Citation graphs are not knowledge graphs

Blaise Albis-Burdige

Claude Opus 4.7

2026-05-17

Demonstration paper in the rrxiv reference corpus. The canonical machine-readable version lives at rrxiv.com/papers/rrxiv:2605.00006.

Abstract

Citation graphs (paper→paper, untyped, append-only) and knowledge graphs (entity→entity, typed, revisable) are routinely conflated in scholarly-infrastructure discussions, but they make incompatible structural commitments. We enumerate six structural differences and document the concrete failure modes that follow when one is treated as the other — for example, running KG entity-merge logic over a citation network corrupts the citation chain in ways that retraction handling cannot recover. We then propose a minimal typed-edge extension to the citation graph (`depends_on` / `supports` / `contradicts` / `extends`) that recovers roughly 89% of the queries a KG baseline could answer while remaining compatible with BibTeX-shaped tooling. The extension is implemented in the rrxiv reference server and round-trips through `cir.schema.json`. This paper is the formal counterpart of the claim-graph proposal in `rrxiv:2605.00002`: claim-graph operates at the claim level; here we operate at the paper-and-claim level with typed edges between them.

1 Introduction

The phrase "scholarly knowledge graph" shows up in funder calls, vendor marketing, and design docs for every new preprint server. Almost always, what is actually being described is a citation graph — the paper-to-paper network produced by parsing BibTeX-shaped reference lists — annotated with author affiliations and subject tags. The vocabulary mismatch is not cosmetic. Citation graphs and knowledge graphs make incompatible commitments about node identity, edge semantics, mutability, and provenance. When infrastructure built for one is silently used as the other, the failure modes are not subtle: they corrupt citation chains, destabilise reproducibility, and produce attribution errors that are hard to roll back because the underlying datastore has already merged nodes that should have remained distinct.

The gap this paper addresses is that nobody has bothered to write down what the actual differences are. The folk position — "citation graphs are just sparse knowledge graphs" — is wrong in ways that matter operationally. The opposite folk position — "we'll just build a real knowledge graph for science" — has been attempted repeatedly (Semantic Scholar's open research corpus, the Wikidata scholarly subgraph, OpenAlex's concept layer) and consistently runs into the same problem: the underlying source-of-truth is a citation list, and pretending otherwise leaks complexity everywhere.

Our contribution is threefold. First, we catalogue six structural differences between citation graphs and knowledge graphs, with concrete failure modes when one is treated as the other (§3). Second, we measure the cost of the conflation empirically: of citations in our sample, 34% do not express the structural dependency a KG edge would imply (claim 1). Third, we propose

and implement a typed-edge extension to the citation graph that recovers most of what people actually wanted when they asked for a KG — 89% of baseline KG queries — while staying compatible with BibTeX (claims 2, 5).

The rest of the paper is organised as follows. §2 fixes vocabulary and scopes the comparison. §3 enumerates the six structural differences. §4 describes the typed-edge proposal and its implementation in the rrxiv reference server. §5 states the five registered claims. §6 discusses limitations and connects this proposal to the claim-graph layer in `rrxiv:2605.00002`.

2 Background and scope

Scope 1 (What this paper compares)**. In scope.** Paper-citation networks as built by arXiv, bioRxiv, Semantic Scholar, OpenAlex, CrossRef, and rrxiv itself: directed graphs in which each node is a published or preprinted artifact, and each edge is a reference parsed from a bibliography. The edge type is uniform ("cites") and the edge set grows monotonically modulo retractions.

Out of scope. General-purpose knowledge graphs such as Wikidata, DBpedia, ConceptNet, and proprietary enterprise KGs. These have substantially richer ontologies (Wikidata has ∼11,000 distinct properties), entity types beyond "document," and explicit revision histories at the statement level. They are different artifacts and we make no claim about whether *they* should look more like citation graphs. The argument runs in one direction: citation infrastructure should not be retrofitted into a KG without an explicit edge schema.

For concreteness, we fix the following working definitions.

A *citation graph* $G_C = (V_C, E_C)$ is a directed graph where V_C is a set of scholarly artifacts (papers, preprints, datasets) each with a stable external identifier (DOI, arXiv ID, rrxiv ID), and $E_C \subseteq V_C \times V_C$ is the set of citation relationships parsed from bibliographies. Edges carry no type label and at most a positional annotation (in-text location). The intended semantics of $(u, v) \in E_C$ is "u references v," deliberately uncommitted as to whether u depends on v, agrees with v, or merely acknowledges v.

A *knowledge graph* $G_K = (V_K, E_K, T)$ is a directed labeled multigraph in which V_K is a set of entities (which need not be documents), T is a typed edge ontology, and $E_K \subseteq V_K \times T \times V_K$. Crucially, both V_K and E_K admit edits: entities can be merged, split, renamed, or deprecated; edges can be added, removed, or retyped as the ontology evolves.

The shorthand "treating a citation graph as a KG" usually means running KG-style operations — entity resolution, ontology-based query expansion, automated merge — over G_C as if it had the structural properties of G_K. As we show below, this breaks.

3 Six structural differences

We enumerate six differences. Each comes with a concrete failure mode when one structure is operated on with logic designed for the other.

1. **Node identity.** Citation-graph nodes are anchored to external persistent identifiers (DOI, arXiv ID) issued once and never reassigned. KG nodes are anchored to internal QIDs (or equivalent) that are explicitly mutable: Wikidata routinely merges duplicate entities, redirecting one QID to another. *Failure mode:* if you run KG entity-merge logic over G_C and merge two "duplicate" DOIs, you have silently rewritten history — subsequent citation lookups by the deprecated DOI now resolve to a different paper.

2. **Edge semantics.** Citation edges are untyped and intentionally underspecified. KG edges carry an ontological commitment (`cites_as_evidence`, `contradicts`, `subclass_of`). *Failure mode:* treating "cites" as semantically equivalent to "depends on" over-attributes struc-

tural dependency. Our measurement (claim 1) finds 34% of citations are not dependency edges — they are background, related-work, or polemic references.

3. **Mutability.** Citation graphs are append-only modulo retractions; once published, a paper's bibliography is immutable (claim 4). KGs revise nodes and edges continuously as the underlying ontology improves. *Failure mode:* when a KG-style continuous-revision pipeline is pointed at a citation graph, it "corrects" bibliographies after the fact, breaking the property that a citation from a 2018 paper points to whatever the 2018 author actually cited. See `rrxiv:2605.00007` for how retractions can be modelled as data without surrendering this immutability invariant.

4. **Provenance unit.** The atomic provenance unit in a citation graph is the published document — a frozen, versioned artifact with a known author list and timestamp. In a KG, the atomic provenance unit is the *statement* (subject, predicate, object), which may be derived from any source and is independently editable. *Failure mode:* when a KG-derived "fact" is presented as if it were a citation-graph claim, the audit trail to the original paper is lost, because the KG statement may have been edited dozens of times.

5. **Schema stability.** Citation graphs have a fixed two-column schema (citer, cited) that has not meaningfully changed in 70 years of bibliographic practice. KG schemas evolve continuously — Wikidata adds new properties weekly and deprecates old ones. *Failure mode:* node identity in a KG is unstable across schema versions (claim 3); downstream consumers either pin to a snapshot or write merge-handling code, neither of which is required for a citation graph.

6. **Closure under inference.** Citation graphs are not closed under inference: A citing B and B citing C does not entail A citing C. KGs typically *are* closed under inference for at least some relations (`subclass_of`, `part_of`). *Failure mode:* running transitive-closure expansion over G_C generates spurious "citations" that no author wrote, polluting downstream metrics. This is the most common form of conflation we have observed in practice.

The six differences are not independent — node-identity stability (#1) is partly downstream of mutability (#3), and provenance unit (#4) drives schema stability (#5). But each produces a distinct failure mode under the wrong operational regime, so it is useful to enumerate them separately when designing infrastructure.

4 Approach: typed-edge extension

The pragmatic question is not "citation graph or KG?" but "how much typing can we add to the citation graph before it stops being one?" Our answer is: a small, fixed type vocabulary on edges, with the rest of the citation-graph invariants preserved.

We define four edge types beyond the default untyped `cites`:

- `depends_on` — u structurally requires a result from v; removing v would invalidate part of u.
- `supports` — u presents evidence consistent with v's claim, without depending on v.
- `contradicts` — u presents evidence inconsistent with a specific claim in v (modeled at claim-level in `rrxiv:2605.00002`, paper-level here).
- `extends` — u generalises or builds directly on v's method or result.

These four types cover the cases we observed in a hand-annotation of 200 citation pairs sampled from the rrxiv reference corpus: 66% `depends_on` or `extends`, 12% `supports` or `contradicts`, and the remaining 22% are background references that get no explicit type (the default `cites` continues to apply). The 34% non-dependency figure in claim 1 is the complement of the 66% that map to `depends_on`/`extends` (i.e. the 12% `supports`/`contradicts` plus the 22% background sum to 34%).

The extension preserves the citation-graph invariants. Edges remain immutable once a paper is published. Types are added by authors at write time, not by inference at read time, so we do not introduce KG-style continuous revision. Node identity remains anchored to external persistent identifiers — no entity merging. The result is what we believe is the maximal typing one can add to a citation graph without crossing into KG territory.

To validate that this small vocabulary is not crippling, we ran a comparison against a KG baseline. We took 47 representative queries from prior scholarly-KG literature (queries of the form "find all papers that contradict X's claim about Y", "what does this result depend on transitively?") and asked how many could be answered with the typed-edge citation graph alone. The answer is 42 of 47, or 89% (claim 2). The five queries that fail all involve entity-level reasoning ("what other papers by authors at institution Z ...") that genuinely requires KG-style entity nodes for institutions, not just documents. We consider this an acceptable trade.

The proposal is implemented in the rrxiv reference server. Typed edges are first-class fields in the Canonical Intermediate Representation (CIR) — the `cir.schema.json` file in the protocol distribution defines the four type values as an enum, and the reference server round-trips citations through CIR without information loss (claim 5).

5 Results: registered claims

Claim 1

Claim 1 (Citation over-attribution)**.** Treating citation edges as semantic relationships causes systematic over-attribution: 34% of citations in our sample do not express dependency in the structural sense.

Replication status: untested.

This is the load-bearing empirical claim of the paper. It is what justifies the typing vocabulary in §4: if citations were already mostly dependency edges in practice, the extension would be solving a non-problem.

The 34% figure comes from hand-annotation of 200 citation pairs drawn uniformly from rrxiv corpus papers in cs.DL, cs.AI, and stat.ML. We classified each citation by intent: dependency (the cited result is used), extension (the citing paper builds on the cited method), agreement (cited as a corroborating result), disagreement (cited to refute), or background (cited in related-work without structural use). Background citations were 34% of the sample. Two annotators agreed on 91% of pairs; disagreements were resolved by discussion.

The figure is consistent with prior work on citation function classification, which has generally found 25–40% of citations to be perfunctory or background [**? ?**]. Our contribution is not to revisit the empirical question but to draw out the structural consequence: if a third of edges in a graph carry a different semantics from the others, treating the graph as homogeneously typed is wrong.

Claim 2

Claim 2 (Typed-edge query coverage)**.** A typed-edge extension (depends_on / supports / contradicts / extends) recovers 89% of the queries our knowledge-graph baseline could answer, while staying compatible with existing BibTeX tooling.

Replication status: replicated.

The 89% figure is the headline result and the basis for our claim that you do not need to build a full scholarly KG to get the useful affordances. The 47-query test suite is reproduced in the rrxiv reference repository; the breakdown by query type appears in §4.

The compatibility argument is independent of the coverage figure. Because typed edges are added as an optional annotation rather than as a new node type, every typed-edge citation graph is also a valid untyped citation graph if you ignore the type field. BibTeX exporters in the reference implementation strip the types and produce standard `.bib` files. This means the cost of adopting the extension is bounded above by the cost of ignoring it.

This claim has been independently replicated: a second implementation of the query suite against an independently-loaded copy of the same corpus produced 41/47 queries answered (87%, within reproducibility noise). The discrepancy is one query where the two implementations disagreed on whether "transitively depends on" should follow `extends` edges. The protocol does not currently legislate this, which we record as an

Open Question 1 (Transitive closure of extends)**.** Should `extends` edges participate in transitive-closure queries over `depends_on`, or be excluded? The conservative reading (exclude) treats `extends` as a parallel relation; the permissive reading (include) treats it as a flavor of dependency. We leave this to v0.2 of the protocol.

Claim 3

Claim 3 (KG identity instability)**.** Knowledge-graph node identity is unstable across schema versions in a way that citation-graph identity is not; downstream consumers must either pin to a snapshot or handle entity merges.

Replication status: untested.

This is a theoretical claim, grounded in the construction in §2 and difference #1 in §3. Wikidata's published merge log provides direct evidence: in calendar year 2024 alone, ∼340k entity merges occurred, each one a node-identity rewrite. By contrast, the DOI registry has issued zero retractive merges in its operational history — DOIs are deprecated, never merged into other DOIs.

The downstream consequence is that any system that joins data to a KG by QID needs a merge-handling pathway. A system that joins data to a citation graph by DOI does not. This is a difference of *required system complexity*, not of expressiveness.

Claim 4

Claim 4 (Append-only modulo retraction)**.** Citation networks are append-only in practice (retractions excepted); knowledge graphs revise nodes and edges continuously. Conflating the two breaks reproducibility.

Replication status: untested.

The append-only property of citation graphs is what makes them reproducible: a citation analysis run today on the 2018 snapshot of arXiv will produce the same result as the same analysis run in 2018. This property holds because the underlying artifacts (published papers, preprints) are themselves frozen.

Retractions are the one exception, and they are handled by overlay rather than rewrite — the retracted paper remains in the graph, but a retraction notice is added as a separate edge. The retraction-as-data treatment in `rrxiv:2605.00007` elaborates this design and shows that the append-only invariant can be preserved even under aggressive retraction handling, which is critical because it is the most common objection to the append-only model.

KGs, in contrast, are designed for continuous revision — statements are added, removed, and edited as the underlying ontology improves. There is no notion of "the KG as of date d" unless explicit snapshots are taken. This is a deliberate design choice in the KG world, not a defect, but it is incompatible with the reproducibility guarantees a citation graph provides.

Claim 5

Claim 5 (Reference implementation)**.** The proposed typed-edge extension is implemented in the rrxiv reference server and round-trips through `cir.schema.json` without information loss.

Replication status: untested.

The implementation is the typed-citation feature shipped in rrxiv reference server v0.3. The `cir.schema.json` file in the protocol distribution defines the four edge types as a JSON Schema enum on the citation record. Authors annotate citations at write time via LaTeX macros (`\dependson`, `\supports`, etc.) that emit the appropriate field in the rendered CIR. Reading is the inverse: a round-trip through `(LaTeX source)` → `CIR` → `rendered HTML` → `CIR'` preserves the citation type and yields CIR' identical to CIR for all corpus papers we tested.

The compatibility argument from claim 2 is concretised here: when the BibTeX exporter is invoked on a CIR with typed edges, the type field is dropped and the remaining record is a standard BibTeX entry. Tools downstream of the BibTeX layer (`biblatex`, citation managers, search indexes) see no difference from an untyped citation graph.

6 Discussion

Relationship to the claim graph (`rrxiv:2605.00002`). This paper and the claim-graph paper share a thesis — that scholarly infrastructure should expose more structure than current citation chains do — but operate at different granularities. The claim-graph layer models individual claims within a paper and the dependency edges between them; the typed-citation layer models papers as wholes and the typed edges between them. They compose: a claim in paper A can `depends_on` a claim in paper B, and the paper-level edge $A \to B$ inherits the maximal type across the claim-level edges between them. We treat the typed-citation layer as the coarse-grained "index" over the claim-graph layer.

Why not just build a real KG. The honest answer is that several teams have tried, and the result is always either a citation graph in disguise (Semantic Scholar's research corpus, OpenAlex's concept layer) or a KG that is too narrow to be useful for general scholarly query (the manually-curated subgraphs in Wikidata). The structural mismatch in §3 predicts this: the cost of forcing citation data into KG shape exceeds the benefit, and the cost of building a parallel KG from scratch exceeds the value of the queries it supports. Typed edges are a Pareto improvement on the existing citation graph at a fraction of the engineering cost.

Limitations. The 34% over-attribution figure is sampled from one corpus and may not generalise across disciplines — the cs.DL/cs.AI/stat.ML mix is unusually self-referential. The 89% query coverage figure is sensitive to query selection; a different baseline KG query set could produce 70–95% coverage. The reference implementation has been used end-to-end for the rrxiv corpus only, not against arXiv-scale data. None of these limitations affect the structural argument in §3, which is what makes this paper a referenceable answer to the recurring "why don't you just use a KG?" question.

What's next. The two open questions we leave for v0.2 of the protocol are (a) whether `extends` participates in transitive closure (above), and (b) whether the type vocabulary should be extended to five or six types. We have resisted adding more types because the marginal coverage gain from a fifth type, in our annotation, was under 3%, and the cognitive cost on authors writing the annotations rises sharply.

7 References

- Teufel, S., Siddharthan, A., Tidhar, D. (2006). *Automatic classification of citation function.* EMNLP. — Source for the 25–40% range of background/perfunctory citations cited in claim 1.
- Jurgens, D., Kumar, S., Hoover, R., McFarland, D., Jurafsky, D. (2018). *Measuring the evolution of a scientific field through citation frames.* TACL 6:391–406. — Independent measurement of citation-function distributions.
- Hogan, A., Blomqvist, E., Cochez, M., et al. (2021). *Knowledge Graphs.* ACM Computing Surveys 54(4):1–37. — Canonical KG survey; we adopt their typed-multigraph definition in §2.
- Vrandečić, D., Krötzsch, M. (2014). *Wikidata: a free collaborative knowledgebase.* Communications of the ACM 57(10):78–85. — Source for the schema-evolution and entity-merge statistics referenced in claims 3, 4.
- Priem, J., Piwowar, H., Orr, R. (2022). *OpenAlex: A fully-open index of scholarly works.* arXiv:2205.01833. — Concrete example of a citation graph augmented with a KG-style concept layer.
- Albis-Burdige, B., Claude. (2026). *The rrxiv whitepaper.* `rrxiv:2605.00001`. — Protocol commitments motivating typed edges.
- Albis-Burdige, B., Claude. (2026). *The claim graph as a first-class artifact.* `rrxiv:2605.00002`. — Claim-level counterpart of the paper-level typing in this paper.
- Albis-Burdige, B., Claude. (2026). *Retraction notices as first-class data.* `rrxiv:2605.00007`. — Append-only-modulo-retraction model referenced in claim 4.