



Cutting the Fat: A Self-Applied Survey of Excess in AI-Generated Work

Stargazing Creativities Lab

Signal. A defensible cut removes burden while preserving the user’s required content, behavior, and access.

AI-generated work can contain unnecessary explanation, edits, features, and decoration. This survey connects research on preference optimization and evaluation bias with summarization, software engineering, HCI, and technical writing. Length and formatting can influence training and judging, but brevity alone does not establish quality. Direct support for preservation prompting is strongest in constrained code repairs; self-review results depend on the task and feedback. We propose a reversible cut pass that states protected requirements, nominates removals, checks preservation, and restores damaging cuts. Evidence grades distinguish direct findings from transferred evidence and practitioner guidance. We apply the procedure to this paper using a frozen first draft, script-generated section counts, a cut ledger, and a recorded restoration. This worked case demonstrates an auditable reduction, not unchanged reader comprehension or validated effectiveness across artifact types.

1 Introduction

An AI-built deliverable can impose avoidable work on its recipient. A function may arrive with an unrequested abstraction, an explanation with repeated conclusions, or an interface with unused controls. These examples define the concern; they do not estimate its prevalence. Direct evidence is strongest for response length, evaluation bias, and constrained code edits (Singhal et al., 2024; Zheng et al., 2023; Zhu et al., 2026).

We define *excess* as material or behavior removable without violating the agreed task, reducing required correctness, or making the intended user’s work harder. Context matters. An example may help a novice, an input check may implement a required behavior, and a label may prevent guessing (Kalyuga et al., 2003; World Wide Web Consortium, 2024). Length, redundancy, complexity, and unrequested scope are therefore separate properties. Fewer lines or visible elements need not mean less work for the recipient.

The intended reader is a small studio controlling prompts, edits, tests, and acceptance criteria, often without access to model training. We examine training evidence to explain documented incentives, then concentrate on interventions available during production. The goal is an appropriately scoped deliverable with preserved value. No universal deletion target follows from the evidence.

Contributions.

1. An evidence map separating documented causes from broader claims about artifact bloat.
2. Measures that pair reduction with requirements, correctness, and task performance.
3. A reversible cut procedure and evidence-graded findings.
4. A self-application with a preserved draft, computed counts, and an edit audit.

2 Method of the Survey

We searched on 22 October 2026 using two web-search routes, followed by direct retrieval from arXiv, ACL Anthology, conference proceedings, publisher and author sites, and official documentation. The three search areas were alignment and evaluation; editing and software engineering; and summarization, HCI, cognitive load, and writing. Terms included “length bias”, “verbosity”, “RLHF”, “DPO”, “minimal code edits”, “self-correction”, “visual clutter”, and “redundancy”. Appendix A records all 61 discovery queries and their search routes.

Inclusion required a relevant cause, measure, intervention, or preservation risk. We preferred primary studies, methodological papers, standards, and official documentation, adding selected reviews and explicitly labeled practitioner guidance. English-language sources accessible as full text or an informative abstract were eligible; abstract-only access supported only claims stated there. Unsupported anecdotes, promotional claims without methods, and model-compression research unrelated to user-facing excess were excluded.

For all 51 retained sources, we checked title, authors, year, and venue or URL against an opened source and recorded claim support, limitations, access, and version. The audit distinguishes publication from deposit and revision dates. We did not enumerate every search hit or pool incompatible outcomes. This is a structured narrative survey, not an exhaustive systematic review or statistical meta-analysis.

Evidence grades. Tags apply to bounded claims. [STRONG] denotes convergent direct evidence or an established measurement property. [MODERATE] denotes a narrower direct study or evidence with transfer limits. [LIMITED] denotes a proposed adaptation, practitioner recommendation, or little direct testing. Standards establish applicable obligations, not experimental effect sizes. Grades are editorial judgments; successful components do not establish the effectiveness of their combination.

3 Why AI Over-Produces

3.1 Preference optimization can reward length

Demonstrations, preference comparisons, reward modeling, and policy optimization provide a route for learned response-style incentives (Ouyang et al., 2022). In three helpfulness settings, Singhal et al. (2024) found that increasing length drove much of the reward improvement; a length-only reward reproduced much of the measured gain. Shen et al. (2023) also identified sequence length as a reward-model shortcut. Direct preference optimization can exploit length without a separately trained reward model (Park et al., 2024).

These findings support a conditional mechanism, not a universal law. Singhal et al.’s harmlessness-only experiment did not increase length. The studies do not establish that preference training causes every extra feature or abstraction. [STRONG] In a synthetic study, optimizing an imperfect proxy could lower a separate gold-standard reward even as the proxy rose; the gold standard was itself a model (Gao et al., 2023).

3.2 Evaluation can reward the same appearance

LLM judges can prefer redundant expansion, with susceptibility differing by judge (Zheng et al., 2023). In the setting studied by Saito et al. (2023), GPT-4 favored longer responses more than human raters. Length-Controlled AlpacaEval adjusts estimated preference to an equal-length comparison and reduces verbosity manipulation in that benchmark (Dubois et al., 2024). This adjustment neither removes every evaluation bias nor establishes that useful length differences should disappear.

Formatting matters too. Lists, links, bold text, and emojis can affect preference judgments and become targets of alignment optimization (Zhang et al., 2025b). This is evidence for separating substance from presentation, not for deleting every heading or list. Text-format effects also do not directly establish a cause of graphical interface decoration. A cut reviewed only for polish or concision can inherit the evaluator’s biases. [MODERATE]

3.3 Sycophancy and reasoning overhead are separate

Sharma et al. (2024) found that assistants and preference judgments can favor agreement with user beliefs over truth. Applying this concern to agreeable editing feedback is an inference; the study does not show that sycophancy causes feature bloat. In warmth-training experiments, responses became shorter on average while accuracy declined and sycophancy increased (Ibrahim et al., 2026). Approval seeking and verbosity are distinct axes.

Chen et al. (2025) document repeated solutions with little additional accuracy or diversity in studied long-reasoning models. This concerns computational and reasoning overhead, which differs from unnecessary user-facing content. It does not justify removing verification or explanation on difficult tasks. The surveyed evidence supports several bounded mechanisms rather than one explanation of excess across all media.

4 Measuring Excess

Record burden and preservation separately. A word count, changed-line count, or clutter score describes an artifact; it does not establish whether the user can complete the task. Record a reduction measure beside checks of required content, behavior, and access. Treat critical requirements as constraints, not quantities that a large reduction can offset. For positive draft count W_d and final count W_f , the descriptive removal rate is $100(1 - W_f/W_d)$ percent. A negative value records growth.

4.1 Prose and redundancy

SummEval evaluates relevance, consistency, coherence, and fluency separately. Its relevance rubric penalizes redundancy and excess information (Fabbri et al., 2021). ROUGE measures overlap with reference summaries, which cannot by itself certify factual support or usefulness (Lin, 2004; Maynez et al., 2020). The Pyramid method and QAPyramid assess coverage through content units and question–answer pairs (Nenkova and Passonneau, 2004; Zhang et al., 2025a). Adapting that idea, a studio can list the facts or actions its reader must recover before shortening a document. This adaptation is a proposed checklist, not a validated summarization metric for arbitrary reports.

SARI distinguishes adding, keeping, and deleting n-grams against reference simplifications; its deletion precision penalizes deletions unsupported by those references (Xu et al., 2016). It does not verify every semantic condition. Maximal Marginal Relevance balances relevance with novelty when selecting material (Carbonell and Goldstein, 1998). Both make a useful distinction between reducing repetition and discarding relevant content, but neither establishes the right length for a new brief.

4.2 Code size, complexity, and change scope

Lines, tokens, dependencies, and changed files measure different kinds of size. McCabe’s cyclomatic complexity concerns control-flow structure, so fewer physical lines need not imply simpler control flow (McCabe, 1976). Clone detection identifies duplicated code. Inconsistent clone changes can cause faults, while some duplication supports independent evolution (Juergens et al., 2009; Kapser and Godfrey, 2008). No one measure identifies every unnecessary abstraction or feature.

For local repairs, Zhu et al. (2026) compare token edit distance with a known minimal reversal of a controlled corruption. This exposes changes beyond that reference repair, with correctness evaluated separately. A different correct implementation can still have greater edit distance. The measure is informative when a narrow reference change exists; it is less decisive for an open-ended redesign.

4.3 Clutter, workload, and task performance

Feature congestion and related image measures predict aspects of perceived clutter and visual search difficulty (Rosenholtz et al., 2005, 2007). Their meaning depends on the display and task. Chu et al. (2025) found a nonlinear association between text-to-ink ratio and perceived complexity in static visualizations. Their observational, task-free results do not show that removing annotations improves interactive usability.

Cognitive load concerns a person doing a task. Subjective effort should be interpreted with performance (Paas et al., 2003). NASA-TLX separates mental, physical, and temporal demands, performance, effort, and frustration (Hart, 1986). A studio should name the instrument and scoring method it uses, and retain completion and error measures. Lower reported effort alone could accompany missing functionality. Table 1 proposes paired checks without combining them into a universal score.

5 Methods for Cutting

5.1 Specify scope and preservation

A brief should name the audience, task, required output, protected content or behavior, and exclusions. An optional word or change budget makes scope inspectable. If the budget conflicts with required content, resolve the conflict instead of silently dropping content. IFEval checks explicit constraints such as word counts and required terms; FollowBench tests

Table 1: Reduction signals and preservation checks for a studio audit. These pairings are proposed, not validated as a universal metric.

Artifact	Reduction signal	Preservation check
Prose	Words, repeated claims, irrelevant passages	Required content, factual support, usable explanation
Code	Changed tokens, files, branches, dependencies	Required behavior, edge cases, security obligations
Interface	Unused controls, clutter, interaction steps	Task completion, labels, status, error recovery
Design	Competing elements, redundant encodings	Message recognition, legibility, access to meaning

progressively added constraints (Zhou et al., 2023; Jiang et al., 2024). Neither benchmark establishes that stating a constraint guarantees compliance.

Pair exclusions with positive targets and checks. For example, specify the approved dependency and check that no new dependency appears. This is a practical proposal. Evidence that older models struggled with negated tasks concerns task inversions, not every ordinary prohibition (Jang et al., 2023).

The most direct preservation evidence comes from Zhu et al. (2026), a 2026 preprint on 400 controlled Python function repairs. Across 50 matched model settings, a preservation clause reduced mean excess normalized token edit distance from 0.195 to 0.131, with edit-fidelity metrics restricted to passing repairs. Mean Pass@1 over all tasks rose by about 2.3 percentage points. Both prompts already protected signatures, default arguments, and docstrings. The result supports explicit preservation in constrained repairs, with limited transfer to open-ended repository work. [MODERATE]

5.2 Use revision with informative feedback

Self-Refine improved outputs through feedback and revision on its studied tasks (Madaan et al., 2023). Intrinsic self-correction also failed or damaged initially correct reasoning in other experiments (Huang et al., 2024). The survey by Kamoi et al. (2024) emphasizes feedback sources and fair comparisons. These findings do not support a universal benefit from asking a model to reconsider.

CRITIC uses tools in its review loop; Self-Debugging uses code explanations and execution feedback in its studied settings (Gou et al., 2024; Chen et al., 2023). For a cut, choose feedback that can expose the property at risk, such as a behavior test, source passage, or observed user task. Record its coverage. An unsupported approval from another model is a judgment, not an independent preservation check. Section 7 turns this distinction into an editing procedure.

5.3 Keep code changes coherent and reviewable

A local patch exposes unrelated changes, but diff format does not guarantee correctness. Structure-aware and adaptive formats improved the accuracy–cost tradeoff in trained code-editing systems studied by Cheng et al. (2026). Those results do not establish that requesting a unified diff alone makes edits minimal or correct.

Google’s small-change and review guides are practitioner sources. They recommend self-contained changes and review of functionality, complexity, edge cases, tests, comments, and unnecessary generality (Google, n.d.b,n). They also recognize that a change can be too small to understand. This fits observational evidence that understanding a change is central to code review (Bacchelli and Bird, 2013). Preserve context that explains why behavior exists.

5.4 Reduce competing material without hiding the task

Nielsen’s practitioner heuristics combine relevant information with visible status, recognition, error prevention, recovery, and help (Nielsen, 1994). The National Archives practitioner checklist treats headings, lists, and whitespace as useful reading aids (National Archives and Records Administration, 2019). A design pass should ask what job each element serves. Secondary controls may be grouped or disclosed later, but the task must remain discoverable. Too little displayed information can add view switching (Rosenholtz et al., 2005). Test the revised task, and state any expressive goal before treating decoration as excess.

5.5 Change training incentives when possible

Length regularization, reward separation, desensitization, and counterfactual augmentation address particular length effects (Park et al., 2024; Shen et al., 2023; Liu et al., 2024; Kim et al., 2026). Their assumptions and access requirements differ; aggressive length desensitization can hurt performance. A studio using a hosted model can separate content value from length during evaluation without claiming to implement these training methods.

6 When Cutting Hurts

Claims lose their conditions. Removing a qualification, comparison group, or source can change a claim’s meaning. Summarization research documents fluent but unsupported content (Maynez et al., 2020); it does not establish that shortening itself causes hallucination. Check rewritten statements against their sources. Deletion and paraphrase expose different risks.

Explanations become too terse for their audience. Guidance that helps novices can become redundant for experienced learners (Kalyuga et al., 2003). Interesting but irrelevant details can impair learning, with effects moderated by instructional and methodological factors (Sundararajan and Adesope, 2020). These findings support audience-specific cuts. A definition, example, or reminder can perform a necessary job even when it repeats information.

Cleaner interfaces lose necessary cues. WCAG 2.2 specifies labels or instructions, error identification, and error prevention under stated conditions (World Wide Web Consortium, 2024). Its official labels guidance balances sufficient cues against excessive instruction (W3C Accessibility Guidelines Working Group, 2026). An accessible name does not replace every visible-label obligation. Preserve required input guidance and recovery paths; the applicable task and criterion determine whether a confirmation is necessary.

Small programs lose rare or security-relevant behavior. Stronger EvalPlus tests exposed errors missed by earlier code-generation tests (Liu et al., 2023). Early Copilot experiments found vulnerabilities in selected security scenarios (Pearce et al., 2022). Neither estimates failures caused by deletion, and the latter does not establish current-model prevalence. They support checking beyond common inputs. Preserve the security requirements, review, and testing obligations specified for the project, consistent with NIST’s SSDF (Souppaya et al., 2022).

Deduplication creates inappropriate coupling. Similar code may need to evolve independently, while inconsistent updates to clones can also cause faults (Kapsner and Godfrey, 2008; Juergens et al., 2009). A shared helper needs a shared concept and behavior. Eliminating every duplicate is not a sufficient design goal.

Reduction becomes the wrong target. Excessive length intervention and naive length corrections can degrade performance or reward truncation (Liu et al., 2024; Dubois et al., 2024). Stop when a further cut removes necessary content, introduces ambiguity, or cannot be checked. Keeping a longer version can be the correct result.

✦ ✦ ✦

7 The Cut Pass

This procedure combines the surveyed components into a proposed workflow. Its effectiveness across artifact types has not been tested. [LIMITED] It requires a saved baseline, a brief, an intended user, and checks for facts, behaviors, explanations, and access needs that must survive. Unresolved requirements remain protected. A size budget cannot override them.

Procedure 1 A reversible cut pass for an AI-built deliverable**Input:** Artifact, brief, intended user, preservation checks, optional size budget.**Output:** Accepted revision, cut ledger, and unresolved limitations.

- 1 Save the baseline and list required content, behavior, and access needs.
- 2 Measure burden and run preservation checks before editing.
- 3 Map each paragraph, component, helper, or element to a required purpose.
- 4 Nominate removals, merges, or moves; give each a reason and a specific check.
- 5 **for** each small, coherent edit batch **do**
- 6 Apply the edit and inspect the diff.
- 7 Repeat relevant checks and look for new omissions.
- 8 **if** a protected requirement fails or its preservation remains unresolved **then**
- 9 Restore or revise the batch; record the reason.
- 10 **else**
- 11 Accept the batch and record reduction and preservation evidence.
- 12 **end if**
- 13 **end for**
- 14 Compare the final artifact with the brief, including cross-component effects.
- 15 Stop when no justified, checkable cut remains; report counts and limits separately.

Choose candidates by purpose. Remove a unit whose purpose is absent before compressing its sentences or internals. Merge repeated claims only when their scope and evidence match. This ordering is a proposal, not a proven optimal search strategy. For each batch, ask whether it reduces an identified burden, retains protected obligations, and avoids new ambiguity, unsupported content, dependencies, or hidden actions. A failed critical check rejects the cut even if an aggregate score improves.

Reusable review instruction. Given this artifact, brief, and protected list, propose a small set of removals, merges, or moves. State the reason and preservation check for each. Keep conditions, edge cases, and access needs. Return the revision and a separate change summary. Restore any edit whose required property fails or remains unresolved. These words are a practical template, not experimentally optimized wording.

7.1 Findings

F1. Size does not establish task quality. Words, edit distances, and clutter describe different burdens; none alone certifies content, behavior, or usability (Fabbri et al., 2021; McCabe, 1976; Rosenholtz et al., 2007). [STRONG] The proposed audit therefore records reduction and preservation separately.

F2. State what must survive. Controlled repair evidence supports explicit preservation; instruction benchmarks show why stated constraints still need checks. Transfer to arbitrary studio work remains uncertain (Zhu et al., 2026; Zhou et al., 2023; Jiang et al., 2024). [MODERATE]

F3. Use feedback that can reveal failure. Self-review has mixed results. Tests and tool evidence can expose errors, but their coverage matters (Huang et al., 2024; Gou et al., 2024; Liu et al., 2023). [MODERATE]

F4. Retain necessary detail. Audience knowledge, access requirements, and maintenance needs affect whether explanations, labels, and duplication are expendable (Kalyuga et al., 2003; World Wide Web Consortium, 2024; Kapser and Godfrey, 2008). [MODERATE]

F5. Judge preference is insufficient evidence of value. Length and format can influence model judgments. Adjusting benchmark bias does not certify the user's task (Zheng et al., 2023; Dubois et al., 2024; Zhang et al., 2025b). [STRONG]

F6. The complete cut pass remains a proposal. Its components have support under different conditions. Self-application demonstrates an inspectable process, not effectiveness across domains. [LIMITED]

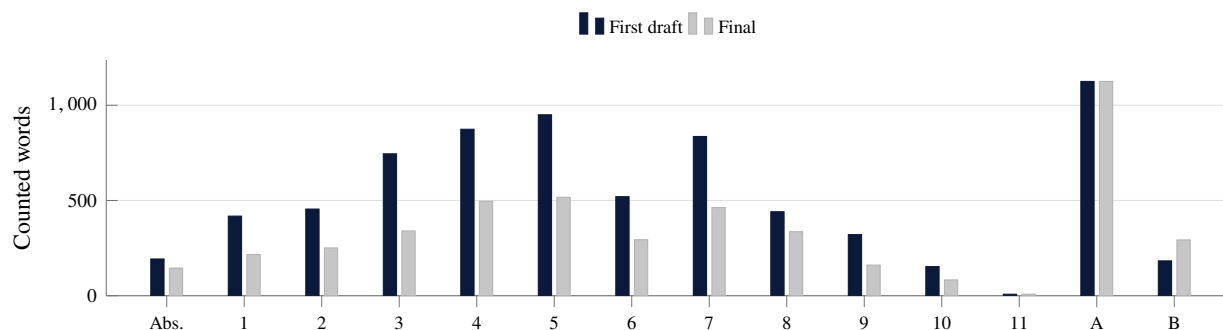


Figure 1: Draft and final counts by section. The abstract and Sections 1–11 form the core total. A and B are appendices, reported separately.

8 Applying the Method to This Paper

Baseline and protected requirements. We completed and saved `drafts/first-draft.tex` before editing, then recorded its hash and a copy of its bibliography. The preservation list covered the research question, five topic areas, source qualifications, survey method, executable procedure, audit trail, and disclosure placeholder. We cut each section against that list. The source audit records the support and limits of each included reference.

Accepted cuts. We merged repeated explanations of burden versus preservation, condensed sequences of paper summaries, and removed studio examples already represented in the measurement table. Operational advice now has one main home in the procedure. We kept counterexamples to simple brevity rules, including novice guidance, visible labels, legitimate duplication, and models that became less reliable despite shorter answers. The appendix retains every recorded search query.

A cut we reversed. The first cut retained SummEval’s four dimensions but removed the explanation that relevance penalizes redundancy and excess information. A separate review against the baseline and source identified that loss. We restored the link in Section 4, because a list of dimensions did not explain how excess was assessed (Fabbri et al., 2021). The same review restored the synthetic setting of the proxy-reward experiment and the nonlinear annotation-complexity finding. These are documented information losses, not measured reader failures. The frozen candidate and review record preserve the rejected cuts.

Count convention. The included script counts prose, captions, literal table text, run-in heads, and procedure text inside marked section regions. It excludes headings, citations, references, mathematics, URLs, and formatting commands. Hyphenated compounds count as one word. It also excludes the generated ledger, chart, and totals paragraph to avoid counting the numerical report inside its own measurement. The audit records these spans and source hashes. Appendices are reported separately, and added audit material can produce a negative removal percentage.

The core count changed from 5914 to 3310 words, a reduction of 2604 words (44.0%). Appendix counts changed from 1308 to 1418. Table 2 and Figure 1 are generated from the actual sources by `scripts/wordcount.py`.

What this demonstrates. Source checks, requirement review, and compilation can reveal specific failures. They do not establish unchanged reader comprehension, task performance, or perceived effort. This is one worked editorial case with a reproducible size measurement, not a controlled evaluation of the cut pass.

✦ ✦ ✦

9 Limitations

The evidence spans different tasks, models, measures, and audiences. Controlled repairs, summaries, learning materials, and static images only partly represent studio deliverables. Evidence for removing unrequested product features or over-engineered abstractions is thinner than evidence for length bias. Creative work also needs an explicit account of its expressive purpose.

Table 2: Cut Ledger. Counts follow the stated convention. Negative percentages denote growth; notes describe the principal edit, not every changed word.

Section	Draft	Final	Removed	Main change
Abstract	193	145	24.9%	Repeated catalogue and disciplinary list; retained scope limits.
1. Introduction	418	216	48.3%	Repeated distinctions and examples; retained definition and contributions.
2. Survey method	455	251	44.8%	Repeated scope caveats; access and version detail consolidated in A.
3. Causes	745	340	54.4%	Paper-by-paper repetition and unsupported explanations; restored study setting.
4. Measurement	874	496	43.2%	Repeated warnings and examples; restored rubric and nonlinear finding.
5. Cutting methods	950	517	45.6%	Repeated procedural advice; retained intervention evidence and boundaries.
6. Cutting harms	520	294	43.5%	Repeated rationale and stopping advice; retained distinct failure modes.
7. Cut pass	836	463	44.6%	Duplicated inputs, acceptance prose, and studio examples.
8. Self-application	441	336	23.8%	Prospective plans replaced by actual edits, restoration record, and counting rules.
9. Limitations	321	161	49.8%	Repeated source-level caveats; retained gaps and evaluation limits.
10. Conclusion	153	83	45.8%	Repeated framing and recommendations.
11. AI Disclosure	8	8	0.0%	Unchanged required placeholder.
Core total	5,914	3,310	44.0%	Abstract through AI Disclosure
<i>Appendices, reported separately</i>				
A. Search log	1,125	1,125	0.0%	All recorded queries and access notes retained.
B. Change summary	183	293	-60.1%	Added observed changes and rejected cuts in place of plans.
Appendix total	1,308	1,418	-8.4%	Excluded from core reduction

The search is selective and English-language. Search ranking, terminology, and accessible full texts affect coverage; newer preprints and older model studies have different uncertainties. Two sources used only accessible abstracts, so their treatment remains qualitative. The grades are judgments of claim support, not pooled effect sizes or formal risk-of-bias scores.

The same production process created and edited this paper, with a separate review pass but no blinded reader study. Counts depend on the published convention. Compilation and citation checks do not prove that all useful explanation survived. Review also costs time. The survey establishes no universal target length, deletion percentage, number of review rounds, or guarantee that the full workflow preserves quality.

10 Conclusion

Research supports several reasons to distrust length as a quality proxy, plus bounded methods for specifying preservation and checking revisions. It also explains why useful guidance, access cues, and behavior can be lost through indiscriminate cuts. The proposed cut pass joins these lessons in a reversible procedure. Its application to this paper makes the edits and counts inspectable while leaving reader outcomes unmeasured. A defensible stopping point is reached when no further cut has both a clear purpose and an adequate preservation check.

11 AI Disclosure

These research papers were ideated by the researchers at Stargazing Creativities Lab, generated by artificial intelligence, and human revised.

References

- Alberto Bacchelli and Christian Bird. Expectations, Outcomes, and Challenges of Modern Code Review. In *Proceedings of the International Conference on Software Engineering*, 2013. URL <https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/ICSE202013-codereview.pdf>.
- Jaime Carbonell and Jade Goldstein. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 335–336, 1998. URL https://www.cs.cmu.edu/~jgc/publication/The_Use_MMR_Diversity_Based_LTMIR_1998.pdf.
- Xingyu Chen, Jiahao Xu, Tian Liang, Zhiwei He, Jianhui Pang, Dian Yu, Linfeng Song, Qiuzhi Liu, Mengfei Zhou, Zhuosheng Zhang, Rui Wang, Zhaopeng Tu, Haitao Mi, and Dong Yu. Do NOT Think That Much for 2+3=? On the Overthinking of Long Reasoning Models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 9487–9499, 2025. URL <https://proceedings.mlr.press/v267/chen25bx.html>.
- Xinyun Chen, Maxwell Lin, Nathanael Schärli, and Denny Zhou. Teaching Large Language Models to Self-Debug. arXiv preprint arXiv:2304.05128, 2023. URL <https://arxiv.org/abs/2304.05128>.
- Wei Cheng, Yongchang Cao, Chen Shen, Binhua Li, Jue Chen, Yongbin Li, and Wei Hu. To Diff or Not to Diff? Structure-Aware and Adaptive Output Formats for Efficient LLM-based Code Editing. In *Findings of the Association for Computational Linguistics: ACL 2026*, pages 29677–29694, 2026. doi: 10.18653/v1/2026.findings-acl.1483. URL <https://aclanthology.org/2026.findings-acl.1483/>.
- Mengdi Chu, Zefeng Qiu, Meng Ling, Shuning Jiang, Robert S. Laramée, Michael Sedlmair, and Jian Chen. What makes a visualization image complex? arXiv:2510.08332v3, 2025. URL <https://arxiv.org/abs/2510.08332v3>. Accepted at IEEE VIS 2025.
- Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B. Hashimoto. Length-Controlled AlpacaEval: A Simple Way to Debias Automatic Evaluators. arXiv preprint arXiv:2404.04475, revised 2025, 2024. URL <https://arxiv.org/abs/2404.04475v2>.
- Alexander R. Fabbri, Wojciech Kryściński, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir Radev. SummEval: Re-evaluating summarization evaluation. *Transactions of the Association for Computational Linguistics*, 9:391–409, 2021. doi: 10.1162/tacl_a_00373. URL <https://aclanthology.org/2021.tacl-1.24/>.
- Leo Gao, John Schulman, and Jacob Hilton. Scaling Laws for Reward Model Overoptimization. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 10835–10866, 2023. URL <https://proceedings.mlr.press/v202/gao23h.html>.
- Google. What to look for in a code review. Google Engineering Practices Documentation, n.d.a. URL <https://google.github.io/eng-practices/review/reviewer/looking-for.html>. Practitioner source. Accessed 22 October 2026.
- Google. Small CLs. Google Engineering Practices Documentation, n.d.b. URL <https://google.github.io/eng-practices/review/developer/small-cls.html>. Practitioner source. Accessed 22 October 2026.
- Zhibin Gou, Zhihong Shao, Yeyun Gong, Yelong Shen, Yujiu Yang, Nan Duan, and Weizhu Chen. CRITIC: Large Language Models Can Self-Correct with Tool-Interactive Critiquing. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/fef126561bbf9d4467dbb8d27334b8fe-Abstract-Conference.html.
- Sandra G. Hart. NASA task load index (TLX): Paper and pencil package - volume 1.0. Technical Report NTRS 20000021488, NASA Ames Research Center, 1986. URL <https://ntrs.nasa.gov/citations/20000021488>.
- Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. Large Language Models Cannot Self-Correct Reasoning Yet. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/8b4add8b0aa8749d80a34ca5d941c355-Abstract-Conference.html.
- Lujain Ibrahim, Franziska Sofia Hafner, and Luc Rocher. Training language models to be warm can reduce accuracy and increase sycophancy. *Nature*, 652:1159–1165, 2026. doi: 10.1038/s41586-026-10410-0. URL <https://www.nature.com/articles/s41586-026-10410-0>.
- Joel Jang, Seonghyeon Ye, and Minjoon Seo. Can Large Language Models Truly Understand Prompts? A Case Study with Negated Prompts. In *Proceedings of The 1st Transfer Learning for Natural Language Processing Workshop*, volume 203, pages 52–62, 2023. URL <https://proceedings.mlr.press/v203/jang23a.html>.
- Yuxin Jiang, Yufei Wang, Xingshan Zeng, Wanjuan Zhong, Liangyou Li, Fei Mi, Lifeng Shang, Xin Jiang, Qun Liu, and Wei Wang. FollowBench: A Multi-level Fine-grained Constraints Following Benchmark for Large Language Models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4667–4688, 2024. doi: 10.18653/v1/2024.acl-long.257. URL <https://aclanthology.org/2024.acl-long.257/>.
- Elmar Juergens, Florian Deissenboeck, Benjamin Hummel, and Stefan Wagner. Do Code Clones Matter? In *Proceedings of the 31st International Conference on Software Engineering*, 2009. doi: 10.1109/ICSE.2009.5070547. URL <https://arxiv.org/abs/1701.05472>.
- Slava Kalyuga, Paul Ayres, Paul Chandler, and John Sweller. The expertise reversal effect. *Educational Psychologist*, 38(1):23–31, 2003. doi: 10.1207/S15326985EP3801_4. URL https://doi.org/10.1207/S15326985EP3801_4.

- Ryo Kamoi, Yusen Zhang, Nan Zhang, Jiawei Han, and Rui Zhang. When Can LLMs Actually Correct Their Own Mistakes? A Critical Survey of Self-Correction of LLMs. *Transactions of the Association for Computational Linguistics*, 12:1417–1440, 2024. doi: 10.1162/tacl_a_00713. URL <https://aclanthology.org/2024.tacl-1.78/>.
- Cory J. Kapser and Michael W. Godfrey. “Cloning considered harmful” considered harmful: patterns of cloning in software. *Empirical Software Engineering*, 13:645–692, 2008. doi: 10.1007/s10664-008-9076-6. URL <https://link.springer.com/article/10.1007/s10664-008-9076-6>.
- Hyeonji Kim, Sujeong Oh, and Sanghack Lee. Mitigating Length Bias in RLHF Through a Causal Lens. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(21):17517–17525, 2026. doi: 10.1609/aaai.v40i21.38806. URL <https://ojs.aaai.org/index.php/AAAI/article/view/38806>.
- Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, 2004. URL <https://aclanthology.org/W04-1013/>.
- Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. Is Your Code Generated by ChatGPT Really Correct? Rigorous Evaluation of Large Language Models for Code Generation. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-0943. URL https://papers.nips.cc/paper_files/paper/2023/hash/43e9d647ccd3e4b7b5baab53f0368686-Abstract-Conference.html.
- Wei Liu, Yang Bai, Chengcheng Han, Rongxiang Weng, Jun Xu, Xuezhi Cao, Jingang Wang, and Xunliang Cai. Length Desensitization in Direct Preference Optimization. arXiv preprint arXiv:2409.06411, 2024. URL <https://arxiv.org/abs/2409.06411>.
- Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegrefe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-Refine: Iterative Refinement with Self-Feedback. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-2019. URL https://papers.neurips.cc/paper_files/paper/2023/hash/91edff07232fb1b55a505a9e9f6c0ff3-Abstract-Conference.html.
- Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1906–1919, 2020. doi: 10.18653/v1/2020.acl-main.173. URL <https://aclanthology.org/2020.acl-main.173/>.
- Thomas J. McCabe. A Complexity Measure. *IEEE Transactions on Software Engineering*, SE-2(4):308–320, 1976. URL https://www.broy.in.tum.de/lehre/vorlesungen/vse/WS2004/1976_mccabet_complexity_measure.pdf.
- National Archives and Records Administration. Plain writing checklist, 2019. URL <https://www.archives.gov/open/plain-writing/checklist.html>. Practitioner guidance. Last reviewed July 2, 2019; accessed October 22, 2026.
- Ani Nenkova and Rebecca Passonneau. Evaluating content selection in summarization: The pyramid method. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, pages 145–152, 2004. URL <https://aclanthology.org/N04-1019/>.
- Jakob Nielsen. 10 usability heuristics for user interface design. Nielsen Norman Group, 1994. URL <https://www.nngroup.com/articles/ten-usability-heuristics/>. Practitioner guidance. Last reviewed January 30, 2024; accessed October 22, 2026.
- Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems 35*, 2022. URL <https://proceedings.neurips.cc/paper/2022/file/b1efde53be364a73914f58805a001731-Paper-Conference.pdf>.
- Fred Paas, Juhani E. Tuovinen, Huib Tabbers, and Pascal W. M. van Gerven. Cognitive load measurement as a means to advance cognitive load theory. *Educational Psychologist*, 38(1):63–71, 2003. doi: 10.1207/S15326985EP3801_8. URL https://doi.org/10.1207/S15326985EP3801_8.
- Ryan Park, Rafael Rafailov, Stefano Ermon, and Chelsea Finn. Disentangling Length from Quality in Direct Preference Optimization. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 4998–5017, 2024. doi: 10.18653/v1/2024.findings-acl.297. URL <https://aclanthology.org/2024.findings-acl.297/>.
- Hammond Pearce, Baleegh Ahmad, Benjamin Tan, Brendan Dolan-Gavitt, and Ramesh Karri. Asleep at the Keyboard? Assessing the Security of GitHub Copilot’s Code Contributions. In *IEEE Symposium on Security and Privacy*, 2022. URL <https://arxiv.org/abs/2108.09293>.
- Ruth Rosenholtz, Yuanzhen Li, Jonathan Mansfield, and Zhenlan Jin. Feature congestion: A measure of display clutter. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 761–770, 2005. URL <https://www.mit.edu/~yzli/clutter.pdf>.
- Ruth Rosenholtz, Yuanzhen Li, and Lisa Nakano. Measuring visual clutter. *Journal of Vision*, 7(2):17, 1–22, 2007. doi: 10.1167/7.2.17. URL <https://doi.org/10.1167/7.2.17>.
- Keita Saito, Akifumi Wachi, Koki Wataoka, and Youhei Akimoto. Verbosity Bias in Preference Labeling by Large Language Models. arXiv preprint arXiv:2310.10076, 2023. URL <https://arxiv.org/abs/2310.10076>.
- Mrinank Sharma, Meg Tong, Tomasz Korbak, David Duvenaud, Amanda Askell, Samuel R. Bowman, Newton Cheng, Esin Durmus, Zac Hatfield-Dodds, Scott R. Johnston, Shauna Kravec, Timothy Maxwell, Sam McCandlish, Kamal Ndousse, Oliver Rausch,

- Nicholas Schiefer, Da Yan, Miranda Zhang, and Ethan Perez. Towards Understanding Sycophancy in Language Models. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/0105f7972202c1d4fb817da9f21a9663-Paper-Conference.pdf.
- Wei Shen, Rui Zheng, Wenyu Zhan, Jun Zhao, Shihan Dou, Tao Gui, Qi Zhang, and Xuanjing Huang. Loose lips sink ships: Mitigating Length Bias in Reinforcement Learning from Human Feedback. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 2859–2873, 2023. doi: 10.18653/v1/2023.findings-emnlp.188. URL <https://aclanthology.org/2023.findings-emnlp.188/>.
- Prasann Singhal, Tanya Goyal, Jiacheng Xu, and Greg Durrett. A Long Way to Go: Investigating Length Correlations in RLHF. In *Conference on Language Modeling (COLM)*, 2024. URL <https://arxiv.org/abs/2310.03716>.
- Murugiah Souppaya, Karen Scarfone, and Donna Dodson. Secure Software Development Framework (SSDF) Version 1.1: Recommendations for Mitigating the Risk of Software Vulnerabilities. Technical report, NIST Special Publication 800-218, 2022. URL <https://csrc.nist.gov/pubs/sp/800/218/final>.
- NarayanKripa Sundararajan and Olusola Adesope. Keep it coherent: A meta-analysis of the seductive details effect. *Educational Psychology Review*, 32:707–734, 2020. doi: 10.1007/s10648-020-09522-4. URL <https://link.springer.com/article/10.1007/s10648-020-09522-4>.
- W3C Accessibility Guidelines Working Group. Understanding success criterion 3.3.2: Labels or instructions, 2026. URL <https://www.w3.org/WAI/WCAG22/Understanding/labels-or-instructions>. Official informative guidance, updated March 9, 2026; accessed October 22, 2026.
- World Wide Web Consortium. Web content accessibility guidelines (WCAG) 2.2. W3c recommendation, World Wide Web Consortium, 2024. URL <https://www.w3.org/TR/2024/REC-WCAG22-20241212/>. December 12, 2024 version.
- Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen, and Chris Callison-Burch. Optimizing statistical machine translation for text simplification. *Transactions of the Association for Computational Linguistics*, 4:401–415, 2016. doi: 10.1162/tacl_a_00107. URL <https://aclanthology.org/Q16-1029/>.
- Shiyue Zhang, David Wan, Arie Cattan, Ayal Klein, Ido Dagan, and Mohit Bansal. QAPyramid: Fine-grained evaluation of content selection for text summarization. In *Conference on Language Modeling*, 2025a. URL <https://arxiv.org/abs/2412.07096v2>. arXiv:2412.07096v2.
- Xuanchang Zhang, Wei Xiong, Lichang Chen, Tianyi Zhou, Heng Huang, and Tong Zhang. From Lists to Emojis: How Format Bias Affects Model Alignment. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 26940–26961, 2025b. doi: 10.18653/v1/2025.acl-long.1308. URL <https://aclanthology.org/2025.acl-long.1308/>.
- Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. In *Advances in Neural Information Processing Systems 36*, 2023. URL <https://arxiv.org/abs/2306.05685>.
- Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-Following Evaluation for Large Language Models. arXiv preprint arXiv:2311.07911, 2023. URL <https://arxiv.org/abs/2311.07911>.
- Tongyao Zhu, Wei Hern Lim, and Min-Yen Kan. When Models Edit Too Much: On the Fidelity of Minimal Code Edits. arXiv preprint arXiv:2609.04061v2, 2026. URL <https://arxiv.org/abs/2609.04061v2>. Version 2, revised 20 October 2026. Source reports acceptance to EMNLP 2026 Main.

A Search Log

All searches below were run on October 22, 2026. The table preserves the actual query strings, including provisional venue names that were corrected after opening the sources. S1 and S2 identify the web-search routes called `system1` and `system2`. Queries with a `site:` term restricted the requested domain; other queries were unrestricted. Only E19 used a recency filter, set to 730 days. Search rankings and indexing change, so rerunning these queries need not reproduce the same ordering or results.

Discovery was followed by direct retrieval from arXiv, ACL Anthology, PMLR, NeurIPS and ICLR proceedings, OpenReview, AAAI, publisher sites, author repositories, NASA, W3C, NIST, Google Engineering Practices, Nielsen Norman Group, and the National Archives. These are retrieval destinations, not a claim of exhaustive searches of each database. The three original machine-readable logs and the source-level verification record are in the project’s `audit/` directory.

Table 3: Complete discovery query log. All 61 queries were issued on October 22, 2026.

ID	Exact query	Source route
Causes and evaluation		
C01	LLM length bias reinforcement learning human feedback verbosity disentangling length arxiv	Web S2
C02	LLM as a judge verbosity bias length controlled AlpacaEval	Web S2
C03	Towards Understanding Sycophancy in Language Models paper	Web S2
C04	"A Long Way to Go: Investigating Length Correlations in RLHF"	Web S1
C05	"Disentangling Length from Quality in Direct Preference Optimization"	Web S1
C06	"Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena"	Web S1
C07	"Verbosity Bias in Preference Labeling by Large Language Models"	Web S1
C08	"Loose lips sink ships" "Length Bias"	Web S1
C09	"From Lists to Emojis: How Format Bias Affects Model Alignment"	Web S1
C10	"Length Desensitization in Direct Preference Optimization"	Web S1
C11	"Training language models to be warm can reduce accuracy and increase sycophancy"	Web S1
C12	"Scaling Laws for Reward Model Overoptimization"	Web S2
C13	"Training language models to follow instructions with human feedback" proceedings neurips	Web S2
C14	"The Illusion of Diminishing Returns" LLM length bias	Web S2
C15	"Do NOT Think That Much" "Overthinking"	Web S1
C16	"Not All Tokens Are What You Need" reasoning	Web S1
C17	"Disentangling Length from Quality" "2024" COLM	Web S1
C18	"Length Desensitization" proceedings ACL 2025	Web S1
C19	"Verbosity Bias in Preference Labeling" NeurIPS workshop	Web S1
C20	"Length-Controlled AlpacaEval" "COLM 2024" openreview	Web S2
Editing and software		
E01	Self-Refine Iterative Refinement with Self-Feedback Madaan 2023 NeurIPS	Web S2
E02	Large Language Models Cannot Self-Correct Reasoning Yet Huang ICLR 2024	Web S2
E03	Teaching Large Language Models to Self-Debug Chen ICLR 2024	Web S2
E04	When Can LLMs Actually Correct Their Own Mistakes A Critical Survey of Self-Correction of LLMs Kamoi	Web S2
E05	"Is Your Code Generated by ChatGPT Really Correct" NeurIPS 2023	Web S1
E06	"Instruction-Following Evaluation for Large Language Models" Zhou 2023	Web S1
E07	"FollowBench" "2024" fine-grained constraints	Web S1
E08	"Reflexion" "NeurIPS" 2023 Shinn	Web S1
E09	McCabe 1976 A Complexity Measure IEEE pdf	Web S2

Continued on next page

Table 3: Complete discovery query log (continued).

ID	Exact query	Source route
E10	Juergens Deissenboeck Hummel Wagner 2009 Do code clones matter pdf	Web S2
E11	Expectations Outcomes and Challenges of Modern Code Review Bacchelli Bird 2013	Web S2
E12	site:google.github.io eng-practices review small CLs complexity over engineering	Web S2
E13	Kapser Godfrey "Cloning considered harmful" 2008	Web S2
E14	Asleep at the Keyboard Assessing the Security of GitHub Copilot Generated Code 2022 authors	Web S2
E15	site:csrsrc.nist.gov SP 800-218 Secure Software Development Framework 2022	Web S2
E16	LLM negative constraints negation instructions do not empirical instruction following	Web S2
E17	"Can Large Language Models Truly Understand Prompts"	Web S1
E18	"Cloning considered harmful" site:link.springer.com	Web S1
E19	"LLM" "minimal" "diff" code editing evaluation 2025 2026	Web S1; 730-day filter
HCI and document measurement		
H01	Rosenholtz Li Nakano 2007 Measuring visual clutter Journal of Vision	Web S2
H02	Paas Tuovinen Tabbers van Gerven 2003 Cognitive Load Measurement means advance cognitive load theory	Web S2
H03	Fabbri SummEval 2021 re evaluating summarization evaluation ACL	Web S2
H04	Maynez Narayan Bohnet McDonald 2020 Faithfulness Factuality abstractive summarization	Web S2
H05	site.aclanthology.org Evaluating Content Selection Summarization Pyramid Method 2004	Web S1
H06	site.aclanthology.org ROUGE A Package for Automatic Evaluation of Summaries Lin 2004	Web S1
H07	site.nasa.gov NASA Task Load Index TLX Hart Staveland 1988	Web S1
H08	site.tandfonline.com Expertise Reversal Effect Kalyuga Ayres Chandler Sweller 2003	Web S1
H09	site.nngroup.com ten usability heuristics aesthetic minimalist design recognition rather than recall	Web S2
H10	site.digital.gov guidelines concise writing plain language unnecessary words	Web S2
H11	site.w3.org WCAG 2.2 labels instructions error prevention consistent identification	Web S2
H12	Sundararajan Adesope 2020 Keep It Coherent A Meta-Analysis of the Seductive Details Effect	Web S2
H13	"Measuring visual clutter" Rosenholtz pdf MIT	Web S1
H14	"The expertise reversal effect" 2003 pdf Kalyuga	Web S1
H15	"Cognitive Load Measurement as a Means" pdf Paas Tuovinen	Web S1
H16	"QAPyramid" ACL 2025	Web S1
H17	site.aclanthology.org "Optimizing Statistical Machine Translation for Text Simplification"	Web S2
H18	"The use of MMR" "Carbonell" "Goldstein" 1998	Web S2
H19	"Measuring visual clutter" filetype:pdf	Web S2
H20	"The Use of MMR" filetype:pdf site:cmu.edu	Web S1
H21	"Measuring visual clutter" filetype:pdf site:mit.edu	Web S1
H22	"Feature congestion" Rosenholtz 2005 761 770 DOI	Web S1

Selection. Sources were retained for a supportable claim about generation incentives, evaluation, constraints, editing, software review, measurement, or preservation risks. Primary research, reviews, standards, and official documentation were preferred. Practitioner guidance is labeled in the body. Duplicate coverage, secondary explainers, token-economy work without a deliverable-quality link, and unsupported generalizations were excluded. The inventory contains 51 distinct retained sources. No pooled effect estimate or exhaustive-screening count is claimed.

Access and verification. Some OpenReview pages required browser verification, and some publisher pages returned access errors. Proceedings, arXiv, or author copies supplied the relevant text. For Rosenholtz et al. (2007) and Kalyuga et al. (2003), author-uploaded manuscript headers corrected misleading repository landing metadata. The Sundararajan

and Adesope (2020) entry relies on the publisher abstract and metadata; no full-text effect size is used. An inaccessible 2025 expertise-reversal meta-analysis was not retained. Other exclusions and retrieval alternatives are recorded in the original logs.

Version discipline. The record distinguishes original preprint years from the opened versions. The NASA workload package is dated 1986 in official metadata despite the initial query seeking a 1988 chapter. The WCAG source is the December 2024 Recommendation; its labels explanation was updated in March 2026. QAPyramid was checked as a 2025 COLM version, and the visualization-complexity paper was checked against arXiv version 3. These corrections preserve the search history rather than silently rewriting the queries.

B Draft-to-Final Change Summary

The archived first draft is the complete pre-cut manuscript. The final paper is assembled from the section files. The ledger gives the numerical comparison; this appendix explains the editorial decisions.

Framing and evidence. The abstract and introduction lost repeated catalogues of excess and several restatements of the distinction between size and quality. The survey method retains the search date, inclusion rules, evidence grades, and narrative-survey status. Source access and version details remain in Appendix A. The causes section now groups mechanisms while retaining contrary findings and limits on causal interpretation.

Measurement, methods, and harms. Repeated warnings about single metrics were consolidated. The measurement table carries the domain examples. The methods section now separates supported interventions from the operational proposal. The harms section preserves distinct failure modes and drops repeated stopping advice. SARI is described as a reference-based n-gram metric, and code clones as duplicated code rather than automatically duplicated behavior.

Procedure and close. The procedure absorbs instructions previously repeated in inputs, acceptance prose, and worked examples. Its restoration branch and final cross-component review remain. Findings retain evidence grades. Limitations and conclusion no longer restate every source-specific caveat. The disclosure placeholder is unchanged. The search log retains all queries; self-application replaces prospective plans with completed edits and counting rules.

Rejected cuts. Review of the frozen cut candidate identified three losses. We restored the relevance rubric's treatment of redundancy, the synthetic setting of the reward-proxy experiment, and the nonlinear relation between annotation text and perceived complexity. The first was needed to explain a measurement; the others retained a study boundary and a specific finding. The record in `audit/candidate-review.json` contains exact before-cut and candidate passages. `audit/restorations.json` records the accepted repairs. This audit material adds words, which the appendix ledger reports rather than conceals.