Free reasoning, fixed record

16 Nov 2026

Work in progress. A preprint still in draft, published early so people can argue with it. Part of Pilots, not passengers — reconstructible reporting in evaluation.

Free reasoning, fixed record: reconstructible reporting when AI does the analysis

Steve Powell, Causal Map Ltd, Bath, United Kingdom (hello@causalmap.app)

Preprint, November 2026. Licence: CC BY 4.0.

For evaluators, commissioners of project evaluations and methodologists working with analyses of text carried out with AI.

Abstract

Frontier language models can now carry out much of the desk-based analysis of text in an evaluation. We set out an idea, in two halves, for how evaluators might respond. The first is *free reasoning*: given that current frontier LLMs already often beat any purpose-built analysis pipeline, or will do very soon, the model should be allowed to reason, and design each question's workflow, as it sees fit. The second half is the caveat: *the model must set out its working as auditable steps*, building steps that a lay reader can in principle follow; fixed rules are then applied to rebuild the report from those steps. "Reconstructible reporting" joins the two halves: any AI-generated analysis should come with a bundle of text and data files from which anyone, or any frontier model, can rebuild and rerun without any particular software. We also discuss what this set of basic steps might look like, without claiming that a single minimal set exists. We also ask how much of the desk work in well-known evaluation methods can be rebuilt using such steps. Methods that move between desk work and participation can be rebuilt only in their desk phases. A working proof of concept, Rubicon, shows that the idea can be built and used.

The premise

Our premise is that a frontier language model, given the terms of reference and the documents from an evaluation, can now do much of the desk-based part of a typical evaluation question such as reading a folder of interviews and reports, coding them in different ways, comparing groups and drafting a judgement according to a rubric. Until recently, the evidence for this kind of ability was mixed. Across 75 qualitative studies that used language models, agreement with human coders ranged from 36% to 99% (Kempny et al., 2026). In interviews with Rohingya refugees and their Bengali hosts, Ashwin et al. (2025) found that the errors made by language models in coding were "not random with respect to the characteristics of the interview subjects". More recently, Marston et al. (2026) compared 46 models with expert human coding of 150 synthetic humanitarian transcripts and found that "multiple LLMs can perform deductive coding at reliability levels comparable to experienced human coders", though "not as substitutes for human judgment".

The studies above are about coding, which is only one part of an evaluation's analysis, and most of them tested older models. Published studies of current frontier models answering whole evaluation questions from documents are not yet available, so rather than assert what such a model can now do, we report a test of our own. On 6 October 2026 we gave Claude Opus 5.5, running in Claude Code without any method guidance, the openly licensed transcripts of 13 interviews with people who had attended a "train the trainer" course in Making Every Contact Count (MECC), a programme that teaches health and care staff to use everyday contacts for brief conversations about health.[1] With them came the list of documents, which records who had gone on to train colleagues, and a question taken from the aims of the researchers who collected the interviews: what stopped attenders from delivering the training in their own organisations, and what helped the ones who did? The model answered in under two minutes. A separate Claude session, told nothing about where each text came from, then *checked every sentence of that answer, and of the results section of the researchers' published paper (Nichol et al., 2026), against the 13 transcripts*. It judged that the model's answer "holds up well", with almost every quotation word for word. It found 15 errors in about 120 sentences tested, counting each misreading once, of which 3 would change what a reader concludes. Checked the same way, *the original paper's results had a similar rate of errors*; more of them were more substantial.[2] Two judging models, from OpenAI and Google, each read the two texts in both orders and preferred the model's answer in all four judgements, on every criterion they scored. The comparison is perhaps tilted against the paper, which also drew on 6 more interviews and the free-text answers to 373 post-training surveys and was written for a journal rather than to answer one evaluator's question. Nor is it a verdict on the study. Its authors did what few researchers do and deposited their transcripts openly alongside their paper, which is excellent practice and the only reason such a test is possible. We have every reason to think they analysed them with care and skill. The test is also a single run of AI judging AI. The reviewer was Claude checking Claude's answer, the judges were other companies' models, and no person has checked the reviewer's error counts. Even so, the unaided model wrote an answer that a blind check found about as accurate as a peer-reviewed paper's results, and with fewer of the errors that matter. Evaluators using frontier models in the latter half of 2026 tell us that the results, not just in coding but in answering entire evaluation questions, often exceed their expectations, from Outcome Harvesting to causal mapping (for earlier AI-assisted work on each, see Britt et al., 2025; Powell

& Caldas Cabral, 2025). Meanwhile the models are improving faster than evaluation methods and tools can be redesigned around them.

Who, then, is going to decide what the analysis workflow should be and what counts as a good enough answer? Are evaluators and methodologists quickly going to lose control over relatively opaque AI powerhouses? How can a commissioner, or anybody else, check an analysis that a model has done, and will they even want to bother?

Free reasoning: do not try to out-design the model

The obvious response is to use the models to carry out a designed procedure: a chain of prompts for coding, a fixed sequence of stages for a thematic analysis, one step to draft a codebook and another to check it. Each such pipeline encodes an expert's view of how the analysis should go. But does this produce better results than what a frontier model would do just given the data and the question, and otherwise left to itself?

Sutton (2019) drew the general lesson from AI research: "general methods that leverage computation are ultimately the most effective, and by a large margin", and "building in how we think we think does not work in the long run". We take a narrower version for evaluation. Method knowledge built into a tool as a fixed procedure ages with the model it was designed around. Offered to a capable model as advice, the same knowledge may help, for a while. So we treat any deliberate, designed help with how to do the analysis as temporary, whether it is a pipeline that chains LLM-powered steps or guidance in the prompt on how to analyse. We test each such aid against the bare model, again with each new model, and drop it once the bare model does as well without it.[4] The exception is guidance on the basic steps from which the audit trail is built (see below).

Evaluation questions also differ in kind. A ranking, a question about contribution, a typology and a judgement against a rubric each need a different workflow. Evaluators have always adapted their methods as the work proceeds: Apgar et al. (2024) make "effective methodological bricolage" one of the three domains of their framework for rigour. Step-by-step recipes for a method help in teaching but are usually broken in practice. So the first half of our idea is to let the model reason as it sees fit, as a human expert would, and design its own workflow for each question.

The audit trail

We can ask the model to narrate what it did, but its account of its own work is only a plausible story about it. Chain-of-thought explanations "can systematically misrepresent the true reason for a model's prediction" (Turpin et al., 2023).

Ferretti (2026, p. 3) draws a parallel with autopilots in aviation. Researchers there described "automation surprises": moments when an aircraft behaved exactly as programmed but not as its pilots expected, so that they could no longer explain what it was doing or predict what it would do next (Sarter et al., 1997, as cited in Ferretti, 2026). The pilots' question at such moments was 'What's it doing now?' Ferretti puts the same question to evaluators: "If AI gives us a convincing interpretation, can we tell what assumptions and analytical choices shaped it?" For the reason just given, that question cannot be answered from the model's own account of its reasoning.

The second half of the idea therefore constrains the record and only to a lesser extent the reasoning. Whatever design the model chooses, it must set out its working as steps drawn from a small fixed vocabulary of steps, each with its own settings and output: for example, which documents were sampled, which passages were coded and with what, how the codes were grouped, what was counted out of what denominator, which rubric was applied and who signed off on it. The LLM's original report is kept, but fixed rules then rebuild the key parts of that report from that record.

Deterministic algorithms (not LLMs) recount every count from the coded passages, apply each rubric rule to its counts and combine the verdicts according to a declared rule, and find every quotation word for word in its source. A finding that rests on reading rather than counting is recorded with the passages for and against it, and the algorithms confirm that those passages exist. A report which cannot be reproduced from the record is refused.

A version of this pattern is used in machine-learning research: a deterministic solver executes a symbolic chain written by the model, so that the chain is "a faithful explanation of the final answer" (Lyu et al., 2023; see also Gao et al., 2023). Two things differ here. We choose the vocabulary for a lay reader. And the central operations in text analysis such as coding a passage are fundamentally non-deterministic. These are judgements, made by a model or a person and recorded as a step with its quotation, for later checking.

Design does not disappear under this arrangement, but it changes. The workflow for each question is still bespoke, but the model composes it from standard steps, so a reader who knows the steps can follow any design made from them. Either half alone is weaker: free reasoning alone cannot be checked, and a fixed pipeline designed by an expert is already being overtaken (we claim) by the bare model.

Reconstructible reporting

Joining up the two halves — that is what we call reconstructible reporting (perhaps there is a better word). An evaluation report based on an analysis of text comes with a bundle of data and text files containing what somebody else would need to rebuild the analysis and run it again: the sources; the workflow, with the settings each step was given; optionally the intermediate results produced by each step, such as coding tables with their quotations and counts with the documents behind them; the rubrics; and a record of how the run was carried out. The bundle is machine-readable, so a competent reader with time, or a frontier model in minutes, can rebuild and rerun the workflow without any particular required software.

A rerun will not always give the same results. That's why we talk about "reconstructibility" rather than "reproducibility". Text coding will never be deterministic or approach perfect itner-rater reliability. A step that branches on a threshold such as "more than half" can tip the other way when a count is close to it, and a model may read a passage differently a second time. What can be promised is reproducibility of the method, in a form that runs: findings in the same range as the first, open to comparison with

them. The smaller and more explicit the early steps, the better the chance that a rerun agrees.

The principle itself is old. Buckheit and Donoho (1995) distilled Claerbout's insight into the slogan that an article about computational science "is not the scholarship itself, it is merely advertising of the scholarship". Qualitative researchers are urged to keep an audit trail of "all key stages and theoretical, methodological, and analytical decisions" (Carcary, 2021). Moravcsik (2014) proposed linking each citation behind a contestable claim to an excerpt from the source, with a note on how the excerpt supports the claim. Powell (2019) set out a reproducible workflow for evaluation reports. Recent work comes closer. Ifargan et al. (2024) built a system whose quantitative papers can "programmatically chain results, methods and data"; Jeldtoft and Yousef (2026) combine "interpretative LLM inference with deterministic procedural control" in a thematic analysis that keeps "explicit links to the source material"; and Kohler et al. (2026) test whether agents can reproduce social-science results "given only a paper's methods description and original data".

What has changed is that rebuilding is now cheap. Following a workflow to the letter and reporting where it breaks used to take a colleague a fortnight, so commissioners took the method on trust; a model can now do it in minutes. Ferretti (2026, p. 2) warns that the better automation works, the less we check it, and asks: "After twenty convincing AI-generated interview syntheses, what would make us question the twenty-first?" When the rebuild is done by fixed rules, the twenty-first is recounted and rechecked exactly as the first was.

What we have not found elsewhere is the combination we present here for the analysis of text in evaluation: a model free to design the workflow using steps a lay reader can follow, and a report which can be rebuilt from the record by fixed rules, findings and quotations included. The principle applies equally to analysis done by people, with or without AI.

The jury test

Which steps, then? As a question of logic there is no answer. One can always invent another basic analytical operation.

A better question is which operations a lay person can follow. An untrained reader can follow a handful of operations without being taught anything first: count, compare, group, filter, total, weigh one thing against another. The list is limited by its readers, so it need not be lengthened for each new field. Juries are asked to do exactly this. A complicated case is explained to people who know nothing of the field, who must follow it and reach a verdict without first learning the discipline. We call the requirement that every step of an analysis be one such a reader could follow and check *the jury test*.[5] The reader we have in mind is the person an evaluation report goes to, assumed to bring ordinary sense, perhaps a bit of context. Contribution tracing already borrows the courtroom idea: Befani and Stedman-Bryce (2017) stage a symbolic "contribution trial", in which the evidence for and against a contribution claim is exposed to the scrutiny of a "jury".

By the same test, each step should be *the coarsest that a lay reader can still check*. Ssteps which are too fine could never be followed by any mortal. A step which is *too coarse* might contain a leap the reader might never catch, such as an unstated judgement.

The standard of worth stays with people, for example Where "adequate" begins and ends, which cannot be read off the material. King (2025): "Just because AI can reason algorithmically does not mean it should make evaluative judgements about human affairs." So the evaluator sets the standard, in advance and under their own name, usually as a rubric: criteria for what matters, with a description of each level of performance (King et al., 2013). Each level must say what it takes to reach it. "Every partner described a change they could date and name a cause for" is a standard; "green" or "passed" is a label. Where several criteria bear on one verdict, the rubric also says how they combine. The model then applies the rubric.

What are the basic building blocks, or steps?

Miles, Huberman and Saldaña (2014) list 13 tactics for drawing meaning from qualitative data, from noting patterns and counting to building a logical chain of evidence. That is a catalogue of what analysts do. Evaluation has a short list, the "logic of evaluation": define evaluative criteria, set standards of performance for them, measure performance, and "arrive at a judgment about performance based on synthesis of criteria, standards, and evidence" (Fournier, 1995, and Scriven, 1991, as cited in Montrosse-Moorhead, 2023, p. 124). We do not read either list as a basis from which every analysis follows. What each means in practice, such as what it is to group passages into a theme, is itself open to dispute. There is an empirical workaround: first, ask LLMs to answer different evaluation questions and note retrospectively what steps they use; then form a reasonable list of these and see if, given new and broad range of questions, an LLM can answer them using the chosen set of steps. Then iterate and improve.

Our own format uses eight kinds of step; others will choose differently, and overlapping lists are fine. The principle matters more than the precise list: whatever the steps are, a lay reader can follow each one, and the report can be rebuilt from them by fixed rules.

Desk work and participation

The largest caveat is that many evaluation methods are notpure desk analyses at all. They move between desk work and participatory input. Several such steps usually come before any document is read: agreeing the questions, the theory of change and the rubric with the commissioner and, where possible, with the people the programme affects. If nobody outside the evaluation team recognises a theory of change, testing it link by link maps only somebody's imagination. Human input also comes at the end. King et al. (2013, p. 15) develop rubrics with stakeholders and then present the findings to them for "collective sense making", rather than retreating to the "evaluator cave" to make the judgements. Following such a method properly means building the participatory steps into the workflow. The record then includes phases that cannot be carried out by any algorithm: the work pauses while people meet or think, then continues with what they decided, or starts again.

That makes the process as a whole non-deterministic. Only the desk phases can be rebuilt and rerun. The participatory phases can be

recorded (who took part, what they were shown, what they changed) but not rerun. A reader of the bundle can see which parts are which. The same is true of the drafts of a report that go back and forth with stakeholders.

An open format, and a working proof of concept

To make the idea concrete we have published an open format for the bundle, now at version 0.4 (Causal Map Ltd, 2026a).[6] Its workflows are built from eight kinds of step: sample, code, group, profile, tabulate, judge, claim and write. Each revision of a run is dated, with what changed in the workflow, the coding and the answer, so a reader can see whether a threshold was moved after the results came in. That follows the rule King et al. (2013, p. 17) give for rubrics: they can be changed as understanding evolves, "but not in order to avoid confronting findings". A published version of the format never changes.

We have also built a demonstration of how to implement this idea, as a proof of concept that works. Rubicon by Causal Map is a free plugin for Claude, released under the MIT licence, with its code on GitHub[7] (Causal Map Ltd, 2026b). The evaluator agrees the questions and any rubrics with the model before any document is read. The model then reasons as it likes, except that it has to write down *afterwards* the workflow it followed using the format's steps. Fixed rules then make the checks set out in the premise: they recount every count in the report from the coded passages, apply each rubric rule to its counts, find every quotation word for word in its document, and confirm that the passages each key finding is tagged with are in the record. If any check fails, no report is created. A sealed zip of the whole run is provided alongside the report. The report can be recreated from the zip byte for byte, deterministically. What a reader receives is best seen in the published example reports. Each is a single page that opens in a browser, in which each count can be clicked to show the judgementsn and passages behind it and any finding can be followed back through each step to the documents. One example deals with what households credit for changes in their health, with a causal map.[8] In another, the sustainability of a training programme is judged against a rubric.[9] A third is on reluctance to use AI at work.[10] We make no claims here about the accuracy of any of them. Together they are evidence that the arrangement described here can be built and used.

Is there already a standard for "reconstructible reporting"?

Guidance on AI in evaluation already asks that "technical documentation should enable professional peer review and methodological scrutiny" (UK Evaluation Society, 2025), yet in real life, even basic settings go unreported: of 75 qualitative studies that used language models, 56 reported no parameter settings at all (Kempny et al., 2026). When we surveyed existing standards in September 2026, and again in November, we did not find one that offers the combination set out under Reconstructible reporting.[11] Reporting checklists such as COREQ (Tong et al., 2007), now with an extension for studies using language models (Fehring et al., 2026), say what a report must describe, but a checklist is filled in post-hoc and cannot be run. REFI-QDA moves coded projects between qualitative software packages; its authors state that "the focus in developing the standard was only on exchange" (REFI-QDA, 2019). The Reproducible Open Coding Kit comes closest, as a plain-text standard for coded qualitative data that people and software can both read (Peters & Zörgő, 2026).

Limits and open questions

- Rebuilding cannot show that the design was a good one, nor how many designs were tried before this one. The second can be addressed by registering the workflow before reading the bulk of the data, at a cost to flexibility some evaluators will not want to pay.
- A bundle containing interview transcripts often cannot be published, and may go only to those entitled to the data.
- Not everything an analyst wants to do can be expressed with any given minimum set of steps.
- Under the jury test, expert inputs that a lay reader cannot check, such as the probabilities for Bayesian updating, enter only as declared judgements.
- Run-to-run variation remains
- participatory phases are recorded but not reproduced.

An invitation

We offer this as a position to argue with. If the premise is wrong, if the jury test is too strict or too loose, or if a better set of basic steps exists, we would like to hear it. What we hope remains after the argument is the core of the idea: let the model reason as it likes, but have it set out its working in steps that anyone can follow and from which the report can be rebuilt.

This paper was drafted with the assistance of Claude, an AI model from Anthropic; the arguments, judgements and any errors are our own.

References

Apgar, M., Bradburn, H., Rohrbach, L., Wingender, L., Cubillos Rodriguez, E., Baez-Silva Arias, A., Dioma, A., et al. (2024). Rethinking rigour to embrace complexity in peacebuilding evaluation. *Evaluation, 30*(3), 408–433. https://doi.org/10.1177/13563890241232405

Ashwin, J., Chhabra, A., & Rao, V. (2025). Using large language models for qualitative analysis can introduce serious bias. *Sociological Methods & Research, 55*(3), 795–839. https://doi.org/10.1177/00491241251338246

Baumann, J., Röttger, P., Urman, A., Wendsjö, A., Plaza-del-Arco, F. M., Gruber, J. B., & Hovy, D. (2025). *Large language model hacking: Quantifying the hidden risks of using LLMs for text annotation* (arXiv:2509.08825). arXiv. https://arxiv.org/abs/2509.08825

Befani, B., & Stedman-Bryce, G. (2017). Process tracing and Bayesian updating for impact evaluation. *Evaluation, 23*(1), 42–60. https://doi.org/10.1177/1356389016654584

Britt, H., Powell, S., & Caldas Cabral, G. (2025). *Strengthening Outcome Harvesting with AI-assisted causal mapping*. Causal Pathways Initiative. https://5a867cea-2d96-4383-acf1-7bc3d406cdeb.usrfiles.com/ugd/5a867c_ad000813c80747baa85c7bd5f3f044cd.pdf

Buckheit, J. B., & Donoho, D. L. (1995). WaveLab and reproducible research. In A. Antoniadis & G. Oppenheim (Eds.), *Wavelets and statistics* (Lecture Notes in Statistics, Vol. 103, pp. 55–81). Springer. https://doi.org/10.1007/978-1-4612-2544-7_5

Carcary, M. (2021). The research audit trail: Methodological guidance for application in practice. *Electronic Journal of Business Research Methods, 18*(2). https://doi.org/10.34190/JBRM.18.2.008

Causal Map Ltd. (2026a). *The Rubicon open format* (Version 0.4). https://app.causalmap.app/rubicon/spec/0.4/

Causal Map Ltd. (2026b). *Rubicon for Claude* [Computer software]. https://rubicon.causalmap.app/plugin

Collier, D. (2011). Understanding process tracing. *PS: Political Science & Politics, 44*(4), 823–830. https://doi.org/10.1017/S1049096511001429

Davies, R., & Dart, J. (2005). *The 'Most Significant Change' (MSC) technique: A guide to its use*. https://www.mande.co.uk/wp-content/uploads/2018/01/MSCGuide.pdf

Fehring, L., Frings, J., Rust, P., Kempny, C., Thürmann, P., Tamerler, E., & Meister, S. (2026). *Reporting of qualitative research using large language models (COREQ+LLM): A Delphi-based extension of the COREQ reporting guideline* [Preprint]. medRxiv. https://doi.org/10.64898/2026.10.01.26364479

Ferretti, S. (2026, October). *The autopilot lessons* [Discussion handout for a workshop at the European Evaluation Society conference]. https://duckonthemove.com/minisites/2026_10_EES_Workshop/FILES/Autopilot_Lessons.pdf

Gao, L., Madaan, A., Zhou, S., Alon, U., Liu, P., Yang, Y., Callan, J., & Neubig, G. (2023). *PAL: Program-aided language models* (arXiv:2211.10435). arXiv. https://arxiv.org/abs/2211.10435

Ifargan, T., Hafner, L., Kern, M., Alcalay, O., & Kishony, R. (2024). *Autonomous LLM-driven research from data to human-verifiable research papers* (arXiv:2404.17605). arXiv. https://arxiv.org/abs/2404.17605

Jeldtoft, N. J., & Yousef, T. (2026). *Designing an auditable LLM-supported workflow for qualitative thematic analysis* (arXiv:2608.30543). arXiv. https://arxiv.org/abs/2608.30543

Kempny, C., Frings, J., Rust, P., Meister, S., & Fehring, L. (2026). The use and methodological reporting of large language models in qualitative research: A scoping review. *BMC Medical Research Methodology, 26*, 137. https://doi.org/10.1186/s12874-026-02913-1

King, J. (2025, October 6). AI-enhanced evaluation. *Evaluation and Value for Investment*. https://juliankingnz.substack.com/p/ai-enhanced-evaluation

King, J., McKegg, K., Oakden, J., & Wehipeihana, N. (2013). Evaluative rubrics: A method for surfacing values and improving the credibility of evaluation. *Journal of MultiDisciplinary Evaluation, 9*(21), 11–20.

Kohler, B., Zollikofer, D., Einsiedler, J., Hoyle, A., & Ash, E. (2026). *Read the paper, write the code: Agentic reproduction of social science results* (arXiv:2604.21965). arXiv. https://arxiv.org/abs/2604.21965

Lyu, Q., Havaldar, S., Stein, A., Zhang, L., Rao, D., Wong, E., Apidianaki, M., & Callison-Burch, C. (2023). *Faithful chain-of-thought reasoning* (arXiv:2301.13379). arXiv. https://arxiv.org/abs/2301.13379

Marston, J., Kreutzer, T., Garnier, S., Boone, E., Pham, P. N., & Vinck, P. (2026). *Can large language models reliably code qualitative humanitarian data? A benchmark study against human expert adjudication* (arXiv:2606.26541). arXiv. https://arxiv.org/abs/2606.26541

Mayne, J. (2001). Addressing attribution through contribution analysis: Using performance measures sensibly. *Canadian Journal of Program Evaluation, 16*(1), 1–24. https://doi.org/10.3138/cjpe.016.001

Mayne, J. (2008). *Contribution analysis: An approach to exploring cause and effect* (ILAC Brief 16). Institutional Learning and Change Initiative.

Miles, M. B., Huberman, A. M., & Saldaña, J. (2014). *Qualitative data analysis: A methods sourcebook* (3rd ed.). Sage.

Montrosse-Moorhead, B. (2023). Evaluation criteria for artificial intelligence. *New Directions for Evaluation, 2023*(178–179), 123–134. https://doi.org/10.1002/ev.20566

Moravcsik, A. (2014). Transparency: The revolution in qualitative research. *PS: Political Science & Politics, 47*(1), 48–53. https://doi.org/10.1017/S1049096513001789

Nichol, B., Rodrigues, A. M., Audsley, S., Haste, A., Tang, M. Y., Aston, C., Hansen, J., & Haighton, C. (2026). Exploring the 'train the trainer' model for delivering Making Every Contact Count (MECC) training at scale: A qualitative study. *British Journal of Health Psychology, 31*(2). https://doi.org/10.1111/bjhp.70076

Pawson, R., & Tilley, N. (1997). *Realistic evaluation*. Sage.

Peters, G.-J. Y., & Zörgő, S. (2026). *The Reproducible Open Coding Kit: A human and machine readable standard for coding qualitative data* [Preprint]. PsyArXiv. https://doi.org/10.31234/osf.io/4q8bs_v2

Powell, S. (2019). A reproducible workflow for evaluation reports. *The Evaluator*, Spring.

Powell, S., & Caldas Cabral, G. (2025). AI-assisted causal mapping: A validation study. *International Journal of Social Research Methodology*, 1–20. https://doi.org/10.1080/13645579.2025.2591157

REFI-QDA. (2019). *REFI-QDA: Exchange of processed data between qualitative data analysis software packages* (F. van Blommestein, Ed.). https://www.qdasoftware.org/

Sutton, R. (2019, March 13). *The bitter lesson*. http://www.incompleteideas.net/IncIdeas/BitterLesson.html

Tong, A., Sainsbury, P., & Craig, J. (2007). Consolidated criteria for reporting qualitative research (COREQ): A 32-item checklist for interviews and focus groups. *International Journal for Quality in Health Care, 19*(6), 349–357. https://doi.org/10.1093/intqhc/mzm042

Turpin, M., Michael, J., Perez, E., & Bowman, S. R. (2023). *Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting* (arXiv:2305.04388). arXiv. https://arxiv.org/abs/2305.04388

UK Evaluation Society. (2025). *AI in evaluation: Good practice guidelines for practitioners* (Version 1.0). https://evaluation.org.uk/wp-content/uploads/2025/11/UK-Evaluation-Society-AI-in-Evaluation-Good-Practice-Guidelines-for-Practitioners-V-2.0-1125.pdf

Wilson-Grau, R., & Britt, H. (2012). *Outcome harvesting* (Revised November 2013). Ford Foundation Middle East and North Africa Office.

Wong, G., Westhorp, G., Manzano, A., Greenhalgh, J., Jagosh, J., & Greenhalgh, T. (2016). RAMESES II reporting standards for realist evaluations. *BMC Medicine, 14*, 96. https://doi.org/10.1186/s12916-016-0643-1

1. The transcripts are in the UK Data Service's ReShare repository under CC BY 4.0: https://doi.org/10.5255/UKDA-SN-857461 ↩

2. The model's answer, unedited, with both reviews and the judges' verdicts: https://rubicon.causalmap.app/assets/examples/mecc-plain/ ↩

3. https://rubicon.causalmap.app/assets/examples/combined-report-plain/ ↩

4. The principles we work to when AI does some or all of an analysis of text, this one among them: https://garden.causalmap.app/principles/ ↩

5. A longer version of the argument: https://garden.causalmap.app/rubicon-jury/ ↩

6. A longer account of the bundle, the format and what a rerun can show: https://garden.causalmap.app/open-format/ ↩

7. https://github.com/stevepowell99/causalmap-plugins ↩

8. https://rubicon.causalmap.app/assets/examples/quip-health/ ↩

9. https://rubicon.causalmap.app/assets/examples/mecc-sustainability/. It draws on the same MECC transcripts as the test described in the premise. ↩

10. https://rubicon.causalmap.app/assets/examples/ai-at-work-mcp/ ↩

11. What we looked at, and what we could not verify: https://garden.causalmap.app/rubicon-standards/ ↩