Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Consistency-Aware Reranking via Preference Graph Repair: Structural Gains and Conditional Retrieval Effects

Soroush Vahidi

sv96@njit.edu

New Jersey Institute of Technology

Research Article

Keywords: rank aggregation, reranking, preference graphs, feedback arc set, cyclicity, information retrieval

Posted Date: June 17th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-9335700/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Consistency-Aware Reranking via Preference Graph Repair: Structural Gains and Conditional Retrieval Effects

Soroush Vahidi[1*]

[1]Ying Wu College of Computing, New Jersey Institute of Technology, Newark, NJ, USA.

Corresponding author(s). E-mail(s): sv96@njit.edu;

Abstract

Preference-based reranking offers a flexible way to combine multiple retrieval signals, but the induced preference graph may contain directed cycles that prevent a fully consistent ordering. This paper presents a methodological study of consistency-aware reranking through preference-graph repair, focusing on the relationship among vote construction, graph cyclicity, structural alignment, and downstream retrieval effectiveness. We evaluate the framework on a canonical four-dataset package comprising SciDocs, FiQA, HotpotQA, and BRIGHT, using weighted directed graphs built from score-derived pairwise preferences. The results show that vote construction is the dominant driver of cyclicity. Majority-style constructions produce graphs that are nearly acyclic and leave little room for repair, whereas more permissive per-ranker constructions yield substantially higher cyclicity and larger strongly connected components, making repair structurally active. When repair is active, it consistently improves qrels-aligned structural diagnostics, but its effect on normalized discounted cumulative gain is conditional rather than uniform. Across datasets, repaired and unrepaired rankings are often identical in near-acyclic regimes, while in highly cyclic regimes the retrieval effect is mixed and dataset-dependent. These findings support a restrained conclusion: consistency-aware repair is more reliable as a structural intervention than as a universal mechanism for improving retrieval effectiveness. More broadly, the study shows that vote construction, graph repair, and ranking extraction should be analyzed as related but distinct components of retrieval-oriented preference aggregation.

Keywords: rank aggregation, reranking, preference graphs, feedback arc set, cyclicity, information retrieval

1 Introduction

Reranking is a standard mechanism for improving retrieval quality after an initial candidate set has been produced. Rather than replacing first-stage retrieval, it applies a stronger comparison signal to a smaller set of documents in order to refine the final order [1, 2]. This setting has become increasingly important as modern retrieval pipelines combine heterogeneous signals, including sparse and dense retrievers, cross-encoders, and large language models operating in pointwise, pairwise, or listwise modes [3, 4, 5, 6, 7, 8]. A recurring difficulty across such settings is that the available ranking evidence does not always induce a clean, globally consistent order.

A natural representation of this evidence is a weighted pairwise preference graph. If one document is preferred to another, that relation can be encoded as a directed edge; aggregating many such relations yields a query-specific graph. This view is attractive because it can absorb signals from multiple rankers or comparison paradigms while

making disagreement explicit rather than collapsing everything immediately into a single scalar score [9, 10, 11]. At the same time, it introduces a structural difficulty: the induced graph may contain directed cycles, so the available preference evidence cannot be represented exactly by a single consistent ranking.

This issue matters for retrieval because graph-level inconsistency and ranked effectiveness are related but not identical concerns. A cyclic preference graph may indicate substantial local contradiction, yet repairing that contradiction does not necessarily improve the final ranked list under a standard retrieval metric. Conversely, if the induced graph is already acyclic or nearly so, a repair step may be effectively inert. The practical question is therefore not simply whether graph repair is possible, but when it is structurally relevant and whether its structural effect translates into retrieval gains.

This paper studies that question directly. We investigate consistency-aware reranking through preference-graph repair, focusing on two linked issues: how strongly cyclic different preference-construction regimes make the induced graph, and whether repair improves ranked retrieval effectiveness when it is structurally active. To answer these questions, we compare unrepaired and repaired ranking extraction on weighted preference graphs induced from reranking evidence under several vote-construction regimes. The main experiments use a canonical four-dataset package spanning SciDocs, FiQA, HotpotQA, and BRIGHT [12, 13, 14, 15]. We also include a smaller supporting set of real LLM reranking runs to test whether the same structural pattern appears under direct language-model preference elicitation.

The main finding is conditional rather than universal. The strongest determinant of graph inconsistency is the preference-construction regime itself. Conservative construction produces graphs that are nearly always acyclic, in which case repaired and unrepaired pipelines coincide. More permissive construction yields sharply higher cyclicity and larger strongly connected components, creating the regime in which repair becomes meaningfully active. Even there, however, retrieval gains are mixed and dataset-dependent rather than uniformly positive. The resulting conclusion is therefore restrained: consistency-aware repair has clear structural value, but its retrieval benefit is conditional rather than general.

This contribution is distinct from both classical learning-to-rank and recent LLM reranking work. We do not propose a new supervised ranking objective, nor do we propose a new prompting strategy for language-model reranking. Instead, we study a downstream structural question that arises once preferences have already been obtained: what changes when those preferences are represented as a graph, found to be cyclic, and repaired before ranking extraction? By separating preference construction, graph repair, and ranking extraction, the paper shows that these are distinct design choices with distinct empirical consequences.

The rest of the paper is organized as follows. Section 2 situates the study within learning to rank, reranking, and preference-based ranking. Section 3 presents the preference-graph construction, repair, and ranking framework. Section 4 describes the datasets, rankers, construction regimes, baselines, and evaluation protocol. Section 5 reports the main empirical results, including cyclicity patterns and conditional repair effects. Section 6 discusses limitations, and Section 7 concludes.

2 Related Work

2.1 Learning to Rank and Reranking

Learning to rank provides the standard framework for improving ranked retrieval from supervision over relevance or preference signals [1, 16]. In modern retrieval systems, this commonly appears as reranking: a first-stage retriever generates a candidate set, and a stronger second-stage model or aggregation rule reorders that set to improve top-ranked effectiveness [17, 18]. This two-stage setting is especially important when high-quality comparison signals are too expensive to apply over the full corpus. It is also the setting considered in this paper.

Classical learning-to-rank methods are usually organized into pointwise, pairwise, and listwise formulations. RankNet and related work helped establish pairwise comparison and metric-aware ranking optimization as central ideas [2], while later reranking systems demonstrated the practical gains achievable when stronger models are

applied to limited candidate pools [17, 16]. At the same time, practical retrieval pipelines often go beyond a single learned reranker. They may combine sparse and dense signals, fuse heterogeneous rankings, or aggregate multiple retrieval views into a final order [19, 20, 21, 22, 23]. The present work belongs to this broader reranking and aggregation setting.

Recent work has expanded reranking further through large language models. Studies of LLM-based reranking report meaningful differences among pointwise, pairwise, listwise, and setwise formulations, together with nontrivial trade-offs between effectiveness and computational cost [3, 4, 5, 6, 7, 8]. This literature matters here for two reasons. First, it reinforces that reranking is a design space rather than a single method. Second, it makes pairwise preference elicitation newly important, since many modern rerankers operate through direct document-to-document comparisons rather than only through scalar relevance estimation.

Overall, prior work shows that reranking can arise from supervised learning, rank fusion, and increasingly from direct pairwise comparison with large language models. The present study is complementary to these directions. Rather than proposing a new upstream reranker, it examines what happens after comparative evidence has already been obtained and represented as a weighted directed preference graph. This naturally connects the reranking literature to the preference-modeling and graph-based ranking work reviewed next.

2.2 Preference Modeling and Graph-Based Ranking

Preference modeling provides a natural bridge between reranking signals and graph-based aggregation. Rather than assigning each document an absolute relevance score in isolation, preference-based methods represent evidence through comparisons such as $u \succ v$, meaning that document u should be ranked ahead of document v. This view is classical in ranking theory and underlies both probabilistic comparison models and combinatorial rank-aggregation methods [24, 25, 9, 10]. It is especially useful when multiple rankers or judgments disagree, because disagreement can be represented directly as directional evidence instead of being collapsed immediately into a single scalar score.

A standard statistical foundation for this perspective is given by Bradley–Terry-type models, which explain pairwise outcomes through latent item strengths [24, 25, 26, 27]. Related work extends the same preference-centered view to spectral and algebraic ranking methods, including Rank Centrality and Hodge-theoretic decompositions, both of which treat pairwise comparisons as structured signals on a graph [11, 28, 29]. Taken together, these lines of work show that ranking from comparisons is not only a matter of estimating an order, but also of understanding the consistency structure of the comparison data itself.

Graph-based ranking makes that structure explicit. Given weighted pairwise preferences, one can form a directed graph whose vertices are documents and whose edges encode support for one document over another. Rankings can then be extracted in several ways, including win-based summaries, random-walk interpretations, and degree-like scores. In the present manuscript, we focus on Copeland-style and weighted balance scores. Their value here is primarily methodological: they provide transparent and reproducible summaries of the induced graph and therefore make it easier to isolate the effect of graph repair on downstream rankings.

This graph view also connects naturally to rank aggregation under inconsistency. When pairwise evidence is transitive, the induced graph is acyclic and ranking extraction is comparatively straightforward. In practice, however, aggregated preferences often contain directed cycles, so no total order can satisfy all pairwise directions simultaneously. Classical work on rank aggregation and approximation treats this as a central difficulty [9, 10, 30, 31, 32]. From that perspective, cyclicity is not merely noise; it is a structural indication that the available preference evidence is internally contradictory.

The present paper builds on that observation but takes a narrower empirical focus. We do not propose a new general model of pairwise preference learning. Instead, we study a downstream question inside modern reranking pipelines: once preferences have been constructed and represented as a weighted directed graph, when does structural inconsistency materially affect the extracted

ranking, and when does repairing that inconsistency matter? In this way, the paper fits within the broader literature on preference modeling and graph-based ranking while keeping its claim appropriately specific: cyclicity, repair, and retrieval impact should be analyzed jointly rather than assumed to move together.

3 Method

3.1 Problem Setting

Let q denote a query and let $D_q = \{d_1, \ldots, d_n\}$ be its candidate document set. The input to our framework is not raw relevance supervision, but an existing retrieval signal that can be converted into weighted pairwise preferences over document pairs. The task is to produce a final query-specific ranking while accounting for the fact that the induced preference graph may be cyclic and therefore may not admit a globally consistent order.

For each query, we construct a weighted directed graph $G_q = (V_q, E_q, w_q)$ with $V_q = D_q$. A directed edge $(u, v) \in E_q$ indicates evidence that document u should be ranked ahead of document v, and $w_q(u, v) > 0$ denotes the strength of that preference. This representation is closely related to classical rank aggregation and pairwise-comparison ranking [9, 10, 24, 25, 11]. In the present study, however, the goal is not to fit a general probabilistic comparison model. Instead, the emphasis is on how the structure of the induced graph interacts with repair and downstream retrieval effectiveness.

The main structural difficulty is that G_q need not be acyclic. If it contains directed cycles, then the available preference evidence cannot be represented exactly by a single total order. In this manuscript, that issue is central rather than incidental: the amount of cyclicity depends strongly on the preference-construction regime. Conservative construction yields graphs that are nearly always acyclic, whereas more permissive construction can produce larger strongly connected components and denser contradictions. The empirical problem is therefore twofold: to characterize when the induced graph is structurally inconsistent, and to determine whether repairing that inconsistency changes the final ranking in a useful way.

The framework has three stages. First, an upstream ranking signal is converted into weighted pairwise preferences under one of the vote-construction regimes described later. Second, the resulting graph is either left unchanged or passed through a consistency-aware repair step that removes enough conflicting edge mass to make the graph acyclic. This connects the method to feedback-arc-set-style reasoning for ranking from directed comparisons [10, 30]. Third, a final ranking is extracted from either the unrepaired or the repaired graph using a downstream graph-based scoring rule, chiefly Copeland- and balance-based aggregation in the canonical experiments.

The central empirical question is therefore not simply whether one graph score is better than another. Rather, it is whether the same induced preference graph yields a different and better reranked output once it is made more structurally consistent, and under which construction conditions that change is possible at all. In low-cyclicity regimes, repaired and unrepaired pipelines often coincide because repair has little practical work to do. Only when the construction rule retains enough contradictory structure does repair become meaningfully operative.

Accordingly, for each query we compare the prior ranking with rankings extracted from both unrepaired and repaired versions of G_q. This separates three distinct objects: the upstream ranking signal, the induced preference graph, and the final extracted ranking. That separation is important because observed gains or failures may arise from the interaction among preference construction, repair, and extraction rather than from any single component in isolation.

In summary, the problem setting is query-specific reranking from weighted pairwise preferences under possible structural inconsistency. The method operates on induced preference graphs, optionally repairs cyclic contradictions, and then extracts a final ranking for evaluation with standard retrieval metrics. This formulation makes it possible to study both ranked effectiveness and the structural conditions under which consistency-aware repair is inactive, active but neutral, or conditionally useful.

3.2 Preference Graph Construction

For each query q, the reranking pipeline produces a common candidate set $D_q = \{d_1, \ldots, d_n\}$ together with document-level scores from the

constituent rankers. In the canonical publication package, these upstream signals are score files derived from BM25, TF–IDF, and MiniLM rather than direct human or LLM pairwise judgments. We convert these signals into a weighted directed graph

$$G_q = (V_q, E_q, w_q),$$

where $V_q = D_q$, each directed edge $(u, v) \in E_q$ indicates support for ranking document u ahead of document v, and $w_q(u, v)$ denotes the retained support for that direction. This stage is central because it determines whether the induced graph is nearly acyclic or strongly cyclic, and therefore whether consistency-aware repair is later inactive or structurally meaningful.

Construction is performed query by query on a shared document universe. For each ranker, every unordered document pair contributes a directional vote whenever that ranker scores one document above the other. Vote strength is derived from the corresponding score margin, subject to the repository's minimum vote-margin threshold. These ranker-level directional votes are then aggregated into graph edges. If support concentrates on one direction, the graph remains relatively clean; if support is split, both (u, v) and (v, u) may be retained, creating mutual contradictions and larger directed cycles.

We study three explicit vote-construction regimes, summarized in Table 1. The conservative regime, `ms2`, retains a direction only when it has majority-style support, which suppresses most direct contradictions and produces graphs that are nearly always acyclic in the canonical package. The more permissive regime, `ms1`, preserves a broader set of ranker-supported directions and allows opposing edges to coexist when rankers disagree, making cyclic structure much more likely. The third regime, `ms1_drop_mutual`, starts from `ms1` but removes unordered document pairs that appear in both directions before downstream aggregation. As the results later show, this design choice has a large structural effect: `ms2` and most `ms1_drop_mutual` graphs remain nearly acyclic, whereas `ms1` yields the strongly cyclic regime in which repair becomes operative.

Algorithm 1 summarizes the construction procedure. The description is intentionally compact because the contribution is not a new preference extractor, but an analysis of how construction choices affect cyclicity, repair, and downstream retrieval. The graph is fully constructed before any repair is applied, so later differences between repaired and unrepaired rankings can be traced to the structure induced here.

Algorithm 1 Preference graph construction for a query

Require: Query q, candidate documents D_q, ranker score outputs, construction regime $r \in \{\texttt{ms2}, \texttt{ms1}, \texttt{ms1_drop_mutual}\}$
Ensure: Weighted directed preference graph $G_q = (V_q, E_q, w_q)$

```
V_q ← D_q
Initialize an empty weighted edge map w_q
for each constituent ranker do
    for each unordered document pair {u, v} in D_q do
        if the ranker supports u ≻ v then
            Accumulate directional support for (u, v) in w_q
        else if the ranker supports v ≻ u then
            Accumulate directional support for (v, u) in w_q
        end if
    end for
end for
Apply the retention rule associated with r
if r = ms1_drop_mutual then
    Remove unordered pairs that remain in both directions
end if
Construct E_q from retained directions with positive weight
return G_q = (V_q, E_q, w_q)
```

Two properties matter for the later analysis. First, the graph is weighted rather than binary, so the repair step acts on conflicting edge mass rather than on arbitrary unweighted edges. Second, graph construction is separated from graph scoring: the same induced graph can later be ranked by different extraction rules, such as Copeland- or balance-based scoring. This separation allows the experiments to distinguish the effect of preference construction from the effect of ranking extraction.

For that reason, the principal results are reported by regime rather than pooled. If construction already suppresses contradictions, repaired and unrepaired pipelines will coincide regardless of the repair method. If contradictory

Table 1 Vote-construction regimes used in the canonical publication suite. All three regimes are built from the same candidate set and the same underlying ranker score files; they differ only in how directional votes are retained.

Variant	Min support	Min margin	Rule
`ms2`	2	0.1	Retain a direction only if at least two rankers support it and the retained support exceeds the aggregate threshold.
`ms1`	1	0.0	Retain every ranker-supported directional vote, allowing both directions of the same unordered pair when rankers disagree.
`ms1_drop_mutual`	1	0.0	Start from `ms1` and remove all unordered pairs that appear in both directions.

directions are retained, the graph may contain larger strongly connected components and repair can become nontrivial. Preference graph construction is therefore not a passive intermediate representation; it is the structural stage that determines whether the paper's central intervention is active at all.

3.3 Consistency-Aware Repair and Ranking Extraction

Once a query-specific preference graph $G_q = (V_q, E_q, w_q)$ has been constructed, the next question is whether its cyclic structure should be repaired before ranking extraction. In the repository, repair is implemented as a shared preprocessing step on the weighted directed graph. If G_q is acyclic, it is left unchanged. Otherwise, it is converted into a directed acyclic graph (DAG) by repeatedly identifying a directed cycle and removing the minimum-weight edge on that cycle. This produces a repaired graph $\widetilde{G}_q$ together with the total removed edge weight, reported in the repository as feedback-arc mass removed.

This procedure is intentionally simple and deterministic. It does not claim to solve the minimum-weight feedback arc set problem exactly on the full real-data graphs. Instead, it uses a greedy cycle-peeling heuristic that can be applied reproducibly across queries. The repository also includes exact routines for smaller diagnostic settings, but the canonical real-data experiments use the greedy repair pipeline. The manuscript's claim is therefore empirical rather than optimality-based: a consistency-aware repair step can be applied consistently, and its structural and retrieval effects can then be measured directly.

Ranking extraction is performed on either the original graph G_q or the repaired graph $\widetilde{G}_q$. The canonical experiments focus on two transparent graph-based extraction rules that can be applied before and after repair. The first is a Copeland-style score, defined as out-degree minus in-degree. The second is a weighted balance score, defined as outgoing edge mass minus incoming edge mass. Using the same score family on both the original and repaired graphs isolates the effect of repair from the effect of the extraction rule itself.

For the hybrid methods in the canonical package, the graph-derived component is combined with a prior score vector from the upstream retrieval pipeline. Both components are normalized query-wise, and the final score is formed as the prior score plus a scaled graph component:

$$\mathrm{rank}(q) = \mathrm{sort}(s_{\mathrm{prior}}(q) + \alpha\, s_{\mathrm{graph}}(q))\,. \qquad (1)$$

When s_{graph} is computed from G_q, the method is unrepaired; when it is computed from $\widetilde{G}_q$, the method is repaired. In the Copeland variant,

$$s_{\mathrm{cop}}(u) = \deg^+(u) - \deg^-(u), \qquad (2)$$

and in the weighted balance variant,

$$s_{\mathrm{bal}}(u) = \sum_{(u,v)\in E} w(u,v) - \sum_{(v,u)\in E} w(v,u). \qquad (3)$$

The scientific interpretation does not depend on a single value of α; what matters is that the same hybrid template is used before and after repair so that any difference can be attributed to the structural intervention rather than to a change in extraction rule.

Algorithm 2 summarizes the procedure.

This setup clarifies the later empirical findings. If the original graph is already acyclic or nearly so, then $\widetilde{G}_q$ will be identical or very close

Algorithm 2 Consistency-aware repair and ranking extraction for a query

Require: Weighted preference graph G_q, prior scores s_{prior}, extraction component $c \in \{\textsc{Copeland}, \textsc{Balance}\}$

Ensure: Unrepaired and repaired rankings

$G_q^{(0)} \leftarrow G_q$
$\widetilde{G}_q \leftarrow G_q$
while $\widetilde{G}_q$ contains a directed cycle **do**
    Find one directed cycle in $\widetilde{G}_q$
    Remove the minimum-weight edge on that cycle
end while
Compute graph scores $s_c^{(0)}$ on $G_q^{(0)}$
Compute graph scores $\widetilde{s}_c$ on $\widetilde{G}_q$
Form unrepaired ranking from $s_{\text{prior}} + \alpha s_c^{(0)}$
Form repaired ranking from $s_{\text{prior}} + \alpha \widetilde{s}_c$
return unrepaired and repaired rankings

to G_q, and repaired and unrepaired rankings will coincide. If the graph contains larger strongly connected regions, cycle peeling can alter node scores and change the final order. The effect of repair is therefore conditional: it depends first on whether graph construction created enough cyclic structure for repair to act, and second on whether the removed low-weight cyclic edges materially affect the extracted ranking.

This is also why the manuscript reports structural and retrieval outcomes separately. Removed edge mass and qrels-aligned inconsistency measures describe what repair changed in the graph, whereas ΔnDCG describes what changed in ranked effectiveness. As the repository evidence later shows, these are related but distinct: repair can be structurally active without producing a reliable retrieval gain, and in near-acyclic regimes it is often inactive altogether.

4 Experimental Setup

4.1 Datasets and Rankers

Our evaluation uses four retrieval datasets from the canonical evidence package: SciDocs, FiQA, HotpotQA, and BRIGHT [12, 13, 14, 15]. These datasets span scientific retrieval, financial retrieval, multi-hop question answering retrieval, and reasoning-intensive retrieval. The first three also appear in modern zero-shot retrieval benchmarking [18], while BRIGHT provides a more recent reasoning-oriented setting [15]. This mix is intended to test whether the structural behavior of induced preference graphs is robust across domains rather than tied to a single corpus type.

The main experiments use a non-LLM multi-ranker pipeline in which upstream retrieval scores are converted into pairwise preferences and then aggregated by graph-based rules. In the canonical publication suite, the underlying score signals are derived from BM25, TF–IDF, and MiniLM. These rankers are used to generate a shared candidate pool and the score-derived pairwise votes described earlier. The central method comparisons are between unrepaired and repaired variants of Copeland- and balance-based hybrid aggregation, reported later as UCO/RCO and UBA/RBA. These methods are evaluated relative to the prior ranking because the purpose of the paper is to study how preference construction and repair change the final reranked output, not whether they replace the upstream retrieval pipeline altogether.

The repository also contains a broader non-LLM comparison set, including a cross-encoder and alternative aggregation rules such as Bradley–Terry, win-rate, Markov-style aggregation, and tournament sorting. We use these methods as contextual baselines, but the manuscript's primary comparisons remain repaired versus unrepaired graph-based reranking, since those isolate the effect of consistency-aware repair most directly.

The manuscript further includes a smaller stream of real LLM reranking evidence. Real OpenAI pairwise runs are available on SciDocs and HotpotQA; real OpenAI pointwise and listwise runs are currently available on SciDocs; and a small Gemini pairwise pilot on SciDocs is used only as supporting evidence. Accordingly, we treat the four-dataset non-LLM package as the main evidence stream and the LLM runs as a narrower supporting testbed. This distinction is important for positioning: the paper is not presented as a comprehensive LLM reranking benchmark, but as a structural study whose main claims are supported by the canonical non-LLM package and then checked on a smaller modern LLM setting.

Table 2 Datasets and ranker families used in the manuscript. The four-dataset package supports the main non-LLM analysis; the real LLM runs provide supporting evidence.

Dataset	Domain	Main use	Ranker families considered
SciDocs [12]	Scientific retrieval	Canonical non-LLM results; real LLM evidence	BM25/TF–IDF/MiniLM prior + graph-based hybrids (UCO/RCO, UBA/RBA); broader non-LLM baselines; real LLM pointwise, pairwise, and listwise
FiQA [13]	Financial retrieval	Canonical non-LLM results	BM25/TF–IDF/MiniLM prior + graph-based hybrids; broader non-LLM baselines
HotpotQA [14]	Multi-hop QA retrieval	Canonical non-LLM results; real LLM pairwise evidence	BM25/TF–IDF/MiniLM prior + graph-based hybrids; broader non-LLM baselines; real LLM pairwise
BRIGHT [15]	Reasoning-intensive retrieval	Canonical non-LLM results	BM25/TF–IDF/MiniLM prior + graph-based hybrids; broader non-LLM baselines

4.2 Preference Construction Regimes

A central design choice in this study is how upstream ranking evidence is converted into weighted pairwise preferences before graph aggregation. Rather than treating this step as fixed preprocessing, we evaluate three explicit vote-construction regimes: `ms2`, `ms1`, and `ms1_drop_mutual`. This distinction is important because the induced graph structure depends more strongly on preference construction than on the downstream repair step. In particular, the three regimes produce markedly different cyclicity profiles, which in turn determine whether repair is inactive, merely structural, or potentially retrieval-relevant.

The `ms2` regime is the most conservative. It retains only directions with majority-style support and a positive aggregate margin, thereby suppressing most direct contradictions. The `ms1` regime is more permissive. It preserves every ranker-supported directional vote and therefore allows opposing directions for the same document pair to remain in the graph when rankers disagree. The third regime, `ms1_drop_mutual`, begins from `ms1` and removes unordered document pairs that appear in both directions before downstream aggregation. As a result, `ms1_drop_mutual` isolates the contribution of direct two-way contradictions while remaining closer in spirit to `ms1` than to the stricter majority-style construction.

These regimes play distinct analytical roles. `ms2` provides a near-acyclic reference condition in which graph repair often has little or no opportunity to act. `ms1` creates the high-cyclicity setting in which repair becomes meaningfully testable. `ms1_drop_mutual` serves as an intermediate condition that reveals how much of the observed cyclicity is driven specifically by mutual contradictions. This regime-based design is what allows the paper to separate structural conclusions from effectiveness conclusions. When repaired and unrepaired rankings coincide, the correct interpretation is often not that repair failed, but that the chosen construction rule produced a graph in which repair had little practical work to do.

For this reason, all principal graph-based results are reported separately by vote-construction regime rather than pooled. The central retrieval question is not whether repair always helps, but under which preference-construction conditions repair becomes structurally active and whether that activity translates into measurable ranking gains. Preference construction is therefore not a minor implementation detail; it is one of the main experimental factors of the study.

4.3 Baselines and Evaluation Metrics

The evaluation is organized around a narrow primary comparison family and a broader contextual one. The primary comparisons are between the prior ranking and the graph-based hybrid methods derived from it: unrepaired and repaired Copeland

Table 3 Summary of the three preference-construction regimes used in the manuscript. All three are built from the same candidate set and the same underlying ranker score files; they differ only in how directional votes are retained.

Regime	Min support	Min margin	Construction character	Main analytical role
`ms2`	2	0.1	Conservative majority-style retention that suppresses most direct contradictions	Low-cyclicity reference condition
`ms1`	1	0.0	Permissive retention that allows contradictory directions to persist	Main regime for testing whether repair can matter
`ms1_drop_mutual`	1	0.0	`ms1` with mutual contradictions removed before aggregation	Isolates the contribution of mutual contradictions to cyclicity

(UCO/RCO) and unrepaired and repaired balance (UBA/RBA). These comparisons are central because they isolate the effect of graph repair while holding the upstream ranking signal fixed. The resulting contrasts therefore test whether changing structural consistency in the same induced preference graph changes the final ranking.

To place these comparisons in context, the repository also includes a wider set of non-LLM aggregation baselines, including Bradley–Terry, win-rate aggregation, Markov-style aggregation, and tournament-style sorting [24, 25, 11]. These methods help show that the structural findings are not tied to a single extraction rule. Even so, the manuscript keeps the repaired-versus-unrepaired graph comparisons at the center, since they address the paper's main question most directly.

A separate supporting evidence stream comes from the real LLM experiments. There, the comparison extends beyond graph rules to reranking paradigms such as pointwise, pairwise, and listwise [3, 4, 5, 6]. In this manuscript, however, those experiments play a secondary role: they are used to check whether the structural patterns seen in the larger non-LLM package also appear under direct language-model scoring or preference elicitation.

For effectiveness, the main metric is nDCG, reported at the cutoff used in the canonical package [33]. We report mean nDCG by dataset, method, and vote-construction regime, and we also analyze paired repaired-versus-unrepaired differences in nDCG to isolate the marginal effect of repair. For structure, we report the percentage of cyclic query-specific graphs and the mean size of the largest strongly connected component. In the repair-specific analysis, we additionally report mean feedback-arc mass removed as a direct measure of intervention strength. Where relevant, we also report qrels-aligned structural diagnostics such as backward-edge weight and pairwise inconsistency count in order to separate graph-internal change from ranked retrieval change.

Uncertainty is handled with paired query-level bootstrap analysis. For repaired-versus-unrepaired comparisons, we report mean ΔnDCG together with bootstrap confidence intervals. This is important because many descriptive differences are very small, and mean gaps alone can overstate the evidence for a real effect. In many rows outside the strongly cyclic `ms1` regime, the paired difference collapses to an exact or effectively zero result, which is itself informative: it indicates that the induced graph left repair with little or no practical opportunity to change the ranking.

This design keeps the manuscript focused while separating its evidence streams clearly. The prior ranking and repaired/unrepaired graph hybrids provide the main comparison, the broader aggregation baselines provide context, and the LLM paradigms provide a smaller supporting testbed. Likewise, the metric suite is intentionally split between retrieval effectiveness and graph structure, because the central empirical question is not only whether repair changes graph diagnostics, but whether those structural changes translate into retrieval gains.

Table 4 Baselines and evaluation metrics used in the manuscript.

Category	Items	Main purpose
Primary baselines	Prior, UCO, RCO, UBA, RBA	Isolate the effect of graph repair while keeping the upstream ranking signal fixed
Broader non-LLM baselines	Bradley–Terry, win-rate, Markov-style, tournament-style, related aggregation rules	Provide context beyond the core repaired/unrepaired comparisons
LLM paradigms	Pointwise, pairwise, listwise	Check whether the structural pattern also appears under direct language-model scoring or preference elicitation
Effectiveness metric	nDCG	Measure ranked retrieval quality
Structural diagnostics	Cyclic-query %, largest SCC, feedback-arc mass removed; qrels-aligned BEW/PIC where relevant	Measure inconsistency and intervention strength
Uncertainty analysis	Paired bootstrap confidence intervals for ΔnDCG	Assess whether repaired-versus-unrepaired differences are reliable or effectively zero

5 Results and Discussion

5.1 Overall Retrieval Performance

We begin with aggregate retrieval results on the four datasets in the canonical package: SciDocs, FiQA, HotpotQA, and BRIGHT [12, 13, 14, 15]. The main question is whether graph repair improves ranking effectiveness broadly. Table 5 and Figure 1 show that it does not do so uniformly. Instead, its effect depends on the vote-construction regime, the dataset, and the graph scoring rule.

The dominant pattern is that repair is largely inactive in low-cyclicity regimes and becomes relevant only in the more cyclic `ms1` setting. Under `ms2`, repaired and unrepaired variants are identical on all four datasets. Under `ms1_drop_mutual`, the same near-inactivity remains in most cases. Nonzero repaired-versus-unrepaired differences appear only under `ms1`, where cyclicity is highest.

Even within `ms1`, however, the retrieval effect is mixed. On SciDocs, the prior ranking remains strongest, and repair produces at most negligible changes. On FiQA, repaired Copeland yields a small gain over unrepaired Copeland, but the magnitude is modest. On HotpotQA, repaired Copeland improves over unrepaired Copeland, yet the balance-based variants remain strongest overall. On BRIGHT, the prior ranking is best under `ms2`, whereas under `ms1` and `ms1_drop_mutual` the balance-based method is strongest and repair leaves it unchanged. These aggregate results therefore support a conditional rather than universal view of repair.

A second pattern is that the graph scoring rule often matters at least as much as repair itself. Balance-based aggregation is strongest on HotpotQA and BRIGHT, whereas SciDocs favors either the prior ranking or Copeland depending on the regime. This reinforces a central point of the paper: vote construction, graph scoring, and graph repair are distinct design choices rather than a single monolithic reranking procedure. Descriptive mean differences indicate where repair is active, but they are not, by themselves, evidence of a reliable effectiveness gain; that question is addressed by the paired uncertainty analysis later.

Overall, the aggregate results support a restrained interpretation. Repair is not a uniformly stronger reranking strategy. Its effect depends first on whether the construction regime induces enough cyclic structure for repair to act, and only then on whether that structural intervention changes ranked effectiveness. This motivates the next subsection, which examines how vote construction changes cyclicity and thereby determines whether repair is active at all.

Table 5 Mean nDCG for the prior ranking and the graph-based hybrid methods across datasets and vote-construction regimes. Prior denotes the pre-graph hybrid ranking; UCO/RCO denote unrepaired/repaired Copeland; UBA/RBA denote unrepaired/repaired balance. Bold indicates the best mean nDCG within each dataset–variant row.

Dataset	Variant	Prior	UCO	RCO	UBA	RBA
SciDocs	`ms2`	0.3068	**0.3080**	**0.3080**	0.3023	0.3023
SciDocs	`ms1`	**0.3065**	0.3023	0.3021	0.3016	0.3016
SciDocs	`ms1_drop_mutual`	**0.3065**	0.3023	0.3023	0.3017	0.3017
FiQA	`ms2`	0.2254	0.2287	0.2287	**0.2324**	**0.2324**
FiQA	`ms1`	0.2372	0.2414	**0.2428**	0.2415	0.2415
FiQA	`ms1_drop_mutual`	0.2372	0.2414	0.2414	**0.2417**	**0.2417**
HotpotQA	`ms2`	0.8629	0.8550	0.8550	**0.8930**	**0.8930**
HotpotQA	`ms1`	0.8767	0.8892	0.9059	**0.9094**	**0.9094**
HotpotQA	`ms1_drop_mutual`	0.8767	0.8892	0.8892	**0.9094**	**0.9094**
BRIGHT	`ms2`	**0.5681**	0.5539	0.5539	0.5651	0.5651
BRIGHT	`ms1`	0.5228	0.5201	0.5201	**0.5323**	**0.5323**
BRIGHT	`ms1_drop_mutual`	0.5228	0.5201	0.5201	**0.5302**	**0.5302**

5.2 Cyclicity and Structural Effects

The strongest structural effect in the canonical package comes from the vote-construction regime rather than from repair itself. Table 6 and Figure 2 show a clear regime shift. Under `ms2`, the induced preference graphs are effectively acyclic on all four datasets. Under `ms1`, cyclicity rises sharply and larger strongly connected components appear. Under `ms1_drop_mutual`, the graphs return to near-acyclic behavior in almost all cases. The main structural conclusion is therefore that inconsistency is driven primarily by preference construction.

The contrast between `ms2` and `ms1` is substantial. For SciDocs, FiQA, HotpotQA, and BRIGHT, `ms2` yields 0% cyclic queries and an average largest SCC of 1.0, indicating topologically simple query graphs. Under `ms1`, cyclic-query prevalence rises to 87.50%, 95.00%, 51.92%, and 60.00%, while the mean largest SCC increases to 9.33, 12.51, 2.52, and 6.48, respectively. This is a genuine shift from nearly orderable graphs to graphs with substantial contradictory structure.

The filtered regime `ms1_drop_mutual` largely restores the low-cyclicity behavior of `ms2`. For SciDocs, FiQA, and HotpotQA, cyclicity returns to 0% and the average largest SCC returns to 1.0. BRIGHT remains a mild exception, with 6.00% cyclic queries and an average largest SCC of 1.40, but even there the residual inconsistency is far smaller than under `ms1`. This pattern indicates that much of the observed cyclicity arises from allowing opposing directions for the same document pair to persist.

These structural results explain the later effectiveness findings. In near-acyclic settings, repair has little practical work to do, so repaired and unrepaired pipelines coincide. In the higher-cyclicity `ms1` regime, repair becomes structurally meaningful because the graph contains larger strongly connected regions that cannot be ordered without removing conflicting edge mass. The key point is therefore not that repair always changes the graph, but that vote construction determines whether repair is operative at all.

Taken together, these results identify vote construction as the primary determinant of structural inconsistency in the induced preference graphs. The central structural story of the paper is therefore regime-dependent: the graph must first be made sufficiently cyclic for repair to have a meaningful opportunity to act. The next subsection examines whether those structural differences translate into retrieval gains.

5.3 Conditional Effects of Repair

The retrieval evidence shows that the effect of repair is conditional rather than uniform. Table 7 and Figure 3 summarize repaired-versus-unrepaired Copeland differences using paired query-level bootstrap estimates in the canonical four-dataset package. The central pattern is clear. In the near-acyclic regimes, repair is inactive and the retrieval difference collapses to zero. Under

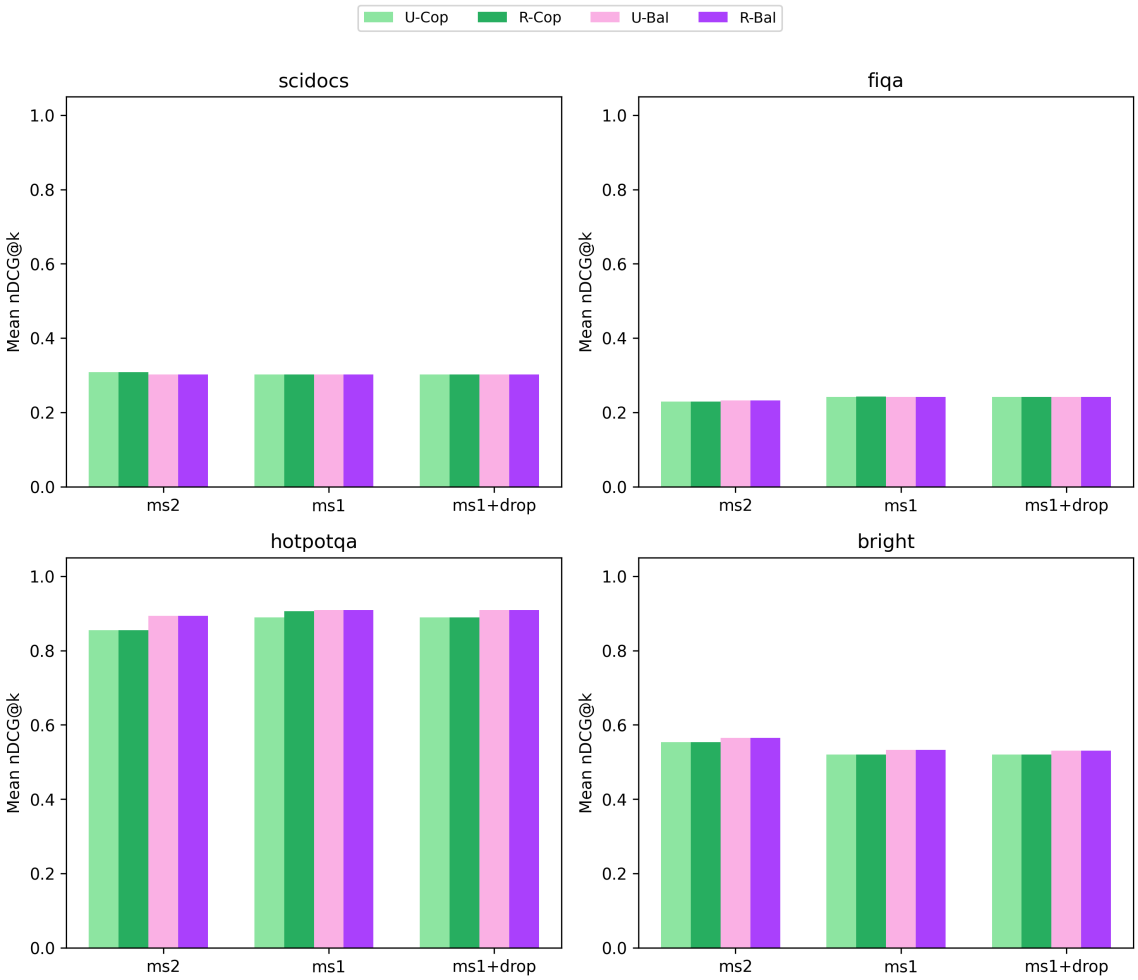


Fig. 1 Mean nDCG of the evaluated hybrid methods across datasets and vote-construction regimes. Repair is largely inactive in the near-acyclic regimes, while in the more cyclic `ms1` regime its effect is mixed and generally smaller than the dataset-specific differences between graph scoring rules.

`ms2`, all four datasets are degenerate at $\Delta\text{nDCG} = 0$, which is exactly what one would expect from the structural results in the previous subsection. Under `ms1_drop_mutual`, the same practical inactivity remains for SciDocs, FiQA, and HotpotQA, with BRIGHT showing only negligible residual repair activity and still a degenerate zero retrieval difference. Once the graph is effectively acyclic, repair has no meaningful downstream effect.

The only regime in which repair becomes substantively testable is `ms1`, where cyclicity is highest. Even there, however, the direction and magnitude of the retrieval effect vary by dataset. On SciDocs, the mean repaired-minus-unrepaired difference is slightly negative (-0.000127), with a confidence interval that straddles zero. On FiQA, the mean difference is positive (0.001456), but again the interval crosses zero. On BRIGHT, the mean difference is essentially nil (0.000022), with a narrow interval centered near zero. These three cases point to the same practical conclusion: once paired uncertainty is taken into account, the evidence does not support a reliable retrieval gain from repair.

HotpotQA is the only dataset in the canonical package that shows a clearly positive repaired-versus-unrepaired effect in the structurally active regime. Its mean ΔnDCG under `ms1` is 0.016713, and the lower bound of the interval is reported at 0.000000. Even this case should be interpreted cautiously. First, it is dataset-specific rather than general. Second, the broader paper-level pattern

Table 6 Cyclicity summary by dataset and vote-construction regime. Cyclic % denotes the percentage of query-specific graphs containing at least one directed cycle. Largest SCC is the mean size of the largest strongly connected component.

Dataset	Variant	Cyclic %	Largest SCC
SciDocs	`ms2`	0.00	1.00
SciDocs	`ms1`	87.50	9.33
SciDocs	`ms1_drop_mutual`	0.00	1.00
FiQA	`ms2`	0.00	1.00
FiQA	`ms1`	95.00	12.51
FiQA	`ms1_drop_mutual`	0.00	1.00
HotpotQA	`ms2`	0.00	1.00
HotpotQA	`ms1`	51.92	2.52
HotpotQA	`ms1_drop_mutual`	0.00	1.00
BRIGHT	`ms2`	0.00	1.00
BRIGHT	`ms1`	60.00	6.48
BRIGHT	`ms1_drop_mutual`	6.00	1.40

still holds: the effect of repair is conditional on entering a sufficiently cyclic regime, and even then it does not translate into a uniform gain across datasets. The more defensible conclusion is therefore not that repair improves retrieval in general, but that it can help in some cyclic settings while remaining neutral in others.

The relationship between repair activity and retrieval gain is also informative. Structural activity, measured here by the mean feedback-arc mass removed, is positive in the `ms1` rows for all four datasets: 0.682080 for SciDocs, 1.099149 for FiQA, 0.121092 for HotpotQA, and 0.387835 for BRIGHT. Yet those structural changes do not map monotonically onto ΔnDCG. FiQA shows the largest mean removed mass but only a small and uncertain retrieval gain; SciDocs and BRIGHT both show nonzero repair activity with near-zero downstream effect; and HotpotQA shows the clearest positive ΔnDCG despite the smallest removed mass among the four `ms1` cases. This disconnect reinforces the main scientific message of the manuscript: structural simplification and retrieval improvement are related but not interchangeable outcomes.

Taken together, the repaired-versus-unrepaired comparisons support a restrained interpretation. Repair is not a universally stronger ranking strategy. Its effect depends first on whether vote construction creates enough cyclic structure for repair to become active at all, and second on whether that structural intervention happens to align with the relevance signal of the dataset. This is why the paper frames repair as a mechanism with clear structural value but only conditional retrieval benefit.

Taken together, these findings show that structural change and retrieval change can diverge and therefore should not be treated as interchangeable indicators of success.

6 Limitations

This study evaluates consistency-aware repair within a specific retrieval-oriented reranking framework rather than as a universal ranking principle. The main conclusions are therefore conditional: repair is structurally meaningful in sufficiently cyclic regimes, but its retrieval effect depends on the dataset, the preference-construction rule, and the downstream ranking rule. The results should not be read as showing that repair is uniformly beneficial across retrieval tasks, ranker families, or preference sources.

A second limitation is that the observed behavior depends strongly on how pairwise preferences are constructed. One of the central findings of the paper is that vote construction largely determines whether cyclicity appears at all, and thus whether repair becomes operative. In addition, the repair procedure used in the large-scale experiments is heuristic rather than globally optimal. The real-data pipeline uses a deterministic greedy cycle-peeling method, which is appropriate for reproducible query-level experiments but does not

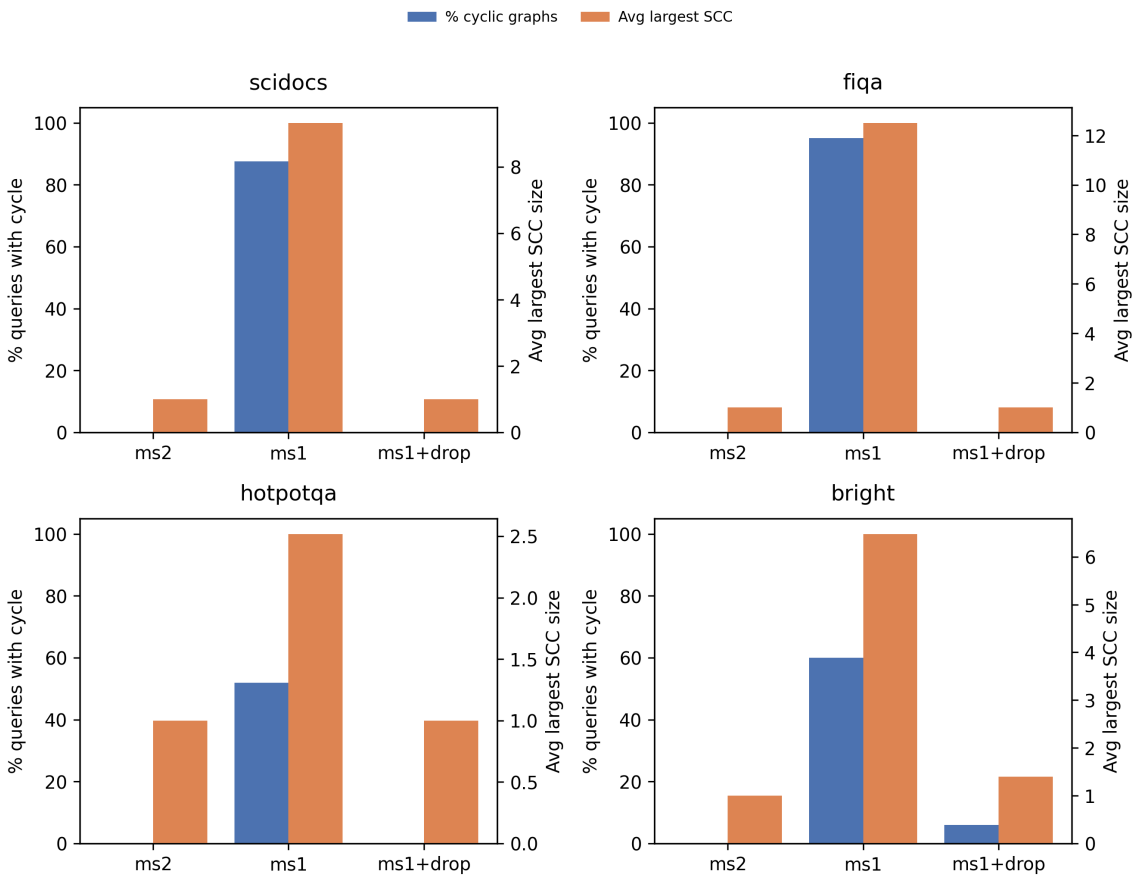


Fig. 2 Structural effect of vote construction on cyclic-query prevalence and largest strongly connected component size. `ms2` is effectively acyclic, `ms1` is strongly cyclic, and `ms1_drop_mutual` largely restores near-acyclic behavior.

imply exact minimum-feedback-arc-set optimality on all graphs.

A third limitation concerns the source of the preference evidence in the canonical package. The main experiments use score-derived pairwise votes induced from multiple retrievers rather than direct human or large-language-model pairwise judgments. This makes the study well suited for controlled analysis of preference construction and graph structure, but it also means that the strongest claims are about retrieval-oriented score aggregation rather than about all possible sources of pairwise preference data.

Finally, the real LLM evidence is narrower than the canonical four-dataset package, and the evaluation focuses on ranked effectiveness and graph structure rather than broader criteria such as stability, efficiency, calibration, or interpretability. Accordingly, the contribution of the paper is best understood as a structural and empirical analysis of when repair becomes active, what it changes in the graph, and when those changes do or do not translate into retrieval gains.

7 Conclusions and Future Research

This paper examined consistency-aware reranking through weighted preference graphs, focusing on the interaction among preference construction, graph cyclicity, repair, and retrieval effectiveness. Across the canonical four-dataset package, with narrower supporting evidence from real LLM runs, the main result is that the effect of repair is conditional rather than universal. The dominant factor is the preference-construction regime itself. Conservative construction yields graphs that are nearly always acyclic, so repaired and unrepaired

Table 7 Repair activity and repaired-versus-unrepaired Copeland ΔnDCG in the canonical four-dataset package. Mean feedback-arc mass removed summarizes structural intervention. ΔnDCG denotes repaired minus unrepaired Copeland. Rows with zero effect on all bootstrap replicates are reported as degenerate at 0.

Dataset	Variant	Mean feedback-arc mass removed	Mean ΔnDCG	95% CI
SciDocs	`ms2`	0.000000	0.000000	degenerate at 0
SciDocs	`ms1`	0.682080	-0.000127	[-0.000844, 0.000595]
SciDocs	`ms1_drop_mutual`	0.000000	0.000000	degenerate at 0
FiQA	`ms2`	0.000000	0.000000	degenerate at 0
FiQA	`ms1`	1.099149	0.001456	[-0.000515, 0.004236]
FiQA	`ms1_drop_mutual`	0.000000	0.000000	degenerate at 0
HotpotQA	`ms2`	0.000000	0.000000	degenerate at 0
HotpotQA	`ms1`	0.121092	0.016713	[0.000000, 0.040523]
HotpotQA	`ms1_drop_mutual`	0.000000	0.000000	degenerate at 0
BRIGHT	`ms2`	0.000000	0.000000	degenerate at 0
BRIGHT	`ms1`	0.387835	0.000022	[-0.001815, 0.001646]
BRIGHT	`ms1_drop_mutual`	0.008793	0.000000	degenerate at 0

pipelines coincide. More permissive construction produces markedly higher cyclicity and larger strongly connected components, creating the setting in which repair becomes structurally active.

In that active regime, repair has clear graph-level effects but mixed retrieval effects. In the canonical experiments, repair removes contradictory edge mass and simplifies the induced graph in a reproducible way, yet these structural gains do not translate into a uniform increase in nDCG. Some dataset–regime combinations show small positive changes, whereas others remain unchanged or slightly worse. The most defensible conclusion is therefore not that repair is a generally stronger reranking strategy, but that it is a meaningful structural intervention whose retrieval value depends on the dataset, the preference-construction rule, and the downstream ranking rule.

More broadly, the results suggest that preference graphs provide a useful common representation for studying reranking signals in both conventional and LLM-supported settings. They make disagreement explicit, reveal when inconsistency is actually present, and separate three design choices that are often conflated in reranking pipelines: preference construction, graph repair, and ranking extraction. This separation is one of the paper's main methodological contributions.

Several directions for future research follow from these findings. A first priority is broader real LLM evaluation across models, prompting strategies, and ranking budgets to determine how stable the observed cyclicity patterns are under direct preference elicitation. A second is to compare the current greedy repair heuristic with stronger exact or approximate feedback-arc-set methods under the same extraction framework. A third is to study preference construction more systematically, including thresholds, support rules, contradiction handling, and pair-selection policies, since these choices largely determine whether repair becomes active at all. Additional work could also examine outcomes beyond nDCG, such as stability, robustness, calibration, and interpretability, or investigate adaptive policies that use graph structure to decide whether repair should be applied.

Overall, the evidence supports a restrained conclusion. Consistency-aware repair is not a universal improvement mechanism, but it is a principled way to analyze and modify cyclic preference structure in reranking pipelines. Its main value lies in clarifying when inconsistency is present, how much structural simplification is achieved, and under what conditions that simplification may or may not improve retrieval performance.

Acknowledgements

The author used generative AI tools to assist with parts of manuscript drafting and code development. All scientific claims, experimental design, implementation decisions, verification of results,

and final manuscript content were reviewed and validated by the author.

Declarations

7.1 Funding

Not applicable.

7.2 Conflict of interest

The author declares no conflict of interest.

7.3 Data and code availability

The datasets used in this study are publicly available from their original sources and are cited in the manuscript. The code and processed artifacts used to generate the reported results are available at https://github.com/SoroushVahidi/consistency-aware-llm-rankin. Additional materials can be provided by the authors upon reasonable request.

References

[1] T.Y. Liu, *Learning to Rank for Information Retrieval* (Springer, Berlin, Heidelberg, 2011)

[2] C.J.C. Burges, From ranknet to lambdarank to lambdamart: An overview. Tech. Rep. MSR-TR-2010-82, Microsoft Research (2010)

[3] W. Sun, L. Yan, X. Ma, S. Wang, P. Ren, Z. Chen, D. Yin, Z. Ren, *Is ChatGPT Good at Search? Investigating Large Language Models as Re-Ranking Agents*, in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, ed. by H. Bouamor, J. Pino, K. Bali (Association for Computational Linguistics, Singapore, 2023), pp. 14918–14937. https://doi.org/10.18653/v1/2023.emnlp-main.923. URL https://aclanthology.org/2023.emnlp-main.923/

[4] Z. Qin, R. Jagerman, K. Hui, H. Zhuang, J. Wu, L. Yan, J. Shen, T. Liu, J. Liu, D. Metzler, X. Wang, M. Bendersky, *Large Language Models are Effective Text Rankers with Pairwise Ranking Prompting*, in *Findings of the Association for Computational Linguistics: NAACL 2024*, ed. by K. Duh, H. Gomez, S. Bethard (Association for Computational Linguistics, Mexico City, Mexico, 2024), pp. 1504–1518. https://doi.org/10.18653/v1/2024.findings-naacl.97. URL https://aclanthology.org/2024.findings-naacl.97/

[5] R. Pradeep, S. Sharifymoghaddam, J. Lin, Rankzephyr: Effective and robust zero-shot listwise reranking is a breeze! CoRR **abs/2312.02724** (2023). https://doi.org/10.48550/arXiv.2312.02724. URL https://arxiv.org/abs/2312.02724

[6] S. Zhuang, H. Zhuang, B. Koopman, G. Zuccon, *A Setwise Approach for Effective and Highly Efficient Zero-shot Ranking with Large Language Models*, in *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ed. by G.H. Yang, H. Wang, S. Han, C. Hauff, G. Zuccon, Y. Zhang (ACM, Washington DC, USA, 2024), pp. 38–47. https://doi.org/10.1145/3626772.3657813. URL https://doi.org/10.1145/3626772.3657813

[7] R. Gangi Reddy, J. Doo, Y. Xu, M.A. Sultan, D. Swain, A. Sil, H. Ji, *FIRST: Faster Improved Listwise Reranking with Single Token Decoding*, in *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, ed. by Y. Al-Onaizan, M. Bansal, Y.N. Chen (Association for Computational Linguistics, Miami, Florida, USA, 2024), pp. 8642–8652. https://doi.org/10.18653/v1/2024.emnlp-main.491. URL https://aclanthology.org/2024.emnlp-main.491/

[8] S. Yoon, E. Choi, J. Kim, H. Yun, Y. Kim, S. won Hwang, *ListT5: Listwise Reranking with Fusion-in-Decoder Improves Zero-shot Retrieval*, in *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (Association for Computational Linguistics, Bangkok, Thailand, 2024), pp. 2287–2308. https://doi.org/10.18653/v1/2024.acl-long.125. URL https://aclanthology.org/2024.acl-long.125/

[9] C. Dwork, R. Kumar, M. Naor, D. Sivakumar, *Rank Aggregation Methods for the Web*, in *Proceedings of the 10th International Conference on World Wide Web* (ACM, Hong Kong, Hong Kong, 2001), pp. 613–622. https://doi.org/10.1145/371920.372165. URL https://doi.org/10.1145/371920.372165

[10] N. Ailon, M. Charikar, A. Newman, Aggregating inconsistent information: Ranking and

clustering. Journal of the ACM **55**(5), 23:1–23:27 (2008). https://doi.org/10.1145/1411509.1411513. URL https://doi.org/10.1145/1411509.1411513

[11] S. Negahban, S. Oh, D. Shah, Rank centrality: Ranking from pairwise comparisons. Operations Research **65**(1), 266–287 (2017). https://doi.org/10.1287/opre.2016.1534. URL https://doi.org/10.1287/opre.2016.1534

[12] A. Cohan, S. Feldman, I. Beltagy, D. Downey, D. Weld, *SPECTER: Document-level Representation Learning using Citation-informed Transformers*, in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (Association for Computational Linguistics, Online, 2020), pp. 2270–2282. https://doi.org/10.18653/v1/2020.acl-main.207. URL https://aclanthology.org/2020.acl-main.207/

[13] M. Maia, S. Handschuh, A. Freitas, B. Davis, R. McDermott, M. Zarrouk, A. Balahur, *WWW'18 Open Challenge: Financial Opinion Mining and Question Answering*, in *Companion of the The Web Conference 2018 on The Web Conference 2018* (ACM, Lyon, France, 2018), pp. 1941–1942. https://doi.org/10.1145/3184558.3192301. URL https://doi.org/10.1145/3184558.3192301

[14] Z. Yang, P. Qi, S. Zhang, Y. Bengio, W. Cohen, R. Salakhutdinov, C.D. Manning, *HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering*, in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing* (Association for Computational Linguistics, Brussels, Belgium, 2018), pp. 2369–2380. https://doi.org/10.18653/v1/D18-1259. URL https://aclanthology.org/D18-1259/

[15] H. Su, H. Yen, M. Xia, W. Shi, N. Muennighoff, H.y. Wang, H. Liu, Q. Shi, Z.S. Siegel, M. Tang, R. Sun, J. Yoon, S.O. Arik, D. Chen, T. Yu, *BRIGHT: A Realistic and Challenging Benchmark for Reasoning-Intensive Retrieval*, in *International Conference on Learning Representations* (2025). URL https://openreview.net/forum?id=ykuc5q381b. Spotlight

[16] N. Craswell, B. Mitra, E. Yilmaz, D. Campos, E.M. Voorhees, *Overview of the TREC 2020 Deep Learning Track*, in *Proceedings of the Thirtieth Text REtrieval Conference (TREC 2020)* (2021)

[17] R. Nogueira, K. Cho, Passage re-ranking with BERT. arXiv preprint arXiv:1901.04085 (2019)

[18] N. Thakur, N. Reimers, A. Rücklé, A. Srivastava, I. Gurevych, *BEIR: A Heterogeneous Benchmark for Zero-shot Evaluation of Information Retrieval Models*, in *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks* (2021)

[19] C.C. Vogt, G.W. Cottrell, The use of linearly combined probability estimates as a basis for information retrieval. Information Retrieval **1**(3), 151–173 (1999). https://doi.org/10.1023/A:1009980519412

[20] G.V. Cormack, C.L.A. Clarke, S. Büttcher, *Reciprocal Rank Fusion Outperforms Condorcet and Individual Rank Learning Methods*, in *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval* (ACM, New York, NY, USA, 2009), pp. 758–759. https://doi.org/10.1145/1571941.1572114

[21] Z. Zhang, W. Zhou, J. Shi, H. Li, *Hybrid and Collaborative Passage Reranking*, in *Findings of the Association for Computational Linguistics: ACL 2023* (Association for Computational Linguistics, Toronto, Canada, 2023), pp. 14003–14021. https://doi.org/10.18653/v1/2023.findings-acl.880. URL https://aclanthology.org/2023.findings-acl.880/

[22] J. Lu, K. Hall, J. Ma, J. Ni, *HYRR: Hybrid Infused Reranking for Passage Retrieval*, in *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation* (ELRA and ICCL, Torino, Italia, 2024), pp. 8528–8534. URL https://aclanthology.org/2024.lrec-main.748/

[23] K. Wang, N. Reimers, I. Gurevych, *DAPR: A Benchmark on Document-Aware Passage Retrieval*, in *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*

(Association for Computational Linguistics, Bangkok, Thailand, 2024), pp. 4313–4330. https://doi.org/10.18653/v1/2024.acl-long.236. URL https://aclanthology.org/2024.acl-long.236/

[24] R.A. Bradley, M.E. Terry, Rank analysis of incomplete block designs: I. the method of paired comparisons. Biometrika **39**(3/4), 324–345 (1952). https://doi.org/10.2307/2334029. URL https://doi.org/10.2307/2334029

[25] D.R. Hunter, MM algorithms for generalized bradley–terry models. The Annals of Statistics **32**(1), 384–406 (2004). https://doi.org/10.1214/aos/1079120141. URL https://doi.org/10.1214/aos/1079120141

[26] R.D. Luce, *Individual Choice Behavior: A Theoretical Analysis* (Wiley, New York, 1959)

[27] R.L. Plackett, The analysis of permutations. Journal of the Royal Statistical Society: Series C (Applied Statistics) **24**(2), 193–202 (1975). https://doi.org/10.2307/2346567

[28] X. Jiang, L.H. Lim, Y. Yao, Y. Ye, Statistical ranking and combinatorial hodge theory. Mathematical Programming **127**(1), 203–244 (2011). https://doi.org/10.1007/s10107-010-0419-x. URL https://doi.org/10.1007/s10107-010-0419-x

[29] S. Vigna, Spectral ranking. Network Science **4**(4), 433–445 (2016). https://doi.org/10.1017/nws.2016.21

[30] C. Kenyon-Mathieu, W. Schudy, *How to Rank with Few Errors*, in *Proceedings of the 39th Annual ACM Symposium on Theory of Computing* (ACM, San Diego, California, USA, 2007), pp. 95–103. https://doi.org/10.1145/1250790.1250806. URL https://doi.org/10.1145/1250790.1250806

[31] R. Fagin, R. Kumar, D. Sivakumar, Comparing top k lists. SIAM Journal on Discrete Mathematics **17**(1), 134–160 (2003). https://doi.org/10.1137/S0895480102412856

[32] R. Fagin, R. Kumar, M. Mahdian, D. Sivakumar, E. Vee, *Comparing and Aggregating Rankings with Ties*, in *Proceedings of the Twenty-third ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems* (ACM, 2004), pp. 47–58. https://doi.org/10.1145/1055558.1055564

[33] K. Järvelin, J. Kekäläinen, Cumulated gain-based evaluation of ir techniques. ACM Transactions on Information Systems **20**(4), 422–446 (2002). https://doi.org/10.1145/582415.582418

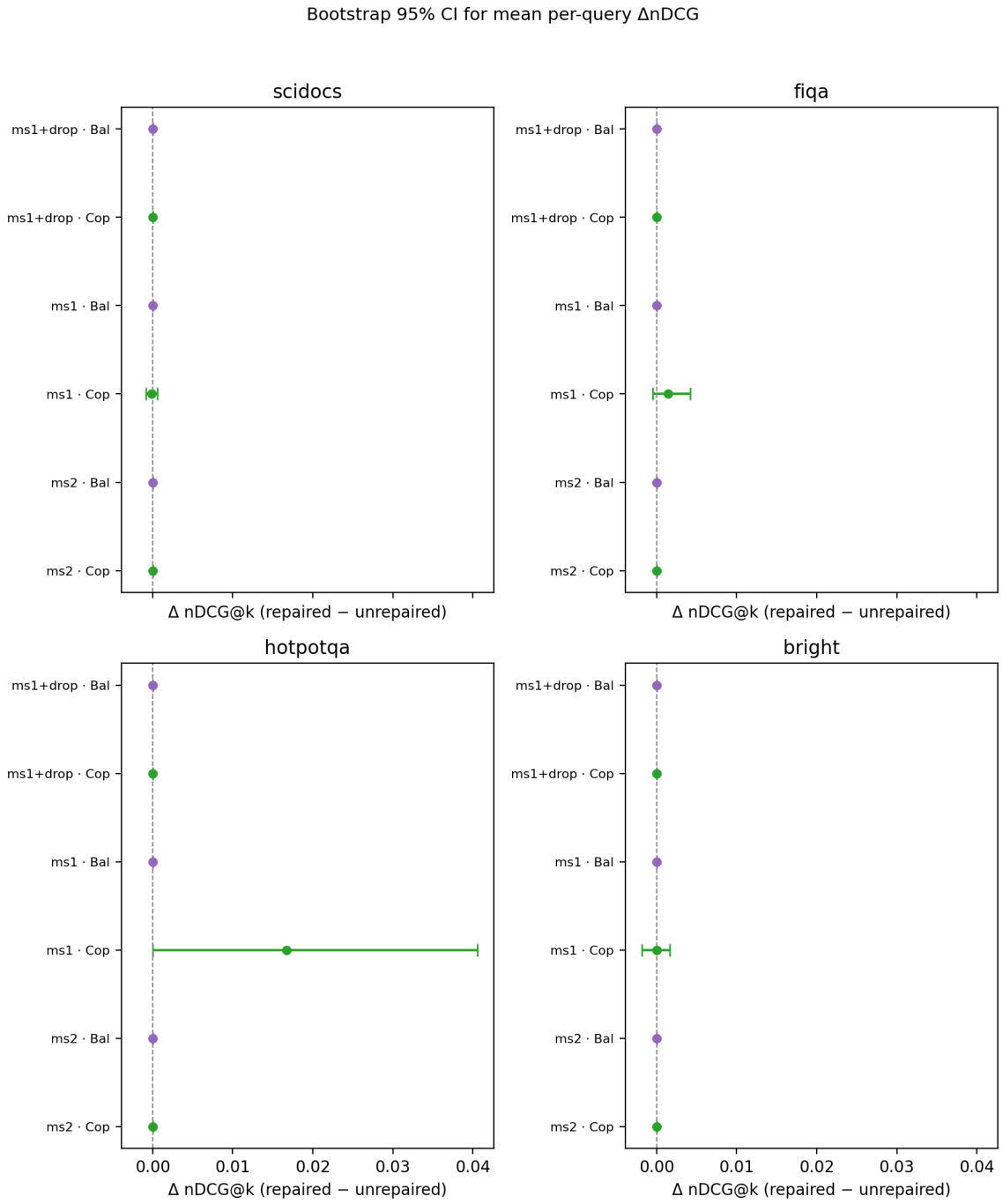


Fig. 3 Paired bootstrap summaries of repaired-minus-unrepaired Copeland ΔnDCG across datasets and vote-construction regimes. The figure highlights the same conditional pattern seen in Table 7: most near-acyclic regimes collapse to zero effect, while the structurally active `ms1` regime yields mixed, dataset-dependent retrieval changes.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- graphicalabstract.png