation baselines. Exact reretrieval LOO is used as the validation target on repeated bounded validation subsets. We report Spearman ranking fidelity to exact LOO over five seeded repeats, reretrieval-based selection and pruning utility, candidate-pool sensitivity, and measured runtime. Appendix C gives additional protocol details, Appendix D gives baseline definitions, and Appendix F reports retriever-type and replacement-coverage diagnostics.

Table 3: Ranking fidelity to exact reretrieval LOO, measured by Spearman correlation (↑) over five seeded bounded exact-validation repeats. Bold indicates the best method and underline indicates the second-best method in each row. Δ reports R-ReLOO minus top-k-only deletion.

Dataset	R-ReLOO	Top-k	Best simple	Δ vs. Top-k
HotpotQA	**0.8527**	<u>0.8325</u>	0.5036	+0.0202
SciFact	**0.7576**	<u>0.6422</u>	0.4309	+0.1154
2WikiMultiHopQA	**0.9262**	<u>0.8138</u>	0.5801	+0.1124

5.2 RQ1: Ranking Fidelity to Exact Reretrieval LOO

Table 3 and Figure 2 report ranking fidelity to exact reretrieval LOO. R-ReLOO achieves the highest Spearman correlation on all three datasets. Compared with top-k-only deletion, it improves Spearman correlation by $+0.0202$ on HotpotQA, $+0.1154$ on SciFact, and $+0.1124$ on 2WikiMultiHopQA. The smaller gain on HotpotQA suggests that many source effects are already visible in the original top-k context, while the larger gains on SciFact and 2WikiMultiHopQA indicate stronger replacement effects.

[1] Code and data are available in the anonymous 4open repository.

Table 4: Paired Spearman differences over five seeded bounded exact-validation repeats. Positive values indicate that R-ReLOO better matches exact reretrieval LOO. Intervals summarize repeated-validation uncertainty.

Dataset	Comparison	Mean $\Delta \uparrow$	Std. dev.	95% CI
HotpotQA	R-ReLOO – Top-k	+0.0202	0.0124	[+0.0096, +0.0311]
HotpotQA	R-ReLOO – best simple	+0.3490	0.0386	[+0.3150, +0.3831]
SciFact	R-ReLOO – Top-k	+0.1154	0.0521	[+0.0747, +0.1621]
SciFact	R-ReLOO – best simple	+0.3267	0.0323	[+0.2987, +0.3546]
2WikiMultiHopQA	R-ReLOO – Top-k	+0.1124	0.0203	[+0.0931, +0.1298]
2WikiMultiHopQA	R-ReLOO – best simple	+0.3462	0.0122	[+0.3347, +0.3563]

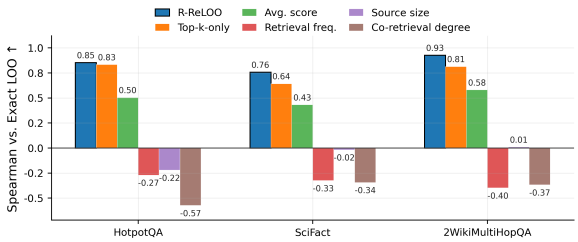

Figure 2: Ranking fidelity against exact reretrieval LOO. Higher is better. R-ReLOO better matches exact reretrieval LOO than fixed top-k-only deletion and retrieval/source proxies in the repeated validation setting.

Table 5: Bounded induced-pool ranking fidelity to exact reretrieval LOO, measured by Spearman correlation (↑). Sampled Shapley uses 30 sampled permutations, and sampled Banzhaf uses 30 sampled coalitions per source with cached retrieval-coalition utility. Bold indicates the best method and underline indicates the second-best method in each row.

Dataset	R-ReLOO	Top-k	Samp. Banzhaf	Samp. Shapley
HotpotQA	**0.8292**	<u>0.8056</u>	0.7943	0.7524
SciFact	**0.7205**	0.4182	<u>0.5909</u>	0.5783
2WikiMultiHopQA	**0.8817**	<u>0.8188</u>	0.7450	0.7372

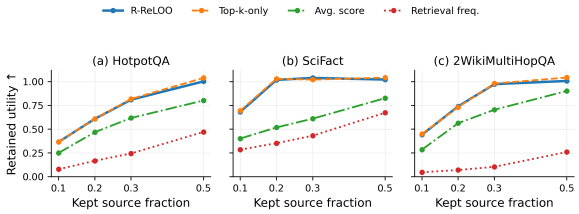

Figure 3: Reretrieval-based retained utility under source selection and pruning budgets. R-ReLOO provides useful decision support, while top-k-only remains a strong operational baseline.

Table 4 shows that the improvements over top-k-only are consistent across all five seeded repeats. R-ReLOO also substantially improves over the best simple proxy, confirming that source value is not captured by retrieval exposure or retriever preference alone. A visualization of the paired differences is provided in Appendix E.

5.3 RQ2: Coalition-Style Valuation Baselines

Table 5 compares R-ReLOO with bounded sampled Shapley and Banzhaf baselines. These coalition-style methods are informative utility-based baselines, but R-ReLOO achieves the best mean Spearman correlation. This supports our distinction between coalition valuation and reretrieval LOO: Shapley/Banzhaf-style methods estimate marginal contribution over sampled source subsets, while R-ReLOO targets full-corpus source removal followed by reretrieval.

5.4 RQ3: Candidate-Pool Sensitivity

R-ReLOO depends on candidate-pool depth M. Larger M exposes more possible replacement evidence, but also increases utility-evaluation cost and may introduce lower-ranked candidates with noisier utility effects. In a bounded sensitivity probe over $M \in \{10, 20, 50\}$, we observe dataset-dependent behavior: deeper pools improve fidelity on 2WikiMultiHopQA, while HotpotQA and SciFact do not improve monotonically. This supports treating M as an explicit fidelity-cost control rather than a hidden implementation choice. Full sensitivity results are provided in Appendix A.5.

5.5 RQ4: Source Selection and Pruning

Table 6 reports reretrieval-based source selection at a 20% keep budget. R-ReLOO provides useful decision support but does not uniformly dominate top-k-only deletion: it matches top-k-only on HotpotQA, trails it on SciFact, and exceeds it on 2WikiMultiHopQA. This shows that better single-source ranking fidelity does not automatically imply dominance in joint keep/prune decisions.

5.6 RQ5: Efficiency

Table 7 and Figure 4 show that R-ReLOO is substantially faster than exact reretrieval LOO because it avoids source-by-source reretrieval. Top-k-only is cheaper, as expected, but its lower cost comes

Table 6: Reretrieval-based source selection at a 20% keep budget. RU denotes retained utility and ER denotes evidence recall; higher is better for both. Bold indicates the best method and underline indicates the second-best method for each dataset and metric; ties are marked in bold.

Dataset	Method	RU ↑	ER ↑
HotpotQA	R-ReLOO	**0.6092**	**0.4350**
	Top-k	**0.6092**	**0.4350**
	Avg. score	0.4685	0.3500
	Retr. freq.	0.1676	0.1050
SciFact	R-ReLOO	1.0188	0.7611
	Top-k	**1.0303**	**0.7699**
	Avg. score	0.5182	0.3363
	Retr. freq.	0.3526	0.1858
2WikiMultiHopQA	R-ReLOO	**0.7397**	**0.4298**
	Top-k	0.7318	**0.4298**
	Avg. score	0.5630	0.3719
	Retr. freq.	0.0702	0.0537

Table 7: Runtime in bounded exact-validation, averaged over five repeats. Exact, R-ReLOO, and Top-k denote runtime for exact reretrieval LOO, R-ReLOO, and top-k-only deletion, respectively. Spd. denotes Exact/R-ReLOO speedup. Lower runtime is better; higher speedup is better. Bold indicates the fastest method and underline indicates the second-fastest method in each row.

Dataset	Exact ↓	R-ReLOO ↓	Top-k ↓	Spd. ↑
HotpotQA	3.4307s	0.0160s	**0.0094s**	214.4×
SciFact	9.1214s	0.1585s	**0.0632s**	57.5×
2WikiMultiHopQA	4.5771s	0.0290s	**0.0119s**	157.6×

from ignoring replacement evidence. Thus, R-ReLOO is most useful when replacement awareness is needed but source-by-source exact reretrieval is too costly.

5.7 Diagnostic Takeaways

The results suggest three patterns. First, replacement modeling explains R-ReLOO's ranking-fidelity gain: fixed-context deletion can misestimate source value when corpus removal promotes replacement evidence. Second, exposure is not marginal utility: frequent or high-scoring sources can be redundant, while rare sources can be useful for difficult queries. Third, R-ReLOO's main failure mode is candidate-pool misspecification, where useful replacement evidence lies outside the cached pool or retrieval changes non-locally after removal. These patterns explain why R-ReLOO is strongest as an approximation to exact single-source reretrieval LOO, while top-k-only remains competitive in some joint selection and pruning decisions. Additional qualitative cases are provided in Appendix G.

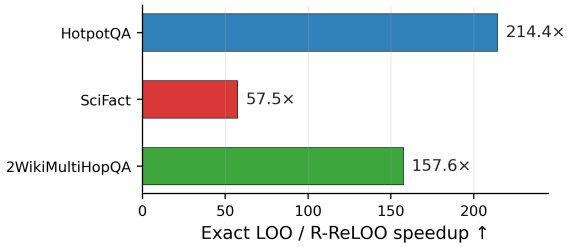

Figure 4: Measured runtime speedup of R-ReLOO relative to exact reretrieval LOO in bounded validation.

6 Limitations

R-ReLOO is a bounded approximation to reretrieval LOO, and its accuracy depends on candidate-pool coverage, retriever stability, and the chosen utility function. Our experiments are limited to three evidence-oriented NLP datasets under fixed source units, retrieval depth, and candidate-pool depth. Appendix diagnostics include bounded sparse, dense, and hybrid retriever probes with exact post-removal reretrieval targets, but neural rerankers, LLM readers, stochastic generation, generated-answer quality, and production RAG remain future work. The resulting values are utility-relative decision-support signals, not intrinsic source worth, data prices, Shapley/Banzhaf values, fairness allocations, causal guarantees, or globally optimal source-selection rules. Additional limitations are discussed in Appendix H. A potential risk is pruning sources without considering coverage or provenance. R-ReLOO should be used as decision support, not an automatic removal rule.

7 Conclusion

We presented R-ReLOO, a replacement-aware approximate LOO estimator for source valuation in retrieval-augmented QA and evidence retrieval. R-ReLOO uses cached candidate pools to approximate post-removal reretrieval, capturing replacement evidence that fixed top-k-only deletion misses while avoiding source-by-source exact reretrieval. Across three evidence-oriented NLP datasets, R-ReLOO improves ranking fidelity to exact reretrieval LOO over fixed-context deletion and retrieval/source proxies, and provides useful source-selection and pruning signals. The method is deliberately scoped to utility-relative retrieval/evidence workloads rather than universal data valuation, pricing, fairness, or production RAG quality.

References

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. 2023. Enabling large language models to generate text with citations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6465–6488, Singapore. Association for Computational Linguistics.

Amirata Ghorbani and James Zou. 2019. Data shapley: Equitable valuation of data for machine learning. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2242–2251. PMLR.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. REALM: Retrieval-augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 3929–3938. PMLR.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020. Constructing a multi-hop QA dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6609–6625, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Dzung Hong and Luo Si. 2013. Search result diversification in resource selection for federated search. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 613–622. ACM.

Chris Kamphuis, Arjen P De Vries, Leonid Boytsov, and Jimmy Lin. 2020. Which bm25 do you mean? a large-scale reproducibility study of scoring variants. In *European Conference on Information Retrieval*, pages 28–34. Springer.

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

Ikhtiyor Nematov, Tarik Kalai, Elizaveta Kuzmenko, Gabriele Fugagnoli, Dimitris Sacharidis, Katja Hose, and Tomer Sagi. 2025. Source attribution in retrieval-augmented generation. *arXiv preprint arXiv:2507.04480*.

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

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.

Milad Shokouhi. 2007. Central-rank-based collection selection in uncooperative distributed information retrieval. In *Advances in Information Retrieval: 29th European Conference on IR Research, ECIR 2007*, volume 4425 of *Lecture Notes in Computer Science*, pages 160–172. Springer.

Luo Si and James P. Callan. 2003. Relevant document distribution estimation method for resource selection. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 298–305. ACM.

Rachael Hwee Ling Sim, Xinyi Xu, and Bryan Kian Hsiang Low. 2022. Data valuation in machine learning: "ingredients", strategies, and open challenges. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*, pages 5607–5614.

David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. 2020. Fact or fiction: Verifying scientific claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7534–7550, Online. Association for Computational Linguistics.

Jiachen T. Wang and Ruoxi Jia. 2023. Data banzhaf: A robust data valuation framework for machine learning. In *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, volume 206 of *Proceedings of Machine Learning Research*, pages 6388–6421. PMLR.

Shengli Wu and Fabio Crestani. 2003. Distributed information retrieval: A multi-objective resource selection approach. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 11(1):83–99.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium. Association for Computational Linguistics.

Jinsung Yoon, Sercan O. Arik, and Tomas Pfister. 2020. Data valuation using reinforcement learning. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 10842–10851. PMLR.

A Extended Method Details

This appendix provides additional details for R-ReLOO, including source granularity, notation, implementation choices, exactness conditions, approximation diagnostics, decision limits, and complexity. It does not introduce empirical claims beyond the bounded validation reported in the main text.

A.1 Source Granularity and Scope

R-ReLOO values are defined for a chosen source granularity. In a retrieval-augmented system, a source may be a passage, document, evidence block, webpage, knowledge-base entry, provider-level bundle, or versioned corpus snapshot. The same leave-one-source-out contrast can be applied to any of these units, but the interpretation changes with the unit of removal. A passage-level value measures the utility of making that passage available to retrieval; a document-level value aggregates over all evidence contained in the document; a provider-level value measures the contribution of an entire source bundle.

This distinction is important because source values are not intrinsic properties of text. They are properties of a source–workload–system tuple: the selected workload Q, retriever R, utility function u, retrieval depth k, candidate-pool depth M, and source granularity jointly determine the measured value. Changing any of these choices can change the resulting ranking. R-ReLOO should therefore be read as estimating a specified operational utility contrast, not as assigning universal data worth.

A.2 Notation Summary

Table 8: Notation used by the R-ReLOO formulation.

Symbol	Meaning
$S = \{s_1, \ldots, s_n\}$	retrievable source corpus
$Q = \{q_1, \ldots, q_m\}$	fixed query or claim workload
$R_k(q; S)$	top-k retrieval over S
$C_M(q; S)$	cached candidate pool, $M \geq k$
A_q	base retrieved top-k context
$A^{\text{del}}_{q,-i}$	fixed-context deletion context
$A^{\text{rep}}_{q,-i}$	exact post-removal context
$A^{M}_{q,-i}$	cached-pool post-removal context
$u(q, A)$	context utility
$U(S)$	aggregate workload utility
Δ_i	exact reretrieval LOO value of s_i
$\hat{\Delta}_{i,M}$	R-ReLOO estimate with pool depth M
$\rho_i(q)$	exact replacement effect
$\hat{\rho}_{i,M}(q)$	cached-pool replacement-effect estimate
$\gamma_M(q, i)$	replacement coverage diagnostic

A.3 Implementation Details

Algorithm 1 gives the main estimator. In implementation, retrieval is performed once per query at candidate depth M. The leave-one-source-out updates are then computed by filtering the cached candidate list and recomputing the utility on the resulting top-k context. This is the main computational difference from exact reretrieval LOO, which reruns retrieval after each source removal.

Only sources that appear in $C_M(q; S)$ receive a nonzero update for query q. This is a deliberate approximation: sources outside the candidate pool are treated as unobserved for that query under depth M. Increasing M expands the set of observable replacement sources, but also increases cached retrieval size and utility-evaluation cost.

The estimator returns signed source values. Positive values indicate that source removal decreases utility; negative values indicate that source removal improves the evaluated context. Negative values can occur when the original source is distracting, redundant, or blocks more useful evidence from entering the top-k context.

A.4 Candidate-Pool Exactness Proof Sketch

For a query q and source s_i, let

$$B_{q,-i} = R_k(q; S \setminus \{s_i\}) \tag{17}$$

be the exact reretrieved context after removing s_i, and let

$$A^{M}_{q,-i} = \text{top}_k(C_M(q; S) \setminus \{s_i\}) \tag{18}$$

be the context produced by R-ReLOO from the cached candidate pool.

R-ReLOO recovers the exact query-level reretrieval LOO value under the following sufficient condition:

> All sources in $B_{q,-i}$ are contained in $C_M(q; S)$, the retriever's relative ordering over the remaining relevant candidates is unchanged by removing s_i, and $u(q, A)$ is deterministic for a fixed context.

Under this condition,

$$A^{M}_{q,-i} = B_{q,-i}, \tag{19}$$

and therefore

$$u(q, A^{M}_{q,-i}) = u(q, B_{q,-i}). \tag{20}$$

The proof is direct. Candidate coverage ensures that every source needed by the exact post-removal

top-k context is available in the cached pool. Ranking stability ensures that filtering s_i from the cached ranking orders the remaining relevant candidates as exact reretrieval would. Taking the top k sources from the filtered cached pool therefore returns the same context as exact reretrieval. Deterministic utility then gives equality of the query-level value.

This condition is diagnostic rather than global. It can fail when useful replacement evidence lies below depth M, when removing a source changes retriever scores non-locally, or when small context changes produce large utility differences.

A.5 Replacement Coverage and Candidate-Pool Sensitivity

A useful diagnostic for candidate-pool adequacy is post-removal replacement coverage:

$$\gamma_M(q,i) = \frac{|R_k(q; S \setminus \{s_i\}) \cap C_M(q;S)|}{k}. \quad (21)$$

When $\gamma_M(q,i) = 1$, all sources that exact reretrieval would select after removing s_i are visible to the cached surrogate. Low $\gamma_M(q,i)$ indicates truncation error: useful replacement evidence lies outside the candidate pool and cannot be recovered by R-ReLOO. This diagnostic gives M an explicit methodological role.

Increasing M exposes more possible replacement sources to the approximation, but also increases cached retrieval size and utility-evaluation cost. A larger candidate pool can improve approximation quality when useful replacement evidence appears just below the original top-k cutoff. However, larger pools may also introduce lower-ranked candidates whose utility effects are noisy or unstable under the chosen utility function. The best M is therefore workload-, retriever-, and utility-dependent.

Table 9 reports the bounded sensitivity probe over $M \in \{10, 20, 50\}$. The pattern is dataset-dependent: deeper pools improve fidelity on 2WikiMultiHopQA, while HotpotQA and SciFact do not improve monotonically. These results should be read as diagnostics of local replacement depth, not as broad robustness guarantees.

A.6 Utility-Sensitive Error View

The query-level approximation error can be written exactly as

$$\begin{aligned}\epsilon_{i,M}(q) &= \delta_i^{\text{exact}}(q) - \hat{\delta}_{i,M}(q) \\ &= u(q, A^M_{q,-i}) - u(q, A^{\text{rep}}_{q,-i}).\end{aligned} \quad (22)$$

Table 9: Candidate-pool sensitivity of R-ReLOO, measured by Spearman correlation (↑) with exact reretrieval LOO on bounded induced validation pools. Bold indicates the best M and underline indicates the second-best M for each dataset.

Dataset	$M = 10$	$M = 20$	$M = 50$
HotpotQA	<u>0.8768</u>	**0.8838**	0.8132
SciFact	**0.7737**	<u>0.5932</u>	0.5098
2WikiMultiHopQA	0.8375	<u>0.9368</u>	**0.9818**

The base-context utility cancels, so the error depends only on the utility discrepancy between the cached surrogate replacement context and the exact reretrieved context.

If u is L_u-Lipschitz with respect to a context distance $d(\cdot,\cdot)$, then

$$|\epsilon_{i,M}(q)| \le L_u\, d(A^M_{q,-i}, A^{\text{rep}}_{q,-i}), \quad (23)$$

and

$$|\Delta_i - \hat{\Delta}_{i,M}| \le \frac{L_u}{|Q|} \sum_{q \in Q} d(A^M_{q,-i}, A^{\text{rep}}_{q,-i}). \quad (24)$$

This bound is conditional on the chosen utility and distance; it is not a universal guarantee for arbitrary retrievers or readers. Its purpose is to clarify that R-ReLOO is accurate when the cached pool preserves the utility-relevant replacement evidence selected by exact reretrieval.

A.7 Approximation Error Channels

The utility-sensitive view can be interpreted through three error channels. Truncation error arises when useful replacement evidence is outside $C_M(q;S)$. Ranking error arises when removing a source changes the retriever ordering in a way that cannot be represented by filtering the cached ranking. Utility sensitivity arises when small context changes produce large downstream utility differences.

This decomposition is explanatory rather than a tight worst-case bound. It motivates our empirical checks on candidate-pool depth and utility components.

A.8 Subset Utility and Decision Limits

For selection and pruning, define the utility of a retained source subset $T \subseteq S$ as

$$F(T) = \frac{1}{|Q|} \sum_{q \in Q} u(q, R_k(q;T)). \quad (25)$$

Exact reretrieval LOO measures the full-corpus removal contrast

$$F(S) - F(S \setminus \{s_i\}). \tag{26}$$

If F were monotone submodular, such marginal quantities would be closely related to diminishing-return subset selection. In retrieval-augmented workloads, however, F need not be monotone or submodular: removing a distracting source can improve utility, and removing several redundant sources can have a different effect from removing any one of them alone. R-ReLOO values should therefore be used as source-ranking and decision-support signals rather than as an optimal subset-selection guarantee.

A.9 Fidelity-Cost and Complexity Discussion

The methods can be compared through a fidelity-cost pair

$$\mathcal{P}(g) = (\mathrm{Fid}(g, \Delta), \mathrm{Cost}(g)), \tag{27}$$

where g is a valuation estimator, Fid measures agreement with exact reretrieval LOO, and Cost measures retrieval and utility-evaluation cost. Top-k-only deletion has low cost but limited replacement fidelity. Exact reretrieval LOO has maximal target fidelity but high cost. R-ReLOO is designed to move the estimator toward the high-fidelity region while keeping cost controlled by candidate-pool depth M, rather than by the number of sources n.

Table 10 summarizes the asymptotic cost of the valuation methods. The comparison separates retrieval work from utility-evaluation work because exact reretrieval is dominated by repeated retrieval, while cached-pool methods mainly incur additional utility evaluations.

The sampled Shapley and Banzhaf costs assume the bounded validation implementation, where coalition utility is evaluated by filtering cached retrieval rankings. If each coalition required fresh retrieval, the retrieval cost would increase substantially. This is why we treat these methods as bounded related baselines rather than replacements for exact reretrieval LOO or R-ReLOO.

B Extended Related Work

B.1 Retrieval-augmented NLP and evidence retrieval

Retrieval-augmented generation and retrieval-integrated language models connect external evidence with downstream prediction by retrieving documents, passages, or other evidence units before generation or classification (Lewis et al., 2020; Guu et al., 2020). KILT frames knowledge-intensive NLP tasks around a shared source snapshot and jointly evaluates downstream task performance and provenance (Petroni et al., 2021). Evidence-oriented benchmarks such as HotpotQA, SciFact, and 2WikiMultiHopQA provide multi-hop question answering, scientific claim verification, and reasoning settings with supporting facts, rationales, or evidence annotations (Yang et al., 2018; Wadden et al., 2020; Ho et al., 2020). These settings make source utility practically important: when a retrieval-augmented system depends on a large evidence corpus, one may want to rank, retain, prune, or audit sources according to their contribution to downstream utility. Our work studies this source-utility question under corpus removal and reretrieval, rather than evaluating answer generation, evidence retrieval, or claim verification accuracy alone.

B.2 Source attribution, provenance, and evidence grounding

A related line of work studies whether model predictions or generated answers are grounded in identifiable evidence. Provenance-aware benchmarks evaluate whether systems can return supporting sources in addition to task outputs (Petroni et al., 2021), while citation-oriented generation evaluates whether generated answers are supported by cited passages (Gao et al., 2023). Recent RAG attribution work further studies how to associate generated answers, statements, or tokens with retrieved documents, using citation prompting, attribution modeling, or model-internal attribution signals (Qi et al., 2024; Nematov et al., 2025). These works ask whether a particular output is supported by retrieved or cited evidence. Our question is complementary: we estimate workload-level source value. A source may be cited or retrieved for one query but be redundant across the workload if other sources provide similar support. Conversely, a rarely retrieved source may have high marginal value if it uniquely supports a difficult subset of queries. R-ReLOO therefore evaluates source contribution through a downstream utility contrast, rather than through citation presence, retrieval exposure, or per-output attribution alone.

Table 10: Complexity comparison. $m = |Q|$, $n = |S|$, M is candidate-pool depth, P is the number of sampled Shapley permutations, B is the number of sampled Banzhaf coalitions, C_R is retrieval cost, and C_u is utility-evaluation cost.

Method	Retrieval work	Utility work	Interpretation
Exact reretrieval LOO	$O(mnC_R)$	$O(mnC_u)$	Highest-fidelity validation target; reruns retrieval after each source removal.
Top-k-only deletion	$O(mC_R)$	$O(mkC_u)$	Cheapest LOO-style baseline; cannot introduce replacement evidence.
R-ReLOO	$O(mC_R)$ at depth M	$O(mMC_u)$	Cached-pool approximation; models in-pool replacement with cost controlled by M.
Sampled Shapley	cached retrieval in bounded validation	$O(PmnC_u)$	Coalition baseline over sampled permutations; not reretrieval LOO.
Sampled Banzhaf	cached retrieval in bounded validation	$O(BmnC_u)$	Coalition baseline over sampled source subsets; not reretrieval LOO.

B.3 Resource selection and corpus management

Resource selection in distributed information retrieval treats collections or resources as selectable units under relevance, coverage, duplication, cost, and efficiency constraints (Wu and Crestani, 2003). Related work on federated search and collection selection studies how to choose resources or collections so that returned results are relevant, diverse, and efficient (Si and Callan, 2003; Shokouhi, 2007; Hong and Si, 2013). This perspective is close to our motivation: retrieval-augmented systems increasingly rely on large evidence corpora, where source ranking and pruning can affect both utility and cost. However, collection-level relevance, source diversity, retrieval frequency, and retrieval score do not directly estimate the utility change caused by removing an individual source and reretrieving from the remaining corpus. Our focus is source-level marginal utility under reretrieval: when one source is removed, the retriever may promote replacement evidence, and this replacement behavior changes the measured value of the removed source.

B.4 Leave-one-out and data valuation

Data valuation methods estimate how individual data units contribute to downstream utility. Leave-one-out is a natural utility contrast, but it can be unstable or incomplete when interactions among data units matter. Game-theoretic approaches such as Data Shapley and Data Banzhaf estimate coalition-based contribution, while learning-based valuation methods such as DVRL learn data weights from downstream validation signals (Ghorbani and Zou, 2019; Wang and Jia, 2023; Yoon et al., 2020). Survey work summarizes these valuation paradigms, including their computational challenges and differences among leave-one-out, Shapley-style, Banzhaf-style, and learning-based valuation (Sim et al., 2022). These methods motivate our utility-based view of source contribution, and we include bounded sampled Shapley and Banzhaf variants as related baselines. However, our target is narrower and retrieval-specific. R-ReLOO estimates a leave-one-source-out utility contrast for a fixed workload, retriever, source granularity, retrieval depth, candidate-pool depth, and utility function. It does not average marginal contributions over all coalitions and does not provide Shapley-style fairness, pricing, or allocation guarantees.

B.5 Replacement-Aware Source Valuation

The key distinction in our setting is replacement under reretrieval. Fixed-context deletion removes a source from the already retrieved top-k context, but it does not model what the retriever would select if that source were removed from the corpus. Exact reretrieval LOO captures this behavior by removing a source, rerunning retrieval, and measuring the utility change, but it is expensive over large workloads and source collections. R-ReLOO is designed as a middle-ground estimator: it uses cached candidate pools to approximate the replacement behavior of exact reretrieval LOO while avoiding source-by-source reretrieval over the full corpus.

C Additional Experimental Details

This section provides additional details for the experimental setup, utility design, baselines, and evaluation protocol used in the main text.

C.1 Datasets and Source Units

We evaluate on three evidence-oriented NLP datasets. HotpotQA and 2WikiMultiHopQA are multi-hop question answering datasets with supporting evidence (Yang et al., 2018; Ho et al., 2020). SciFact is a scientific claim verification dataset with evidence annotations (Wadden et al., 2020). These datasets allow us to evaluate source valuation in both multi-hop QA and scientific evidence-retrieval settings.

A source is the retrievable unit used by the local evidence corpus. The main paper reports all values relative to these evaluated source units, together with the fixed retriever, workload, retrieval depth, candidate-pool depth, and utility function.

C.2 Utility Functions

For each query or claim q and retrieved context A, the utility function $u(q, A)$ measures task-relevant evidence support. Depending on the dataset, this may include evidence recall, answer-string support, claim-support overlap, retrieval precision, or a weighted combination of these components. The utility is a bounded task proxy rather than a retrieval score: it evaluates retrieved evidence against dataset supervision. This distinction is important because a source can be frequently retrieved or highly scored by the retriever while still being redundant for the workload.

C.3 Baselines

We compare against three main baseline families. Retrieval/source proxies include source size, retrieval frequency, average retrieval score, and co-retrieval degree. These test whether source value can be explained by length, exposure, retriever preference, or co-occurrence structure.

Fixed top-k-only deletion retrieves the top-k context once, removes a source from that fixed context, and measures the utility drop. It is the strongest cheap LOO-style baseline in our setting, but it is replacement-blind because it does not model which evidence would enter the context after corpus removal.

Coalition-style valuation baselines include bounded sampled Shapley and Banzhaf estimators. These baselines estimate contribution through marginal utility over sampled coalitions, but they answer a different question from reretrieval LOO. R-ReLOO targets the full-corpus removal contrast $U(S) - U(S \setminus \{s_i\})$, whereas Shapley and Banzhaf average marginal contributions over sampled source subsets. We therefore evaluate them as bounded related baselines rather than oracle targets. Formal baseline definitions are provided in Appendix D.

C.4 Evaluation Protocol

For ranking fidelity, exact reretrieval LOO is computed on bounded induced validation pools and used as the target. We report Spearman correlation between each method's source ranking and exact reretrieval LOO over five seeded repeats. We also report paired differences between R-ReLOO and the strongest baselines.

For decision support, we rank sources by their estimated values, keep or prune sources under a fixed budget, reretrieve from the modified corpus, and report retained utility and evidence recall. This evaluation is separate from ranking fidelity because keep/prune decisions apply many source decisions jointly, whereas exact LOO evaluates one source removal at a time.

For efficiency, we report measured runtime in the bounded exact-validation setting. This isolates the practical cost difference between exact reretrieval LOO, top-k-only deletion, and R-ReLOO.

D Baseline Definitions

This section defines the baseline families used in the experiments. Each baseline assigns a scalar score v_i to source s_i, which is then used for source ranking, selection, or pruning.

D.1 Fixed-Context LOO Baseline

Top-k-only deletion. For each query q, top-k-only deletion first retrieves $A_q = R_k(q; S)$ and then removes s_i only from this fixed retrieved context:

$$v_i^{\mathrm{top}k} = \frac{1}{|Q|} \sum_{q \in Q} \left[u(q, A_q) - u(q, A_q \setminus \{s_i\}) \right]. \tag{28}$$

This baseline is replacement-blind because it does not rerun retrieval after source removal. It is nevertheless a strong operational baseline: if the original top-k context already contains the decision-relevant

evidence, fixed-context sensitivity can closely track source utility.

D.2 Retrieval and Source Proxies

Average retrieval score. This baseline scores a source by its mean retriever score over the queries for which it appears in retrieved candidate lists:

$$v_i^{\text{score}} = \text{mean}_{q:s_i \in C_M(q;S)} \, \text{score}_R(q, s_i). \quad (29)$$

It measures retriever preference, but not downstream utility, redundancy, or replacement after source removal.

Retrieval frequency. This baseline scores a source by how often it appears in retrieved contexts or candidate pools:

$$v_i^{\text{freq}} = \sum_{q \in Q} \mathbb{I}[s_i \in C_M(q; S)]. \quad (30)$$

It measures exposure. Frequent sources may still be redundant, while rare sources may be decisive for a small number of difficult queries.

Source size. This baseline ranks sources by their text length or evidence size:

$$v_i^{\text{size}} = |s_i|. \quad (31)$$

It is a corpus statistic rather than a utility contrast.

Co-retrieval degree. This baseline treats source co-occurrence in retrieved candidate pools as a graph signal. Let $G = (S, E)$ be a co-retrieval graph whose nodes are sources and whose edges connect sources that appear in the same query-level retrieval neighborhood. Co-retrieval degree scores source s_i by its graph degree:

$$v_i^{\text{degree}} = \sum_{j \neq i} \mathbb{I}[(s_i, s_j) \in E]. \quad (32)$$

It captures retrieval-neighborhood connectivity, but it does not measure utility under source removal or reretrieval.

D.3 Bounded Coalition-Style Valuation Baselines

Bounded sampled Shapley. The sampled Shapley baseline draws random source permutations. For a permutation π, let P_i^π be the set of sources preceding s_i in π. The marginal contribution of s_i under this permutation is

$$F(P_i^\pi \cup \{s_i\}) - F(P_i^\pi), \quad (33)$$

where $F(T)$ denotes workload utility when retrieval is restricted to source subset T. The bounded sampled Shapley score averages this quantity over P sampled permutations:

$$v_i^{\text{Shapley}} = \frac{1}{P} \sum_{\pi=1}^{P} \left[F(P_i^\pi \cup \{s_i\}) - F(P_i^\pi) \right]. \quad (34)$$

In our bounded validation, coalition utility is computed by filtering cached retrieval rankings to the available coalition and applying the same utility function to the resulting top-k context. Thus, this is a cached-ranking Shapley-style baseline, not a full fresh-reretrieval Shapley estimator.

Bounded sampled Banzhaf. The sampled Banzhaf baseline draws random coalitions $T_b \subseteq S \setminus \{s_i\}$ and estimates the average marginal effect of adding s_i:

$$v_i^{\text{Banzhaf}} = \frac{1}{B} \sum_{b=1}^{B} \left[F(T_b \cup \{s_i\}) - F(T_b) \right]. \quad (35)$$

As with sampled Shapley, coalition utility is computed by filtering cached retrieval rankings to the available coalition. This is therefore a bounded Banzhaf-style comparison, not an exact Banzhaf value computed over all coalitions.

These coalition-style baselines are included because Shapley and Banzhaf methods are natural data valuation references. They answer a different question from reretrieval LOO: they estimate average marginal contribution under sampled source subsets, whereas R-ReLOO estimates the utility change caused by removing a source from the full corpus and approximating reretrieval. They should not be interpreted as data prices, fairness allocations, causal guarantees, or production RAG guarantees.

E Additional Results

Figure 5 provides the visual counterpart to Table 4. Positive values indicate that R-ReLOO is closer to exact reretrieval LOO than the fixed-context baseline. The consistent positive differences provide an additional view of the ranking-fidelity gains reported in the main experiments.

F Retriever-Type and Replacement-Coverage Diagnostics

This appendix section provides bounded diagnostics that further characterize the approximation be-

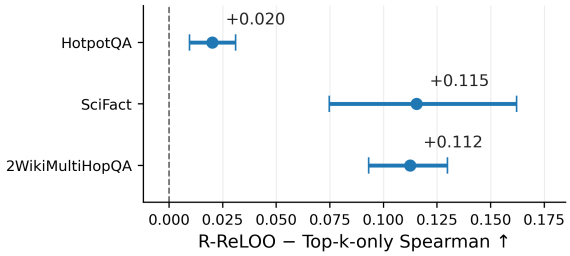

Figure 5: Paired Spearman differences between R-ReLOO and top-k-only deletion over repeated exact-validation subsets. Positive values indicate better agreement with exact reretrieval LOO.

havior of R-ReLOO. These diagnostics use seeded induced validation pools and are not intended as production retrieval-robustness guarantees.

Retriever-type diagnostic. The main experiments use an in-repository BM25 lexical retriever over source titles and texts as the default retrieval backend. The retriever uses the Okapi BM25 scoring form with $k_1 = 1.2$, $b = 0.75$, lowercase alphanumeric tokenization, and one-character token removal. To test whether the replacement-aware advantage is specific to this default backend, we run a bounded retriever-type diagnostic across all three datasets with sparse, dense, and hybrid retrieval settings. The sparse setting uses BM25. The dense settings use MiniLM (`all-MiniLM-L6-v2`) and BGE-small (`bge-small-en-v1.5`) with normalized embedding similarity. The hybrid settings combine BM25 and BGE-small using either min-max score-level interpolation with $\alpha = 0.5$ or reciprocal-rank fusion (RRF).

For BM25 and hybrid settings, the exact target is computed by removing the source and rerunning retrieval over the remaining induced pool. This recomputes BM25 statistics, hybrid score normalization, and hybrid ranks after source removal. For dense retrieval, source removal is equivalent to filtering the fixed dense ranking because the remaining query-source similarities do not change. Table 11 reports both all-source Spearman correlation and a stricter nonzero-source variant (NZ-ρ), which reduces the effect of tied zero-valued sources.

R-ReLOO improves over Top-k-only deletion in most dataset–retriever settings under NZ-ρ, suggesting that the replacement-aware advantage is not limited to the default BM25 retriever. The exception is BM25 on 2WikiMultiHopQA, where Top-k-only slightly exceeds R-ReLOO under NZ-ρ. This confirms that the advantage is retriever- and dataset-dependent rather than universal.

Replacement-coverage diagnostic. We also compute the replacement-coverage diagnostic γ_M, which measures how much of the exact post-removal top-k context is visible inside the cached candidate pool. High coverage indicates that the candidate pool contains the sources that exact reretrieval would select after source removal; low coverage indicates truncation error. Coverage is not sufficient for exact utility agreement, since ranking instability and utility sensitivity may still affect source values.

Table 12 reports coverage under the default BM25 retriever for $M \in \{10, 20, 50\}$. Coverage increases consistently with candidate-pool depth on all datasets. At $M = 50$, mean coverage reaches at least 0.980 and full-coverage rate reaches at least 0.960 across the three datasets. This supports the candidate-pool adequacy assumption used by R-ReLOO and clarifies the role of M: deeper pools reduce truncation error, but increase cached retrieval and utility-evaluation cost.

G Additional Analysis and Case Studies

This section provides qualitative details behind the diagnostic patterns summarized in the main text. These examples are used to interpret the aggregate results; they do not introduce additional empirical claims beyond the bounded validation setting.

Replacement can reduce fixed-context overestimation. In one 2WikiMultiHopQA case asking which film was released first, removing the source "The Mask of Fu Manchu" has an exact reretrieval LOO value of 0.0025. R-ReLOO matches this value, while top-k-only deletion overestimates the effect at 0.0059. The fixed-context baseline measures the utility drop after deleting the source from the observed context, but it does not account for evidence retrieved after corpus removal. This illustrates why replacement-aware valuation can better approximate exact reretrieval LOO when the deleted source's support is partially recovered by other evidence.

Top-k-only remains strong when replacement is irrelevant. Top-k-only deletion can match R-ReLOO when the observed top-k context already contains the decision-relevant evidence or when post-removal replacement does not materially change utility. This explains why top-k-only remains competitive in several decision metrics. In

Table 11: Retriever-type diagnostic with exact post-removal reretrieval targets. Values are computed on bounded induced validation pools. All-ρ denotes Spearman over all sources, while NZ-ρ denotes Spearman over sources where either exact LOO or the evaluated method is nonzero. Δ_{NZ} is R-ReLOO minus Top-k under NZ-ρ. This appendix diagnostic is not a production retriever-robustness guarantee.

Dataset	Retriever	R-ReLOO All-ρ	Top-k All-ρ	R-ReLOO NZ-ρ	Top-k NZ-ρ	Δ_{NZ}
HotpotQA	BM25	0.997	0.890	0.949	0.911	+0.039
HotpotQA	MiniLM	1.000	0.829	1.000	0.841	+0.159
HotpotQA	BGE-small	1.000	0.864	1.000	0.822	+0.178
HotpotQA	BM25+BGE score	1.000	0.901	1.000	0.928	+0.072
HotpotQA	BM25+BGE RRF	0.991	0.831	0.969	0.848	+0.120
SciFact	BM25	0.763	0.563	0.844	0.688	+0.155
SciFact	MiniLM	1.000	0.687	1.000	0.849	+0.151
SciFact	BGE-small	1.000	0.646	1.000	0.807	+0.193
SciFact	BM25+BGE score	0.950	0.676	0.957	0.794	+0.163
SciFact	BM25+BGE RRF	0.965	0.680	0.950	0.794	+0.156
2WikiMultiHopQA	BM25	0.968	0.821	0.873	0.886	-0.013
2WikiMultiHopQA	MiniLM	1.000	0.828	1.000	0.886	+0.114
2WikiMultiHopQA	BGE-small	1.000	0.876	1.000	0.922	+0.078
2WikiMultiHopQA	BM25+BGE score	0.951	0.791	0.983	0.845	+0.138
2WikiMultiHopQA	BM25+BGE RRF	0.478	0.332	0.909	0.438	+0.471

Table 12: Replacement coverage under the default BM25-style sparse retriever. Each cell reports mean γ_M / full-coverage rate, where γ_M is the fraction of exact post-removal top-k sources visible in the cached candidate pool and full coverage is the fraction of evaluated query-source pairs with $\gamma_M = 1$. Higher is better.

Dataset	$M = 10$	$M = 20$	$M = 50$
HotpotQA	0.731 / 0.338	0.872 / 0.659	**0.993 / 0.979**
SciFact	0.618 / 0.279	0.811 / 0.597	**0.980 / 0.960**
2WikiMultiHopQA	0.845 / 0.558	0.959 / 0.879	**1.000 / 1.000**

these cases, the replacement effect is near zero, so the simpler fixed-context estimate can be sufficient.

Exposure proxies can fail in both directions. Retrieval frequency, average retrieval score, source size, and co-retrieval degree are descriptive signals rather than utility contrasts. A rarely retrieved source can be valuable if it uniquely supports a difficult query or a specific evidence chain. Conversely, a frequently retrieved or highly scored source can be redundant if other sources provide similar support. This explains why retrieval-side proxies can be useful summaries but remain weaker than utility-based source valuation in the exact-validation results.

Pruning is interaction-sensitive. Removing low-valued sources can preserve or slightly improve utility when pruned sources are redundant, weakly supportive, or distracting. However, pruning changes many source interactions simultaneously. A source that appears redundant under one-at-a-time removal can become useful after nearby sources are also removed. For this reason, we evaluate keep/prune decisions by reretrieving from the modified corpus rather than assuming that single-source values solve an optimal subset-selection problem.

Candidate-pool misspecification is the main failure mode. R-ReLOO can fail when useful replacement evidence lies outside the cached pool, when source removal changes retrieval behavior beyond local rank promotion, or when the utility function is highly sensitive to small context changes. Larger candidate pools can reduce truncation error, but they also increase computation and may introduce lower-ranked candidates with noisier utility effects. Candidate-pool depth M should therefore be treated as a diagnostic design choice rather than a hidden implementation detail.

Table 13 summarizes the qualitative case-study categories used to interpret replacement effects, proxy failures, pruning behavior, and approximation errors.

H Extended Limitations

R-ReLOO is a cached-pool approximation to reretrieval LOO, not an exact valuation method. Its accuracy depends on whether the cached candidate pool contains the replacement evidence that exact reretrieval would select after source removal. If useful replacement sources lie below depth M, or

Table 13: Qualitative case-study taxonomy. The cases explain when replacement-aware valuation helps, when fixed top-k deletion is already sufficient, and when exposure-based proxies or R-ReLOO can fail.

Case type	Observed behavior	What it shows
Replacement corrects overestimation	Fixed top-k deletion assigns a larger value than exact reretrieval LOO, while R-ReLOO is closer to the exact value.	Replacement evidence can recover part of the removed source's utility; deletion from a fixed context can overestimate source importance.
Top-k-only is sufficient	Top-k-only and R-ReLOO give similar values.	When replacement has little effect, the cheaper fixed-context baseline can already approximate the exact LOO contrast.
Frequency under-ranks useful sources	A source appears rarely but has high exact LOO value for a small set of queries.	Retrieval exposure is not the same as marginal utility; rare evidence can be important.
Retrieval score under-ranks useful sources	A source has modest average retrieval score but high utility under source removal.	Retriever preference alone does not measure downstream contribution.
Frequent source is redundant	A frequently retrieved source has low or near-zero exact LOO value.	Broad exposure can reflect redundancy rather than unique contribution.
R-ReLOO misses replacement	R-ReLOO misranks a source when the useful replacement is outside the cached candidate pool.	Candidate-pool depth M controls what replacement behavior the approximation can observe.
Pruning preserves utility	Removing low-valued sources preserves or slightly improves retained utility after reretrieval.	Source values can support corpus pruning, but pruning must be evaluated after reretrieval.

if source removal changes retrieval scores in a non-local way, R-ReLOO can misestimate source value. The method also inherits the chosen utility function: values can change when utility emphasizes evidence recall, answer support, claim overlap, retrieval precision, or other task-specific signals.

The empirical evidence is bounded to HotpotQA, SciFact, and 2WikiMultiHopQA under fixed source units, retriever, utility function, retrieval depth, and candidate-pool depth. These experiments evaluate replacement-aware evidence-source utility, but they do not establish robustness across dense retrievers, hybrid retrievers, neural readers, LLM readers, stochastic generation, generated-answer quality, or production RAG systems. Extending the evaluation to these settings is important future work.

R-ReLOO values are utility-relative rather than intrinsic source worth. They should not be interpreted as data prices, Shapley or Banzhaf values, fairness allocations, or causal guarantees beyond the specified utility contrast. The decision results also show that top-k-only deletion remains competitive in some selection and pruning settings. R-ReLOO should therefore be used as replacement-aware decision support, not as a uniformly dominant or globally optimal source-selection rule.

I Artifact Use and Documentation

We use publicly available research datasets, including HotpotQA, SciFact, and 2WikiMultiHopQA, and cite their original creators in the main text. Our use is limited to research evaluation of retrieval-augmented source valuation. We do not create new human-subject data, and we do not use the datasets to identify individuals. Dataset statistics, source units, query counts, and retrieval-record counts are reported in Table 2. Code, processed experimental artifacts, and documentation are released in the anonymous repository for research use.