Studies in Knowledge and Innovation
Vol. 2 No. 4A (2026)

Entity-ID Reversal Changes a Temporal Similarity Routine While Endpoint Evidence Stays Fixed

Jeffrey Franklin[1] · Nicholas Murphy[2] · Evan Anderson[3]

[1] Denmark · [2] Norway · [3] Italy

Published: April 30, 2026 | Article 845

Abstract

Changing the numbers used to name entities should be distinguishable from changing their facts. A fixed reversal of entity IDs was applied to two released temporal graph benchmarks while preserving relations, time codes and known correspondences. A literal replay of one pinned preprocessing routine changed 1,727 of 8,566 paired scores in ICEWS05-15 and 160 of 49,172 in YAGO-WIKI. Mean absolute changes were 0.085894 and 0.000751 on a zero-to-one similarity scale. An explicitly defined entity-endpoint reference stayed invariant. Both projections conserved total occurrence mass, so that check alone did not expose their difference. These are finite record and source-routine results, rather than measurements of trained alignment accuracy or claims about the runtime behind published experiments.

Entity-ID Reversal Changes a Temporal Similarity Routine While Endpoint Evidence Stays Fixed

Abstract

Changing the numbers used to name entities should be distinguishable from changing their facts. A fixed reversal of entity IDs was applied to two released temporal graph benchmarks while preserving relations, time codes and known correspondences. A literal replay of one pinned preprocessing routine changed 1,727 of 8,566 paired scores in ICEWS05-15 and 160 of 49,172 in YAGO-WIKI. Mean absolute changes were 0.085894 and 0.000751 on a zero-to-one similarity scale. An explicitly defined entity-endpoint reference stayed invariant. Both projections conserved total occurrence mass, so that check alone did not expose their difference. These are finite record and source-routine results, rather than measurements of trained alignment accuracy or claims about the runtime behind published experiments.

1. A record can stay the same while its representation changes

Entity identifiers are essential for storing a graph, but their numeric spelling need not describe the entities. A correspondence between two nodes should remain the same correspondence when both identifiers and all their occurrences are renamed consistently. This distinction gives a concrete way to examine preprocessing: preserve the endpoint facts, change their numeric encoding, and compare the constructed evidence. The question is not whether two independently collected graphs agree. It is whether the representation of an unchanged graph introduces a difference before any alignment model is trained.

Knowledge graphs bring together entities, relations and other contextual information, so the interpretation of each field matters to their use (Hogan et al., 2022). Entity matching is also broader than string equality or one numerical score; record-linkage foundations and data-matching research explicitly concern how evidence is represented and compared (Fellegi and Sunter, 1969; Christen, 2012). In the present study, the released correspondence list is held fixed. There is no attempt to infer a new set of identities, identify people, or assess the substantive truth of the underlying events.

Temporal alignment makes the issue particularly relevant because a timestamp field can become an input to later evidence combination. The final publisher description of T3A-LLM places relation/time feature encoding before candidate graph matching and subsequent semantic score fusion (Wang et al., 2026). That supplies a specific methodological reason to inspect temporal representations at their point of construction. The experiment here concerns a different project's pinned helper and routine. It neither assumes that T3A uses those files nor evaluates its candidate set, language-model stage or reported performance.

Temporal candidate disagreement can determine whether an alignment pipeline seeks further evidence. ATEA describes consistency-aware reasoning that introduces external evidence when internal candidate signals disagree (Wu et al., 2026). The present ID-reversal result identifies a prerequisite for interpreting such a trigger: a representation change can alter a temporal signal even though the entity facts remain fixed. A preprocessing contract therefore belongs before a conflict is interpreted as semantic uncertainty. We do not show that ATEA uses the audited helper, that its retrieval is triggered by these scores, or that its reported alignment results change.

The source setting is the public LightTEA artifact associated with time-aware entity alignment (Cai et al., 2023). We independently read two of its files and compare a literal field-role replay with an expressly defined entity-endpoint reference. The research question is: under a fixed renaming of entity IDs, which temporal profiles and known-pair similarities change, even though endpoint facts and supplied

correspondences are preserved? The answer is based on complete released numeric tables, not on a new alignment benchmark submission or a reconstruction of published accuracy.

2. An invariance requirement and the observations that can test it

The proposed invariance requirement is conditional on a clear intended representation. If temporal evidence is defined by time codes incident on actual entity endpoints, renaming those endpoints consistently should leave the paired evidence unchanged. Relational inductive-bias and geometric-learning work explains the broader importance of respecting representational symmetries (Battaglia et al., 2018; Bronstein et al., 2021). We do not import their architectures or infer a performance result from them. Their relevant idea here is that a change of labels and a change of graph information are different operations.

This is also a testing problem in which an expected relationship between executions can be useful. The oracle problem concerns how a test determines whether an output is acceptable (Barr et al., 2015); metamorphic testing studies relationships between transformed inputs and outputs (Segura et al., 2016). Applications to machine-learning classifiers illustrate that such relationships can be examined without simply comparing a single output to a known answer (Xie et al., 2011). Our test uses a declared identity-renaming relationship. It does not claim that passing one relationship validates every property of a temporal alignment system.

Field roles must be stated before the transformation is evaluated. A positional tuple can contain several integers that belong to different domains. Merely observing that all entries are integers does not establish that a relation code can be used as an entity identifier. The relational model emphasizes structured data and its operations (Codd, 1970), while work on types and data abstraction distinguishes values through the roles permitted for them (Cardelli and Wegner, 1985). The audit treats entity identity, relation identity and time code as separate roles, even when their integer ranges overlap.

An accompanying assertion must be selected with similar care. Design-by-contract approaches make expected behavior explicit (Meyer, 1992). Here the declared contract is invariance of endpoint-based temporal evidence under a consistent within-graph ID permutation. Conservation of the total number of time occurrences is a second, weaker property. A procedure can satisfy that total while assigning occurrences to different entities. Both properties are therefore measured, rather than treating a successful total as a substitute for the semantic contract.

3. Fixed sources and two profile constructions

The analysis uses the LightTEA data archive and read-only source text at commit a441379b000dcfd90ba146060dc051ffc5e1c78b. The archive contains entity lists, triples and supplied/reference correspondence lists for ICEWS05-15 and YAGO-WIKI. Existing download records preserve archive and member checksums, and the raw triple hashes were checked again during independent recalculation. The two helper files were downloaded as text. No author-provided program, cached pickle, neural model or published runtime environment was executed.

ICEWS05-15 contains 615,105 source rows across its two triple tables. Each row has entity, relation, entity and one time-code field. YAGO-WIKI contains 538,864 rows with two time-code fields after the three graph fields. Those latter fields are handled as two recorded endpoints by the pinned helper, rather than expanded into every intervening calendar year. There are 93,739 rows with equal endpoint codes, whose repeated contributions are retained. Time codes remain opaque integers; no integer-to-calendar conversion or interpretation of code zero as a real year is introduced.

The nonzero-code convention follows the particular temporal routine being examined. It excludes 55 code slots in ICEWS05-15 and 95,283 in YAGO-WIKI. The remaining counts are 615,050 and 982,445 represented time slots. These are source-processing denominators, not independent observed events. A row can contribute more than once through its endpoints and the forward/reverse expansion. No raw-row deduplication, endpoint coalescing or inferred event reconstruction is added to the source convention.

The helper constructs directed records in entity-entity-relation-time order, including forward and reverse relation codes. The consuming temporal routine uses the first and third tuple positions as profile write destinations. A literal replay therefore records a time code at an actual entity-head index and at a recoded relation index. These are statements about the cached pair of files if used together as read. The audit does not determine which import resolution, revision or execution generated any publication result, and does not assign an intention to the source authors.

The comparison reference is defined independently: each directed record contributes its nonzero time to both actual entity endpoints. Because the source expands forward and reverse records, this counts each raw endpoint contribution twice. That uniform duplication is retained; it does not change Dice similarity when both profiles are scaled together. The reference is a transparent endpoint-incidence rule, rather than a claim to reconstruct an author's intended correction or an external gold-standard alignment algorithm.

Temporal Dice similarity is twice the sum of the minimum count for each shared code, divided by the combined profile masses. The source routine returns an arithmetic zero when no time code intersects. The analysis retains separate flags for empty supply and nonempty disjoint supply so that zero would not automatically mean observed temporal disagreement. In these two released cases, all declared entities have nonempty endpoint profiles. Consequently, missing entity-level time supply does not explain the score changes reported below.

The encoding intervention reverses the sorted entity-ID list separately inside each graph. All occurrences of entity endpoints and known-pair keys are mapped through the same bijection. Relation codes, time codes, row multiplicity and graph membership remain unchanged. Both original entity namespaces remain contiguous and disjoint. This is one fixed, deterministic representation change; it is not a search for a maximally adverse permutation. Reversing it again restores the original identity encoding.

The complete known-pair list pools the released supplied and reference pairs because no learning or hyperparameter selection is performed. It contains 8,566 unique pairs in ICEWS05-15 and 49,172 in YAGO-WIKI. Benchmarking studies emphasize that alignment evaluation depends on its data and protocol (Sun et al., 2020), while graph-convolutional alignment is an example of a separate learned modeling approach (Wang et al., 2018). Neither is reproduced here. We compare scores on these fixed pairs without claiming retrieval accuracy, model generalization or a new competitive ranking.

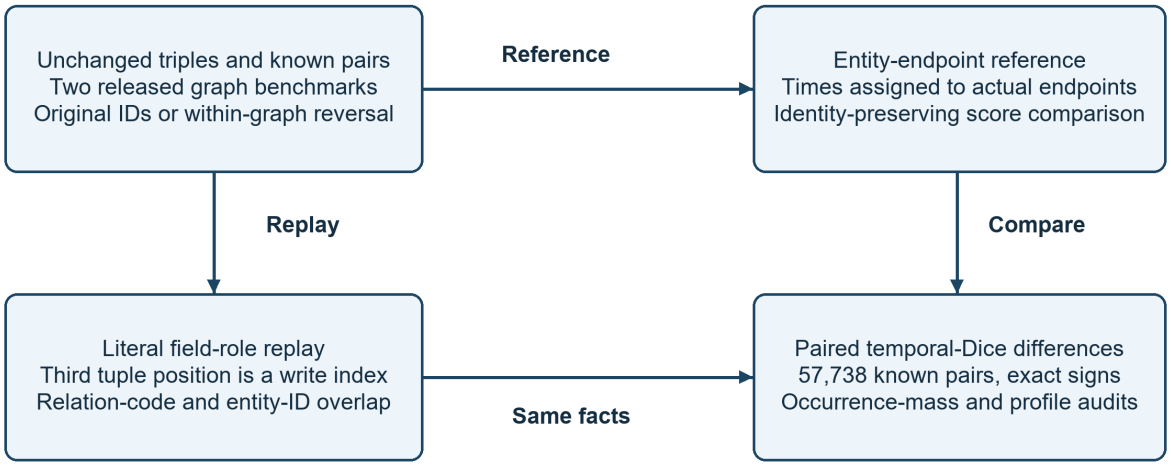


Figure 1. Two projection rules are compared on unchanged facts under two numeric identity encodings. The endpoint-reference branch defines the invariance requirement; the literal branch follows the pinned field-role assignment. Arrows indicate arithmetic construction.

Each pair's numerator and denominator are saved as integers. Exact integer cross-products determine whether the two rational scores differ and the direction of change. Mean differences use stable finite arithmetic. An independent implementation explicitly constructs directed tuples from the raw records and recalculates 346,428 integer score components, then checks the aggregate means with seventy-digit decimal arithmetic. This supplies numerical evidence for the finite comparison, without adding a sampling model or converting deterministic extrema into confidence intervals.

4. Conserved totals and changing pair scores

The endpoint reference remains invariant on every one of the 57,738 known pairs. Its mean scores are 0.611050 for ICEWS05-15 and 0.714373 for YAGO-WIKI. These values describe the explicitly defined projection on the released pair lists. They are not accuracy scores and do not establish whether any correspondence is correct. Their unchanged values provide the comparison required by the stated identity-renaming contract.

The literal projection behaves differently. In ICEWS05-15, the mean known-pair score changes from 0.587838 to 0.548380, a signed difference of -0.039458. Exact rational comparison identifies 1,727 changed pairs, or 20.16% of the list. YAGO-WIKI has a much smaller average shift, from 0.714189 to 0.713819, with 160 changed pairs, or 0.33%. Both results are retained rather than using the more responsive dataset as the sole example.

Table 1. Literal-projection temporal-Dice scores before and after within-graph entity-ID reversal. Each row uses the full released known-pair list. Differences are in similarity-score units on a zero-to-one scale, not percentage points of alignment accuracy. Changed counts use exact rational equality.

Dataset	Known pairs	Original mean	Reversed mean	Mean difference	Changed pairs
ICEWS05-15	8566	0.587838	0.548380	-0.039458	1727 (20.16%)
YAGO-WIKI	49172	0.714189	0.713819	-0.000370	160 (0.33%)

Average absolute change is 0.085894 in ICEWS05-15 and 0.000751 in YAGO-WIKI. The latter small overall value does not mean every pair is nearly invariant: its observed pair differences range from approximately -1.000 to +0.897. Conversely, those extremes do not characterize most YAGO-WIKI pairs, because 49,012 retain exactly the same score. A report of either the small mean or the largest difference alone would leave out an important feature of this finite distribution.

Table 2. Endpoint-reference scores and the distribution of literal score changes. The reference mean is identical under both encodings. Pair extrema and mean absolute differences describe only the fixed reversal; they are neither confidence limits nor worst cases over every possible encoding.

Dataset	Endpoint-reference mean	Mean absolute reversal difference	Smallest pair difference	Largest pair difference
ICEWS05-15	0.611050	0.085894	-0.999961	+0.992647
YAGO-WIKI	0.714373	0.000751	-0.999989	+0.896739

The signs also show why counting increases and decreases is insufficient. ICEWS05-15 has 967 increases and 760 decreases, yet its mean signed change is negative. The decreasing differences are larger in aggregate. YAGO-WIKI has 57 increases and 103 decreases. These counts retain both directions rather than treating every altered pair as a deterioration. No desirable direction is assigned to a larger temporal similarity in the absence of a trained retrieval evaluation or an external correctness assessment.

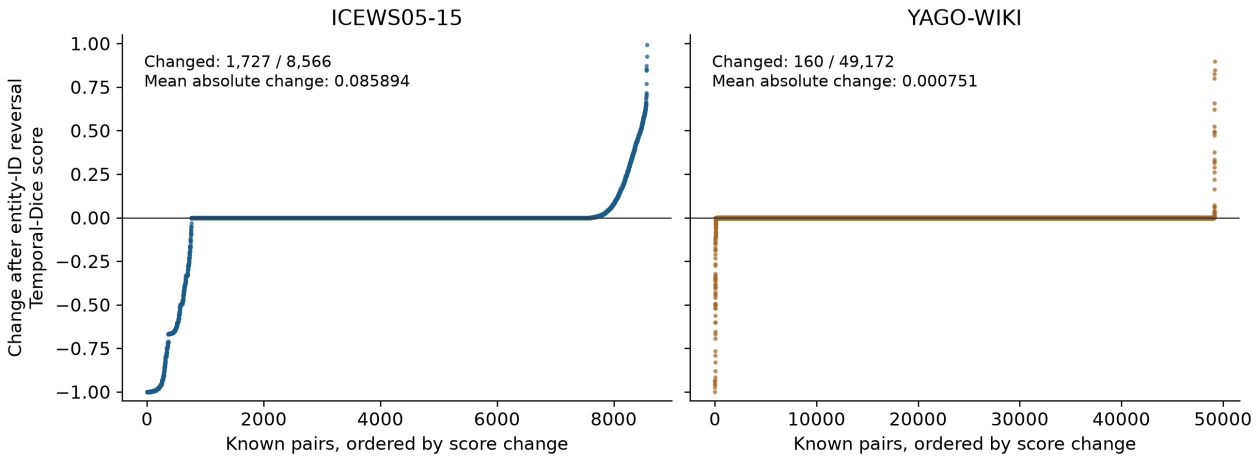


Figure 2. Every pair is plotted in increasing order of its literal score difference. The zero plateau includes unchanged pairs. Different horizontal lengths reflect the distinct list sizes; the common vertical scale prevents a small overall change from being visually expanded into an apparent large effect.

The write-destination accounting explains the concentration. Recoded relation indices fall inside the first graph's entity-ID range in both releases. In the original encoding, nonzero relation-index writes reach 986 first-graph entity IDs in ICEWS05-15 and 82 in YAGO-WIKI. They reach no second-graph entity ID. Reversing the first graph's IDs moves these additions to different underlying entities while preserving the number of numeric destinations. This is an observed property of these releases' number ranges, not a universal property of temporal graphs.

Table 3. Occurrence accounting by graph under the original encoding. A relation-index destination is an entity-number slot receiving a write through the third tuple position. The masses are representation counts, not independent events or sample sizes. Reversal preserves each graph's literal total while changing which underlying first-graph entities receive those writes.

Dataset	Graph	Entities	Relation-index destinations	Endpoint-reference mass	Original literal mass
ICEWS05-15	1	9517	986	1230096	1845148
ICEWS05-15	2	9537	0	1230104	615052
YAGO-WIKI	1	49629	82	1471344	2700562
YAGO-WIKI	2	49222	0	2458436	1229218

Global totals fail to distinguish the two projections. Both have 2,460,200 nonzero occurrence writes in ICEWS05-15 and 3,929,780 in YAGO-WIKI. The literal rule redistributes mass across graph namespaces: for example, ICEWS05-15's endpoint-reference masses are 1,230,096 and 1,230,104, whereas its literal masses are 1,845,148 and 615,052. Each total is internally conserved. The implication is specific: a successful global count check does not prove that occurrences have been assigned to the intended field domain.

5. What can be inferred from this source-level comparison

The analysis isolates a representational dependency without measuring its consequence for a complete learned system. Only one helper/routine pair, one reversal and two released benchmarks are examined. A downstream system could transform, normalize, discard or combine the affected profiles differently. It could also use another revision or data loader. Reproducibility guidance makes the data and computational procedures being repeated important to the meaning of a claimed reproduction (National Academies of Sciences, Engineering, and Medicine, 2019). The present results therefore support inspection of field-role contracts and encoded evidence; they do not support a revised ranking of published alignment methods or an accusation that reported experimental values are wrong.

The identity-renaming contract should also be stated openly. A procedure deliberately designed to use numeric IDs as substantive features would not promise this invariance. Our endpoint-based reference instead defines the intended information as graph facts and time-code incidence. This choice makes the test interpretable. Work on knowledge-graph refinement distinguishes the quality and interpretation of the graph from a particular downstream task (Paulheim, 2016). The present comparison remains at that representation layer and does not resolve the factual reliability of any individual triple.

Provenance is necessary for locating such a difference but does not remove the need to understand it. Source-lineage and database-provenance research supplies ways to describe where an output came from and which inputs contributed (Buneman et al., 2001; Cheney et al., 2009; Simmhan et al., 2005). The PROV ontology provides a further vocabulary for entities, activities and derivations (Lebo et al., 2013). Here immutable source hashes identify the file pair and raw tables. Those records make the comparison traceable; they do not themselves prove that a positional field was interpreted correctly.

Dataset documentation and interoperable research records similarly help expose the relevant assumptions (Gebru et al., 2021; Wilkinson et al., 2016). The analysis keeps source rows, endpoint multiplicity, code-zero handling, ID ranges, correspondence lists and both profile rules explicit. A tidy separation of variables is useful when one column's domain could otherwise be confused with another (Wickham,

2014). None of these documentation practices substitutes for the actual replay: the changing paired scores and conserved totals must still be calculated from the observed records.

Scientific-software testing reviews motivate complementary numerical checks rather than a single aggregate assertion (Kanewala and Bieman, 2014). Here the independent path uses explicit directed tuples instead of the primary calculation's decomposed head and relation-slot counters. Self-loops, equal time endpoints and code-zero exclusions remain part of both paths. Checking complete score components and entity masses is more informative for this question than merely confirming that the final output file exists or that two rounded means look similar.

6. Conclusion and reproducibility

A fixed reversal of entity IDs changes a literal temporal-profile routine's known-pair scores while an entity-endpoint reference remains invariant. The effect is concentrated in part of each released graph, and its prevalence and mean magnitude differ sharply between the two benchmarks. Total occurrence mass stays conserved throughout. These results show why identity encoding and field-role interpretation deserve explicit checks before temporal evidence is used elsewhere. They do not establish a trained model's accuracy, a publisher-runtime error or the optimal method for entity alignment.

Saved outputs include all 57,738 paired scores, their integer components, 117,905 entity-profile accounting records, source hashes and the frozen analysis protocol. The independent raw-record recalculation is separate from the primary transformation. Computational reproducibility guidance supports retaining executable analyses and their dependencies (Peng, 2011; Sandve et al., 2013). NumPy is used in the local tabular and figure workflow (Harris et al., 2020); its availability is not evidence of empirical validity. The independent checker instead uses standard-library counters and high-precision arithmetic rather than a second call to the primary script.

Local Scientific Agent Skills documentation informed procedural preparation (Kassis et al., 2026); its citation records that assistance rather than validating the findings. Source code was read as text and the benchmark's original notices remain distinct from code-library terms. No model training, GPU timing, private data upload or paid analysis API was used. The artifacts remain drafts pending the batch's full reference, originality, figure and PDF review, with accountable human source-claim and authorship approval still to be recorded.

References

1. Harris, Charles R.; Millman, K. Jarrod; van der Walt, Stéfan J.; Gommers, Ralf; Virtanen, Pauli; Cournapeau, David; Wieser, Eric; Taylor, Julian; Berg, Sebastian; Smith, Nathaniel J.; Kern, Robert; Picus, Matti; Hoyer, Stephan; van Kerkwijk, Marten H.; Brett, Matthew; Haldane, Allan; del Río, Jaime Fernández; Wiebe, Mark; Peterson, Pearu; Gérard-Marchant, Pierre; Sheppard, Kevin; Reddy, Tyler; Weckesser, Warren; Abbasi, Hameer; Gohlke, Christoph; Oliphant, Travis E.. (2020). Array programming with NumPy. Nature. https://doi.org/10.1038/s41586-020-2649-2.
2. Cai, Li; Mao, Xin; Xiao, Youshao; Wu, Changxu; Lan, Man. (2023). An Effective and Efficient Time-aware Entity Alignment Framework via Two-aspect Three-view Label Propagation. Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence. https://doi.org/10.24963/ijcai.2023/558
3. Buneman, Peter; Khanna, Sanjeev; Tan, Wang-Chiew. (2001). Why and Where: A Characterization of Data Provenance. Database Theory - ICDT 2001, 316-330. https://doi.org/10.1007/3-540-44503-X_20.

4. Wilkinson, Mark D.; Dumontier, Michel; Aalbersberg, IJsbrand Jan; Appleton, Gabrielle; Axton, Myles; Baak, Arie; Blomberg, Niklas; Boiten, Jan-Willem; da Silva Santos, Luiz Bonino; Bourne, Philip E.; Bouwman, Jildau; Brookes, Anthony J.; Clark, Tim; Crosas, Mercè; Dillo, Ingrid; Dumon, Olivier; Edmunds, Scott; Evelo, Chris T.; Finkers, Richard; Gonzalez-Beltran, Alejandra; Gray, Alasdair J.G.; Groth, Paul; Goble, Carole; Grethe, Jeffrey S.; Heringa, Jaap; 't Hoen, Peter A.C; Hooft, Rob; Kuhn, Tobias; Kok, Ruben; Kok, Joost; Lusher, Scott J.; Martone, Maryann E.; Mons, Albert; Packer, Abel L.; Persson, Bengt; Rocca-Serra, Philippe; Roos, Marco; van Schaik, Rene; Sansone, Susanna-Assunta; Schultes, Erik; Sengstag, Thierry; Slater, Ted; Strawn, George; Swertz, Morris A.; Thompson, Mark; van der Lei, Johan; van Mulligen, Erik; Velterop, Jan; Waagmeester, Andra; Wittenburg, Peter; Wolstencroft, Katherine; Zhao, Jun; Mons, Barend. (2016). The FAIR Guiding Principles for scientific data management and stewardship. Scientific Data. https://doi.org/10.1038/sdata.2016.18.

5. Meyer, B.. (1992). Applying 'design by contract'. Computer. https://doi.org/10.1109/2.161279

6. Paulheim, Heiko. (2016). Knowledge graph refinement: A survey of approaches and evaluation methods. Semantic Web. https://doi.org/10.3233/sw-160218

7. Codd, E. F.. (1970). A relational model of data for large shared data banks. Communications of the ACM. https://doi.org/10.1145/362384.362685

8. National Academies of Sciences, Engineering, and Medicine. (2019). Reproducibility and Replicability in Science. The National Academies Press. https://doi.org/10.17226/25303.

9. Gebru, Timnit, et al. "Datasheets for datasets" *Communications of the ACM*, 2021. https://doi.org/10.1145/3458723.

10. Wickham, Hadley. (2014). Tidy Data. Journal of Statistical Software. https://doi.org/10.18637/jss.v059.i10

11. Peng, Roger D.. (2011). Reproducible Research in Computational Science. Science. https://doi.org/10.1126/science.1213847

12. Hogan, Aidan; Blomqvist, Eva; Cochez, Michael; D'amato, Claudia; Melo, Gerard De; Gutierrez, Claudio; Kirrane, Sabrina; Gayo, José Emilio Labra; Navigli, Roberto; Neumaier, Sebastian; Ngomo, Axel-Cyrille Ngonga; Polleres, Axel; Rashid, Sabbir M.; Rula, Anisa; Schmelzeisen, Lukas; Sequeda, Juan; Staab, Steffen; Zimmermann, Antoine. (2022). Knowledge Graphs. ACM Computing Surveys. https://doi.org/10.1145/3447772

13. Christen, Peter. (2012). Data Matching: Concepts and Techniques for Record Linkage, Entity Resolution, and Duplicate Detection. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-31164-2

14. Wang, Tao; Zhao, Runhao; Tang, Jiuyang; Wang, Xianghan; Liu, Xiaolu; He, Sijie. (2026). T3A-LLM: A Two-Stage Temporal Knowledge Graph Alignment Method Enhanced by LLM. Lecture Notes in Computer Science. https://doi.org/10.1007/978-3-032-15535-1_13.

15. Lebo, Timothy; Sahoo, Satya; McGuinness, Deborah. (2013). PROV-O: The PROV Ontology. W3CRecommendation;editors. https://www.w3.org/TR/prov-o/

16. Fellegi, Ivan P.; Sunter, Alan B.. (1969). A Theory for Record Linkage. Journal of the American Statistical Association. https://doi.org/10.1080/01621459.1969.10501049

17. Barr, Earl T.; Harman, Mark; McMinn, Phil; Shahbaz, Muzammil; Yoo, Shin. (2015). The Oracle Problem in Software Testing: A Survey. IEEE Transactions on Software Engineering. https://doi.org/10.1109/tse.2014.2372785

18. Cardelli, Luca; Wegner, Peter. (1985). On understanding types, data abstraction, and polymorphism. ACM Computing Surveys. https://doi.org/10.1145/6041.6042

19. Cheney, James; Chiticariu, Laura; Tan, Wang-Chiew. (2009). Provenance in Databases: Why, How, and Where. Foundations and Trends in Databases. https://doi.org/10.1561/1900000006

20. Segura, Sergio; Fraser, Gordon; Sanchez, Ana B.; Ruiz-Cortes, Antonio. (2016). A Survey on Metamorphic Testing. IEEE Transactions on Software Engineering. https://doi.org/10.1109/tse.2016.2532875

21. Wu, Wenjun; Xu, Kuangshan; Zhao, Runhao; Zhang, Shiqi; Zeng, Weixin; Tang, Jiuyang. (2026). Adaptive retrieval-augmented evidence-fusion for highly heterogeneous temporal entity alignment. Information Fusion. https://doi.org/10.1016/j.inffus.2026.104516.

22. Wang, Zhichun; Lv, Qingsong; Lan, Xiaohan; Zhang, Yu. (2018). Cross-lingual Knowledge Graph Alignment via Graph Convolutional Networks. Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. https://doi.org/10.18653/v1/d18-1032

23. Simmhan, Yogesh L.; Plale, Beth; Gannon, Dennis. (2005). A survey of data provenance in e-science. ACM SIGMOD Record. https://doi.org/10.1145/1084805.1084812

24. Battaglia, Peter W.; Hamrick, Jessica B.; Bapst, Victor; Sanchez-Gonzalez, Alvaro; Zambaldi, Vinicius; Malinowski, Mateusz; Tacchetti, Andrea; Raposo, David; Santoro, Adam; Faulkner, Ryan; Gulcehre, Caglar; Song, Francis; Ballard, Andrew; Gilmer, Justin; Dahl, George; Vaswani, Ashish; Allen, Kelsey; Nash, Charles; Langston, Victoria; Dyer, Chris; Heess, Nicolas; Wierstra, Daan; Kohli, Pushmeet; Botvinick, Matt; Vinyals, Oriol; Li, Yujia; Pascanu, Razvan. (2018). Relational inductive biases, deep learning, and graph networks. arXiv preprint. https://arxiv.org/abs/1806.01261

25. Xie, Xiaoyuan; Ho, Joshua W.K.; Murphy, Christian; Kaiser, Gail; Xu, Baowen; Chen, Tsong Yueh. (2011). Testing and validating machine learning classifiers by metamorphic testing. Journal of Systems and Software. https://doi.org/10.1016/j.jss.2010.11.920

26. Kanewala, Upulee; Bieman, James M.. (2014). Testing scientific software: A systematic literature review. Information and Software Technology. https://doi.org/10.1016/j.infsof.2014.05.006

27. Sun, Zequn; Zhang, Qingheng; Hu, Wei; Wang, Chengming; Chen, Muhao; Akrami, Farahnaz; Li, Chengkai. (2020). A benchmarking study of embedding-based entity alignment for knowledge graphs. Proceedings of the VLDB Endowment. https://doi.org/10.14778/3407790.3407828

28. Sandve, Geir Kjetil; Nekrutenko, Anton; Taylor, James; Hovig, Eivind. (2013). Ten Simple Rules for Reproducible Computational Research. PLoS Computational Biology. https://doi.org/10.1371/journal.pcbi.1003285

29. Bronstein, Michael M.; Bruna, Joan; Cohen, Taco; Veličković, Petar. (2021). Geometric Deep Learning: Grids, Groups, Graphs, Geodesics, and Gauges. arXiv preprint. https://arxiv.org/abs/2104.13478

30. Kassis, Timothy; Agarwal, Vinayak; He, Yuhuan; Patel, Darshil; Brueckner, Aubrey M.. (2026). Scientific Agent Skills: A Library of Procedural Knowledge for Research Agents. arXiv preprint. https://arxiv.org/abs/2609.00065