

Proof Identity and Divisible Correspondence in Multi-Derivation Reasoning Evaluation

Toprak Seda Karaosmanoğlu
Middle East Technical University
Ankara, Türkiye
toprak.karaosmanoglu@metu.edu.tr

Abstract

Systems that return several valid derivations require explicit choices about proof identity and proof-set correspondence. This study examines both choices on ProofWriter. An application-aware representation distinguishes grounded uses of the same source rule through canonical premise derivations. Source-ID projection collapses distinct reference proofs in 28 of 400 stratified derivable questions (7.0%). The comparison includes exact proof-set F_1 , flattened-atom F_1 , Optimal Subpattern Assignment with Jaccard distance (Jaccard-OSPA), and MCT-J. The label MCT-J denotes a balanced optimal-transport distance with Jaccard distance as the ground metric. On 160 fixed outputs from 32 derivable questions, source-ID projection lowers MCT-J distance by 0.0606 on average. A cluster bootstrap over the underlying ProofWriter worlds gives a 95% interval of 0.0316–0.0955. MCT-J equals Jaccard-OSPA on all 117 equal-cardinality outputs, while Jaccard-OSPA has the larger distance on 38 of 43 unequal-cardinality outputs. Of 433 generated proof members, 430 pass independent rule-level replay. Excluding the three replay failures changes the mean representation effect from 0.0606 to 0.0589. Proof identity and proof-set correspondence therefore change structural evaluation on fixed outputs.

1 Introduction

Proof-producing systems expose derivations together with final answers. PProver and ProofWriter generate proof graphs (Saha et al., 2020; Tafjord et al., 2021), and multiPProver generates several alternative proofs for one answer (Saha et al., 2021). Structured evaluation must therefore specify when two derivations count as the same proof and how reference and predicted proof sets are compared.

Proof identity matters when one source rule is applied more than once. Two grounded applications can cite the same rule statement and depend on different premise derivations. A representation based

on the source identifier (source ID) maps those applications to the same rule node. This changes the represented proof before a proof-set distance is computed. The application-aware representation keeps the applications distinct through canonical premise derivations.

Proof-set correspondence matters when several proofs are present. Exact recovery scores shared proof members. Optimal Subpattern Assignment (OSPA) uses graded one-to-one matching with a cardinality penalty (Schuhmacher et al., 2008). Balanced optimal transport permits normalized proof mass to divide across several counterparts (Peyré and Cuturi, 2019). The paper uses the label MCT-J for the resulting balanced optimal-transport distance with Jaccard distance as the ground metric. One-to-one and divisible correspondence agree at equal cardinality and can separate when cardinalities differ.

ProofWriter offers a controlled setting because provable examples contain the complete proof set under the generated Datalog semantics (Tafjord et al., 2021). Under the open-world assumption (OWA), the depth-5 test split contains 11,376 derivable questions, including 2,456 with multiple reference proofs. The stratified reference analysis finds source-ID proof-set collapse in 28 of 400 derivable questions. On 160 fixed derivable outputs from 32 questions, the mean MCT-J difference between application-aware and source-ID scoring is 0.0606. A cluster bootstrap over the underlying ProofWriter worlds gives a 95% interval of 0.0316–0.0955.

The paper develops a canonical grounded-application representation and tests its consequences in a stratified reference analysis that distinguishes repeated source-rule use from observed proof-set collapse. It then formalizes divisible proof-set correspondence against one-to-one Jaccard-OSPA and measures the effect by rescored the same outputs under both representations. Deterministic rule replay and sensitivity analyses

test the result across replay filtering and the stated encoding choices.

2 Related Work

Proof generation and reasoning-chain evaluation. RuleTaker studies reasoning over natural-language facts and rules (Clark et al., 2020). PProver adds proof generation (Saha et al., 2020), ProofWriter supplies generated implications with proofs (Taffjord et al., 2021), and multiPProver generates multiple proof graphs (Saha et al., 2021). EntailmentBank represents multistep explanations as entailment trees (Dalvi et al., 2021). Recent studies evaluate reasoning steps and reasoning-chain verifiers (Prasad et al., 2023; Jacovi et al., 2024). Other work studies proof-tree generation (Gupta, 2026) and formal logical reasoning (López-Ponce and Bel-Enguix, 2025). Adjacent settings include tool-integrated mathematical reasoning (Obozov et al., 2025) and autoformalization evaluation (Poiroux et al., 2025). The present study focuses on evaluation when one answer has several explicit derivations.

Structured and evidence-set evaluation. Structured metrics can compare outputs through matching of common substructures (Chen et al., 2023). Graph evaluation includes many-to-many alignment (Opitz et al., 2021). Fact-checking research also evaluates retrieved evidence and evidence quality (Thorne et al., 2018; Schlichtkrull et al., 2024; Akhtar et al., 2026). The evaluated object is a set of derivation proofs whose internal rule applications also require an identity convention.

Finite-set assignment and transport. Optimal Subpattern Assignment compares finite sets by one-to-one assignment and a cardinality term (Schuhmacher et al., 2008). Optimal transport compares discrete measures through marginal-constrained couplings (Peyré and Cuturi, 2019). Structured optimal transport extends this framework to structured data (Alvarez-Melis et al., 2018). Jaccard distance is fixed as the proof-level ground cost so that Jaccard-OSPA and MCT-J differ through their cross-set correspondence.

3 Task and Proof Representation

3.1 Benchmark object

A ProofWriter instance contains natural-language facts and rules, a question, a truth status in $\{\text{true}, \text{false}, \text{unknown}\}$, and the complete proof set

for each provable answer under its generated Data-log theory (Taffjord et al., 2021). A proof is a rooted derivation directed acyclic graph (DAG) whose fact steps feed rule-application steps and whose root supports the queried status.

For a derivable question q , let

$$G_q = \{g_1, \dots, g_m\}, \quad P_q = \{p_1, \dots, p_n\}$$

denote the sets of distinct reference and predicted proofs. Exact duplicate predicted proofs collapse to one member before proof-set scoring.

3.2 Grounded application identity

Source identity records the rule statement used by a derivation step. Grounded application identity incorporates the premise derivations into the step identity. Canonical identities are defined recursively. A fact step with source fact f receives

$$\sigma(\text{fact}) = (\text{fact}, f). \quad (1)$$

For an application of source rule r to canonical premise derivations $\sigma_1, \dots, \sigma_k$,

$$\sigma(\text{apply}) = (\text{apply}, r, \text{sort}(\sigma_1, \dots, \sigma_k)). \quad (2)$$

ProofWriter rules use conjunction-style antecedents, so the canonical premise collection is order-invariant. The paper refers to this construction as the *application-aware representation*. Distinct grounded applications of one source rule receive distinct identities whenever their canonical premise derivations differ.

Proposition 1 (Step-name invariance). *Renaming local step identifiers while preserving the rooted derivation DAG, source IDs, and premise relations leaves the application-aware canonical proof unchanged.*

Proof. Consider a topological ordering of the DAG. A fact signature depends only on its source fact. For a rule-application step, the induction hypothesis fixes every premise signature. Eq. (2) then depends only on the source rule and the resulting canonical premise signatures. The root signature and all structural atoms derived from these signatures remain unchanged.

3.3 Source-ID projection

The source-ID representation identifies facts by source fact ID and rule applications by source rule ID. Edges connect these source-level nodes. Repeated grounded applications can therefore map to

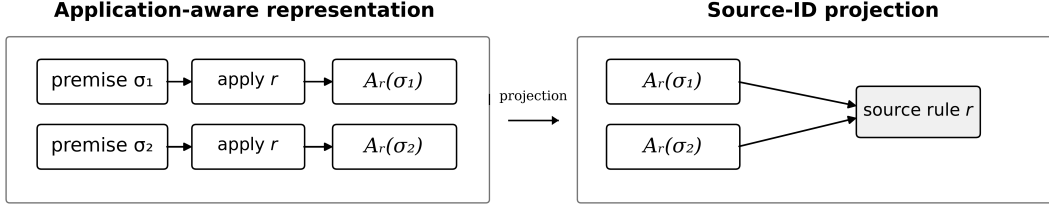


Figure 1: Grounded rule-application identity and source-ID projection. The application-aware representation distinguishes applications of source rule r through their canonical premise derivations σ_1 and σ_2 . Here $A_r(\sigma)$ denotes the grounded application identity of source rule r with canonical premise derivation σ . Source-ID projection maps both application identities to the same source-rule node.

one rule node, as illustrated in Figure 1. Generation uses the application-aware output structure. Source-ID projection is applied later when the same generated DAGs are rescored.

The stratified reference analysis contains direct many-to-one projection cases. One question has four distinct application-aware reference proofs and three distinct source-ID proofs after projection and deduplication. Section 6.1 reports the prevalence of this behavior.

4 Comparing Alternative Proof Sets

The application-aware encoding uses node atoms for canonical step signatures, source-label atoms for source IDs, directed premise-to-application edge atoms, and one root atom. The primary source-ID encoding uses source-ID node atoms, directed edges, and one root atom. This design omits separate source-label atoms from the source-ID representation. Fact identities are already carried by fact-node identifiers in both encodings. Section 6.5 tests a matched atom inventory by adding source-label atoms to the source-ID representation. Table 1 summarizes the four structural measures.

4.1 Exact and flattened comparators

Exact proof-set F_1 treats G and P as sets of canonical proofs:

$$F_1^{\text{exact}}(G, P) = \frac{2|G \cap P|}{|G| + |P|}. \quad (3)$$

When both proof sets are empty, exact F_1 is 1. When exactly one proof set is empty, exact F_1 is 0. Flattened-atom F_1 first unions structural atoms across all proofs and then computes set F_1 . This comparator measures structural overlap after proof grouping has been removed. Exact duplicates collapse under the same set policy before all proof-set measures are computed.

4.2 Jaccard ground distance and OSPA

For proof atom sets g and p ,

$$d_J(g, p) = 1 - \frac{|g \cap p|}{|g \cup p|}. \quad (4)$$

Every represented proof contains a root atom, so the atom sets used by d_J are nonempty. OSPA is instantiated with order parameter 1, cutoff $c = 1$, and d_J as the ground distance (Schuhmacher et al., 2008). Since $d_J \leq 1$, the cutoff is inactive. For $m \leq n$,

$$d_{\text{OSPA}}(G, P) = \frac{1}{n} \left[\min_{\tau \in \mathcal{I}_{m,n}} \sum_{i=1}^m d_J(g_i, p_{\tau(i)}) + (n - m) \right]. \quad (5)$$

where $\mathcal{I}_{m,n}$ is the set of injections from $\{1, \dots, m\}$ into $\{1, \dots, n\}$. The case $m > n$ uses the symmetric expression. This study refers to the resulting instantiation as Jaccard-OSPA.

4.3 Balanced transport over proof sets

For proof sets with at least one member, uniform empirical measures assign equal mass to each distinct proof:

$$\mu_G = \frac{1}{m} \sum_{i=1}^m \delta_{g_i}, \quad \mu_P = \frac{1}{n} \sum_{j=1}^n \delta_{p_j}. \quad (6)$$

Let $\mathbf{a} = (1/m, \dots, 1/m)$ and $\mathbf{b} = (1/n, \dots, 1/n)$. The balanced transport polytope is

$$\Pi(\mathbf{a}, \mathbf{b}) = \left\{ \pi \in \mathbb{R}_+^{m \times n} \left| \begin{array}{l} \pi \mathbf{1}_n = \mathbf{a}, \\ \pi^\top \mathbf{1}_m = \mathbf{b} \end{array} \right. \right\}. \quad (7)$$

The transport distance is

$$d_{\text{MCT-J}}(G, P) = \min_{\pi \in \Pi(\mathbf{a}, \mathbf{b})} \sum_{i=1}^m \sum_{j=1}^n \pi_{ij} d_J(g_i, p_j). \quad (8)$$

Measure	Proof-level comparison	Cross-set correspondence	Cardinality treatment	Role in the analysis
Exact proof-set F_1	Exact canonical proof identity	Exact member overlap	Set sizes enter the F_1 denominator	Strict proof-set recovery
Flattened-atom F_1	Exact structural-atom identity	Union of atoms across proofs	Proof grouping and proof-set cardinality are removed by flattening	Control for loss of proof grouping
Jaccard-OSPA	Jaccard distance between proofs	Optimal one-to-one assignment	Unmatched-member penalty with order 1, cutoff 1	Graded finite-set comparator
MCT-J	Jaccard distance between proofs	Balanced divisible coupling	Uniform marginal mass over all reference and predicted proofs	Graded comparison with divisible matching

Table 1: Structural measures used in the study. Jaccard-OSPA and MCT-J share the same proof-level ground distance and differ in cross-set correspondence.

The label MCT-J denotes this balanced optimal-transport distance with Jaccard distance as the ground metric. It is the standard Wasserstein-1 construction (Peyré and Cuturi, 2019). When both proof sets are empty, the distance is 0. When exactly one proof set is empty, the distance is 1. After duplicate collapse, MCT-J is a metric on finite represented proof sets. Jaccard distance is a metric on finite atom sets, and Wasserstein-1 inherits metricity from its ground distance. Duplicate collapse makes the represented proof-set map injective. Nonempty-set distances are bounded by 1, so the empty-set extension preserves the triangle inequality. The resulting distance is permutation-invariant and bounded in $[0, 1]$. Uniform weighting assigns mass $1/m$ to each reference proof and $1/n$ to each predicted proof. Proof-set size therefore enters the transport geometry through the marginals. ProofWriter leaves proof importance unspecified. The primary analysis uses uniform proof mass, and Section 6.5 tests size-weighted mass.

4.4 Relation to one-to-one assignment

The common Jaccard ground distance isolates the cross-set correspondence difference between Jaccard-OSPA and MCT-J.

Proposition 2 (Equal-cardinality equivalence). *For $|G| = |P| = k > 0$ with uniform proof weights,*

$$d_{\text{MCT-J}}(G, P) = d_{\text{OSPA}}(G, P).$$

Proof. Multiplying every feasible transport entry by k yields a doubly stochastic matrix. A linear objective attains an optimum at an extreme point of this polytope. The Birkhoff–von Neumann theorem identifies these extreme points with permu-

tation matrices. Scaling by $1/k$ yields the minimum average one-to-one Jaccard assignment cost. Jaccard-OSPA has the same objective at equal cardinality because its cardinality term equals zero.

Proposition 3 (Jaccard-OSPA upper bound). *For finite proof sets with at least one member under the definitions above,*

$$d_{\text{MCT-J}}(G, P) \leq d_{\text{OSPA}}(G, P).$$

Proof. Assume $m \leq n$. Symmetry gives the corresponding construction for $m > n$. Take an OSPA-optimal injection from the m reference proofs to m predicted proofs and send mass $1/n$ along every assigned pair. Each reference row retains mass $1/m - 1/n$, with total residual mass $(n - m)/n$. Route this residual mass to the $n - m$ unmatched prediction columns while satisfying the uniform marginals. Since $d_J \leq 1$, residual transport costs at most $(n - m)/n$. Assigned transport costs $(1/n) \sum_i d_J(g_i, p_{\tau(i)})$. Their sum is bounded by the OSPA objective, and the optimal balanced transport cost is bounded by this feasible plan.

Proposition 3 makes unequal proof-set cardinality a requirement for a positive OSPA-minus-MCT-J gap. Some unequal-cardinality pairs still yield equality. A merge example illustrates the separation: if G contains k distinct singleton proof-atom sets and P contains their union as one proof, then

$$d_{\text{MCT-J}} = 1 - \frac{1}{k}, \quad d_{\text{OSPA}} = 1 - \frac{1}{k^2}.$$

The difference $1/k - 1/k^2$ arises from correspondence and cardinality treatment under the same Jaccard ground distance.

5 Data and Evaluation

5.1 Reference data and stratified reference analysis

The study uses the depth-5 test split of ProofWriter OWA V2020.12.3. The test split contains 20,030 questions: 5,688 true, 5,688 false, and 8,654 unknown. Among 11,376 derivable questions, 2,456 have at least two reference proofs (21.59%).

Source-ID representation loss is analyzed on 400 stratified derivable test questions: 80 with one reference proof, 120 with two, 120 with three or four, and 80 with five to eight. These rates describe the stratified reference analysis, while full-test counts provide corpus prevalence.

A high-multiplicity evaluation set extends the stratified reference analysis beyond eight proofs. It contains 40 questions with 17–32 reference proofs and mean proof-set size 21.48. The set is deliberately enriched for questions with large reference proof sets.

5.2 Generation sample

The generation sample contains 40 questions, with eight questions from each proof-set-size stratum 0, 1, 2, 3–4, and 5–8. The 400-question stratified reference sample, the 40-question generation sample, and the 40-question high-multiplicity set were selected deterministically before model generation and are pairwise disjoint. The 32 derivable generation questions span 30 source worlds, where a source world is an underlying generated fact-and-rule theory shared by one or more questions. Five presentation orders per question produce 200 outputs in total, of which 160 correspond to derivable questions. Each presentation deterministically permutes the supplied facts and rules while preserving source IDs and natural-language content.

The public input contains the facts and rules, the question, neutral source IDs, and task instructions. Reference answers, proof annotations, and canonical signatures are used only for evaluation. The task instruction allows at most eight returned derivations in every generation call. Every derivable question in the generation sample has at most eight reference proofs. Returned proofs use a rooted DAG encoded in JavaScript Object Notation (JSON). Generation was run on 8 August 2026 through OpenAI using GPT-5.6 Sol (OpenAI, 2026). Requests used the model identifier gpt-5.6-sol with medium reasoning effort. A strict JSON schema and an 8,192-token output budget constrained generation.

Exact proof-set recovery is reported as a descriptive comparator. The primary comparison is the score change produced by applying two representations and two correspondence rules to the same fixed outputs.

Generated DAGs are checked for valid source IDs, step references, roots, cycles, and orphan steps, meaning steps disconnected from the returned root. Exact canonical matches identify ProofWriter reference proofs. Other structurally valid DAGs receive graded structural distances. A separate deterministic rule-replay procedure executes cited rules from the literals established by their premise steps and checks the supported root.

5.3 Paired comparison and uncertainty

Representation effects are paired at the generated-output level. The primary score difference is the mean difference between application-aware and source-ID scoring across 160 derivable outputs. For uncertainty, the 30 source worlds are sampled with replacement as clusters. Each sampled world contributes all associated questions and presentations, preserving within-world dependence and observed cluster sizes. The paired mean is recomputed in 100,000 replicates with seed 20260808, and percentile 95% intervals are reported. These intervals summarize variation across source worlds. Proof-set-size analyses retain the four derivable strata.

Sensitivity settings. For atom-weighted analyses, weighted Jaccard distance is $d_{J,w}(A, B) = 1 - \sum_{a \in A \cap B} w(a) / \sum_{a \in A \cup B} w(a)$. Node and root weights remain 1, while source-label and edge weights take the tested values. For proof-mass analyses, each proof receives normalized mass proportional to its atom count raised to α . Proposition 2 applies to the uniform-mass case $\alpha = 0$.

6 Results

6.1 Stratified reference analysis

Across all 20,030 test questions, 2,928 (14.62%) contain at least one reference derivation that applies a source rule more than once. Conditioned on the 11,376 derivable questions, the same count is 25.74%. The stratified reference analysis contains 82 of 400 questions (20.5%) with such within-proof reuse. Source-ID projection reduces the number of distinct reference proofs in 28 of 400 questions (7.0%), removing 37 reference proof members through deduplication. All 28 collapse cases

occur within the 82-question subset with repeated source-rule use. By proof-set-size stratum, collapse occurs in 0 of 80 one-proof questions, 0 of 120 two-proof questions, 16 of 120 questions with three or four proofs, and 12 of 80 questions with five to eight proofs. In the separate 17–32-proof high-multiplicity evaluation set, source-ID projection collapses distinct reference proofs in 32 of 40 questions and removes 130 of 859 application-aware reference proofs (15.13%). Table 2 collects the principal reference-data quantities.

Repeated source-rule use records reuse of one source rule inside a derivation. Proof-set collapse records the loss of distinct proof members after source-ID projection. Their counts therefore answer different questions. Collapse appears only in the larger proof-set strata of the stratified reference analysis, which motivates the paired rescoring analysis by proof-set size.

6.2 Output integrity and proof-set cardinality

Across 200 generated outputs, including the eight unknown questions under five presentations, truth-status accuracy and JSON parseability are both 100%. All 433 generated proof members in the 160 derivable outputs pass the structural DAG checks. Application-aware deduplication retains all 433 members, while source-ID projection merges one predicted pair in one output. Twelve of the 160 derivable outputs reach the eight-proof output limit. The mean predicted proof-set size is 2.71 and the mean reference proof-set size is 3.06. Predicted and reference cardinalities match for 117 of 160 outputs (73.1%). Predicted proof sets are smaller in 27 of 160 outputs (16.9%) and larger in 16 of 160 outputs (10.0%).

6.3 Representation effect on fixed outputs

Table 3 reports application-aware and source-ID scores for the same 160 derivable outputs. Application-aware MCT-J averages 0.1447 and source-ID MCT-J averages 0.0842. The paired difference between application-aware and source-ID scoring is 0.0606, with a 95% cluster bootstrap interval over the 30 source worlds of 0.0316–0.0955. Forty-one outputs change, and every changed output receives a lower MCT-J distance under source-ID projection. Relative to the application-aware mean, source-ID projection reduces the measured distance by 41.8% in this paired rescoring analysis.

The five presentation orders have positive mean MCT-J differences ranging from 0.0413 to 0.0880.

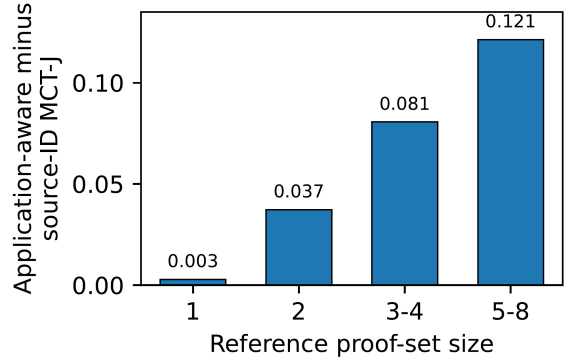


Figure 2: Mean MCT-J difference between application-aware and source-ID scoring by reference proof-set size. Positive values correspond to larger distance under application-aware scoring.

Score changes also extend beyond proof-set collapse. Within the 32-question generation sample, source-ID projection changes reference proof-set size for one question and predicted proof-set size for one generated output. For most of the 41 MCT-J changes, projection leaves both represented proof-set sizes unchanged, so the score change comes from altered grounded-application structure within proof members.

The mean representation effect rises with reference proof-set size (Figure 2): 0.0028 for one proof, 0.0373 for two proofs, 0.0807 for three or four proofs, and 0.1214 for five to eight proofs. Across 32 questions, the descriptive Spearman association between reference proof-set size and the item-level mean representation effect is $\rho = 0.665$. The association pertains to the paired rescoring analysis.

Exact proof-set F_1 changes on 6/160 outputs. Flattened-atom F_1 changes on 44/160 outputs. Its mean application-aware minus source-ID difference is -0.0641 , with a 95% cluster bootstrap interval over the 30 source worlds of $[-0.0980, -0.0349]$. Source-ID projection therefore changes exact proof identity and the atom geometry seen by graded measures.

6.4 Divisible correspondence and cardinality mismatch

For the 117 application-aware outputs with equal predicted and reference proof-set sizes, MCT-J and Jaccard-OSPA have identical values in every case. Among the 43 unequal-cardinality outputs, Jaccard-OSPA has the larger distance in 38 cases and the two distances are equal in five. The

Analysis quantity	Count	Rate
Multi-proof among derivable test questions	2,456/11,376	21.59%
Repeated source-rule use, all test questions	2,928/20,030	14.62%
Repeated source-rule use, stratified reference analysis	82/400	20.50%
Source-ID proof-set collapse, stratified reference analysis	28/400	7.00%
Reference proof members removed by projection	37	—

Table 2: Reference-data quantities. The 2,928/20,030 rate uses all test questions to report corpus prevalence. The stratified-reference rates use the sampled derivable strata.

Measure	Application-aware	Source-ID	Δ	Changed
Exact proof-set $F_1 \uparrow$	.8657	.8697	-.0041	6/160
Flattened-atom $F_1 \uparrow$	.8744	.9384	-.0641	44/160
MCT-J $\downarrow$	.1447	.0842	+.0606	41/160
Jaccard-OSPA $\downarrow$	.1713	.1661	+.0052	8/160

Table 3: Paired rescoring of the same outputs. Δ denotes application-aware minus source-ID. Arrows mark the favorable direction of each measure.

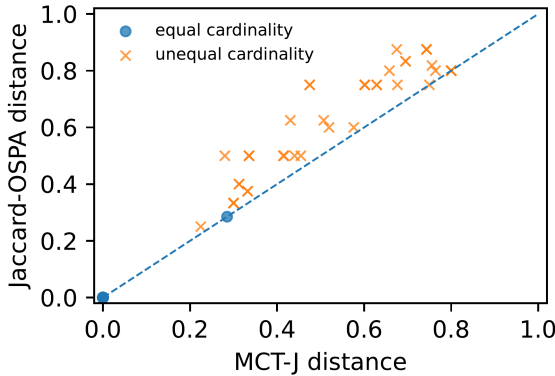


Figure 3: Application-aware MCT-J and Jaccard-OSPA on 160 outputs. Equal-cardinality outputs lie on the diagonal. Unequal-cardinality outputs may separate because balanced transport permits divisible correspondence.

mean OSPA-minus-MCT-J difference over the 43 unequal-cardinality outputs is 0.0988. Figure 3 displays both regimes.

Equal cardinality reproduces the one-to-one optimum established in Proposition 2. Reference and predicted proof-set sizes differ in 43 of 160 generated derivable outputs (26.9%). The eight outputs with a Jaccard-OSPA change are a subset of the 41 outputs with a MCT-J change. This leaves 33 outputs where only MCT-J changes, and reference and predicted proof-set sizes differ in all 33. In each case the Jaccard-OSPA value equals its cardinality penalty because its assigned proof pairs have zero Jaccard cost. Balanced transport still redistributes mass across the unequal proof sets,

so MCT-J remains sensitive to the representation change.

6.5 Rule replay and sensitivity analyses

The rule-replay procedure first reproduces all 2,210 reference proof roots from the stratified reference, generation, and high-multiplicity evaluation sets. These roots contain 4,085 fact nodes and 9,659 rule applications, with zero replay failures. Of 433 generated proof members, 430 pass independent rule-level replay (99.31%). All returned proof members pass replay for 159 of 160 derivable outputs. The three failures occur in one presentation of one question, where the listed premise fails to establish the cited rule antecedent. Excluding these three proof members yields application-aware MCT-J 0.1420 and source-ID MCT-J 0.0831. The paired representation effect is 0.0589 with a 95% cluster bootstrap interval of 0.0297–0.0940, compared with the primary 0.0606 [0.0316, 0.0955]. Renaming local step identifiers and reordering steps and premises leaves the application-aware atom set unchanged for all 433 generated proof members.

One sensitivity analysis adds source-label atoms to the source-ID representation so that both representations use the same structural atom types. The resulting effect is 0.0634 [0.0334, 0.0996], with the same 41 changed outputs. Recursive signature depth is evaluated at depth 1, depth 2, and full recursion. Source-label and edge-atom weights take values in $\{0.5, 1, 2\}$. A one-at-a-time ablation removes each atom type in turn. Proof mass uses $\alpha \in \{0, 0.5, 1\}$ under the definition in Section 5.3. Across these analyses, the effect ranges from 0.0306 to 0.0723, and every reported point-wise 95% cluster bootstrap interval remains above zero.

The adversarial perturbation analysis uses the 24 derivable generation questions with at least two reference proofs. For each question, one reference proof is omitted in turn to form an omission-only predicted family. A different retained proof

serves as the donor. Deleting each non-root donor atom once creates the candidate near-duplicates. Each distinct candidate is added to the predicted family and scored against the original reference family. The construction yields 6,032 perturbations and tests whether a structurally redundant predicted alternative can improve the score. Executable derivation status lies outside this structural diagnostic. Across the perturbations, MCT-J distance decreases in 2,575 cases, increases in 3,032, and ties in 425. Using perturbed minus baseline distance, the mean signed change is -0.000015 . Jaccard-OSPA distance decreases in 5,382 cases and ties in 650, with mean decrease 0.04017.

7 Discussion

The stratified reference analysis isolates a representation effect. Repeated source-rule use is more frequent than proof-set collapse, yet source-ID projection removes distinct reference proofs in 28 of 400 stratified questions. On fixed model outputs, projection also changes grounded-application atoms while the projection usually leaves the represented proof-set sizes unchanged. This explains why MCT-J changes on 41 outputs even though projection changes predicted proof-set size once.

Proof-set size localizes the effect. The one-proof stratum has a mean difference of 0.0028, rising to 0.1214 for five-to-eight-proof questions. Equal-cardinality proof sets reproduce the same one-to-one optimum under Jaccard-OSPA and MCT-J. Unequal cardinality permits divisible transport and produces the observed gap. Across the same fixed outputs, source-ID scoring raises both F_1 measures and lowers both distance measures. All four measures therefore place the source-ID representation closer to the reference on average.

Rule replay changes the mean representation effect from 0.0606 to 0.0589 after the three replay failures are excluded. The sensitivity checks preserve its direction across the tested representation and weighting choices. Under the structural near-duplicate perturbation, Jaccard-OSPA decreases in 5,382 of 6,032 cases with mean decrease 0.04017, while the mean signed MCT-J change is near zero. The two measures therefore respond differently to structurally near-duplicate predicted members.

8 Conclusion

Multi-derivation evaluation requires explicit choices about proof identity and proof-set

correspondence. Grounded application identity preserves distinctions between repeated grounded uses of one source rule. Source-ID projection can collapse those distinctions. In the stratified reference analysis, source-ID projection collapses distinct reference proofs in 28 of 400 derivable questions. On the same 160 generated outputs, it lowers MCT-J distance by 0.0606 on average. Rule replay and sensitivity analyses preserve the direction of this effect. The 17–32-proof high-multiplicity evaluation set extends the collapse result to larger reference proof sets. MCT-J equals Jaccard-OSPA at equal cardinality. Under cardinality mismatch, Jaccard-OSPA has the larger distance for 38 of 43 outputs. Representation and correspondence therefore change structural evaluation on fixed outputs. Explicit specification of both choices supports reproducible evaluation of alternative derivations.

Limitations

ProofWriter uses synthetic rule reasoning under generated Datalog semantics. Transfer to natural mathematical proofs remains a future empirical setting. The primary paired rescoring analysis uses 32 derivable questions from GPT-5.6 Sol. The main experiment evaluates one model, so cross-system evidence remains limited. The stratified reference analysis examines 400 derivable questions. The bootstrap summarizes variation across 30 source worlds, so interval precision is tied to that cluster count.

The structural atom types define the Jaccard geometry. Other structural encodings can produce different local Jaccard distances. The reported sensitivity analyses cover the tested recursion, atom types, weighting, and proof-mass choices. Exact duplicates collapse under set semantics. Structurally distinct near-duplicates retain separate mass. The adversarial perturbation analysis shows that some can lower MCT-J. Human proof-quality judgments and robustness to deliberately constructed near-duplicate proof sets remain separate validation targets.

The output format allows at most eight predicted proofs. Every derivable question in the generation sample has at most eight reference proofs, so its complete reference set fits within this limit. The 17–32-proof high-multiplicity evaluation set extends the analysis of reference proof sets beyond this range. The eight-proof output limit restricts anal-

ysis of larger predicted proof sets. ProofWriter is public, so training-data exposure is possible for contemporary systems.

Evaluation on less controlled natural-language reasoning and on other datasets requires separate studies. The conclusions concern structural evaluation.

References

- Mubashara Akhtar, Michael Schlichtkrull, and Andreas Vlachos. 2026. [Ev2R: Evaluating evidence retrieval in automated fact-checking](#). *Transactions of the Association for Computational Linguistics*, 14:530–561.
- David Alvarez-Melis, Tommi Jaakkola, and Stefanie Jegelka. 2018. Structured optimal transport. In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, pages 1771–1780. PMLR.
- Yunmo Chen, William Gantt, Tongfei Chen, Aaron Steven White, and Benjamin Van Durme. 2023. [A unified view of evaluation metrics for structured prediction](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12868–12882, Singapore. Association for Computational Linguistics.
- Peter Clark, Oyvind Tafjord, and Kyle Richardson. 2020. [Transformers as soft reasoners over language](#). In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 3882–3890. International Joint Conferences on Artificial Intelligence Organization.
- Bhavana Dalvi, Peter Jansen, Oyvind Tafjord, Zhengnan Xie, Hannah Smith, Leighanna Pipatanangkura, and Peter Clark. 2021. [Explaining answers with entailment trees](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7358–7370, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Aayushee Gupta. 2026. [From sentences to proof trees: Leveraging language models for structured reasoning](#). In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, pages 967–981, Rabat, Morocco. Association for Computational Linguistics.
- Alon Jacovi, Yonatan Bitton, Bernd Bohnet, Jonathan Herzig, Or Honovich, Michael Tseng, Michael Collins, Roei Aharoni, and Mor Geva. 2024. [A chain-of-thought is as strong as its weakest link: A benchmark for verifiers of reasoning chains](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4615–4634, Bangkok, Thailand. Association for Computational Linguistics.
- Francisco F. López-Ponce and Gemma Bel-Enguix. 2025. [Into the limits of logic: Alignment methods for formal logical reasoning](#). In *Proceedings of The 3rd Workshop on Mathematical Natural Language Processing (MathNLP 2025)*, pages 112–123, Suzhou, China. Association for Computational Linguistics.
- Mark Obozov, Michael Diskin, Aleksandr Beznosikov, Alexander Gasnikov, and Serguei Barannikov. 2025. [Synthetic proofs with tool-integrated reasoning: Contrastive alignment for LLM mathematics with Lean](#). In *Proceedings of The 3rd Workshop on Mathematical Natural Language Processing (MathNLP 2025)*, pages 195–202, Suzhou, China. Association for Computational Linguistics.
- OpenAI. 2026. [GPT-5.6 Sol model](#). OpenAI API documentation. Accessed 19 September 2026.
- Juri Opitz, Angel Daza, and Anette Frank. 2021. [Weisfeiler-Leman in the Bamboo: Novel AMR graph metrics and a benchmark for AMR graph similarity](#). *Transactions of the Association for Computational Linguistics*, 9:1425–1441.
- Gabriel Peyré and Marco Cuturi. 2019. [Computational optimal transport: With applications to data science](#). *Foundations and Trends in Machine Learning*, 11(5–6):355–607.
- Auguste Poiroux, Gail Weiss, Viktor Kunčák, and Antoine Bosselut. 2025. [Reliable evaluation and benchmarks for statement autoformalization](#). In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 17947–17969, Suzhou, China. Association for Computational Linguistics.
- Archiki Prasad, Swarnadeep Saha, Xiang Zhou, and Mohit Bansal. 2023. [ReCEval: Evaluating reasoning chains via correctness and informativeness](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10066–10086, Singapore. Association for Computational Linguistics.
- Swarnadeep Saha, Sayan Ghosh, Shashank Srivastava, and Mohit Bansal. 2020. [PProver: Proof generation for interpretable reasoning over rules](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 122–136, Online. Association for Computational Linguistics.
- Swarnadeep Saha, Prateek Yadav, and Mohit Bansal. 2021. [multiPProver: Generating multiple proofs for improved interpretability in rule reasoning](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3662–3677, Online. Association for Computational Linguistics.
- Michael Schlichtkrull, Yulong Chen, Chenxi Whitehouse, Zhenyun Deng, Mubashara Akhtar, Rami Aly, Zhijiang Guo, Christos Christodoulopoulos, Oana Cocarascu, Arpit Mittal, James Thorne, and Andreas

- Vlachos. 2024. [The automated verification of textual claims \(AVeriTeC\) shared task](#). In *Proceedings of the Seventh Fact Extraction and VERification Workshop (FEVER)*, pages 1–26, Miami, Florida, USA. Association for Computational Linguistics.
- Dominic Schuhmacher, Ba-Tuong Vo, and Ba-Ngu Vo. 2008. [A consistent metric for performance evaluation of multi-object filters](#). *IEEE Transactions on Signal Processing*, 56(8):3447–3457.
- Oyvind Tafjord, Bhavana Dalvi, and Peter Clark. 2021. [ProofWriter: Generating implications, proofs, and abductive statements over natural language](#). In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3621–3634, Online. Association for Computational Linguistics.
- James Thorne, Andreas Vlachos, Oana Cocarascu, Christos Christodoulopoulos, and Arpit Mittal. 2018. [The fact extraction and VERification \(FEVER\) shared task](#). In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 1–9, Brussels, Belgium. Association for Computational Linguistics.