🌻 Minimalist coding does not help with blockers and enablers

Thanks for inviting me to the discussion, James. **I'll start by describing the "minimalist" approach** to coding causal statements used for QuIP and developed originally by James, Fiona and colleagues at BathSDR and developed and further formalised at Causal Map Ltd in collaboration with BathSDR. This formalisation lives inside the Causal Map app. Then I will try to **answer the question of whether it can help us deal with more complicated constructions like enabling and blocking** and whether this could help us with mid-range theory. As an appendix I'll add a more detailed overview of minimalist causal coding.

The minimalist approach is notable because it is based in our **joint experience of coding thousands and thousands of stakeholder interviews and other data such as project reports**, mostly from international development and related sectors, as well as coding hundreds of thousands of pages with AI-assisted coding. These have nearly always involved **multiple sources talking about at least partially overlapping subject matter**. So this coding produces individual causal maps for each source, which can then be combined in various ways -- rather than constructing single-source maps of expert thinking (Axelrod, 1976) or the collective construction of a consensus map (Barbrook-Johnson & Penn, 2022).

Our experience has been that the vast majority of causal claims in these kinds of texts are easily and satisfactorily coded in the simplest possible form "X causally influenced Y". Explicit invocation of concepts like enabling/blocking, or necessary and/or sufficient conditions, or linear or even non-linear functions, or packages of causes, or even the strength of a link, are relatively rare. The causes and effects are not conceived of as variables, the causal link is undifferentiated, without even polarity, and if any counterfactual is implied it remains very unclear.

This approach is what we call **"Minimalist" or "Barefoot" Coding**.

So what? Can we use minimalist coding to code and make deductions about, say, enablers and blockers?

Using more sophisticated, non-minimalist coding such as DAGs or fuzzy cognitive maps or whatever allows one to code linear or even non-linear causal influences of single or even multiple causes on their effects. One can do the "coding" by simply writing down (using appropriate special syntax) the connections, because one is an expert, and/or one can verify such statements statistically on the basis of observational data. Thus armed, one can make predictions or have sophisticated arguments about counterfactuals. But using minimalist coding we cannot do that, because our claims are formally weaker and therefore our inference rules are weaker. What we *can* do is still really interesting. We can ask and answer useful questions like:

- what are the main influences on (or effects of) a particular factor, according to the sources?
- what are the upstream, indirect influences on (or effects of) a particular factor, bearing in mind The transitivity trap?
- how well is a given programme theory validated by the respondents' narratives? (We can do this basically by using embeddings to get measures of semantic similarity between labels and aggregate these as a goodness of fit of theory to data.)

That is all exciting and useful. It's a surprisingly simple way to make a lot of sense out of a lot of texts which is, with caveats, almost completely automatable, **but** James suggests that maybe we could ascend from formally weaker but numerically overwhelming minimalist-coded data to make other rich conclusions, in particular about enablers and blockers like the headphones and the rain. However, I don't think this is really possible. In minimalist coding, at the level of individual claims, you can code "The headphones enabled James to answer the question in the Zoom call" as

> The headphones --> James was able to answer the question in the Zoom call

... but we cannot easily get inside the *contents* of the effect. We might like to be able to code this as the effect of the headphones not on a simple causal factor but on *another causal connection*, namely between the question on the Zoom call and James' answer, but we do not have any way at the moment to do this. It might be possible to extend minimalist coding to cope with this, perhaps ending up with three factors (headphones, question, answer) and some new syntactic rules to code their relationship, and some corresponding new semantic rules to be able to deduce more things about these three factors, but I think this would be **missing the point**. I'm not sure what we could do with these kinds of subtle relationships at any scale. Let's guess that within a given corpus, five percent of causal claims are of this form: what are the chances of such claims then overlapping enough in content that we could then apply our new more specialised deduction rules in more than a handful of cases?

It might be the case that certain specific more sophisticated causal constructions become **part of ordinary language**. For example: "Her post mocking Farage went viral, so Farage was forced to respond". Here, the concept of *going viral* is perhaps a kind of shorthand for a quite sophisticated causal claim, yet it might be common enough for us to be able to usefully code it (and reason with it) using only unadulterated minimalist coding, without causally unpacking "her post went viral".

So that's useful, and maybe it is even useful in building some kinds of mid-range theory, but without actually understanding or unpacking what "going viral" means.

So that's it, in a nutshell. Sorry to disappoint, James.

Appendix: Minimalist coding

The 90% rule

We have found that it is pretty easy to agree how to apply minimalist coding to say 90% of explicit causal claims in texts, without missing out essential causal information, whereas it is very difficult to find appropriate frameworks to cope with the remaining 10%.

Fewest assumptions

Minimalist coding is perhaps the most primitive possible form of causal coding which makes no assumptions about the ontological form of the causal factors involved (the "causes" and "effects") or about *how* causes influence effects. In particular we do not have to decide if cause and/or effect is perhaps Boolean or ordinal, or if perhaps multiple causes belong in some kind of package or if there is some kind of specific functional relationship between causes and effects.

An act of causal coding is simply adding a link to a database or list of links: a link consists of **one new or reused cause label and one new or reused effect label**, together with the highlighted quote and the ID of the source.

A statement (S) from source Steve:

> I drank a lot and so I got super happy
>
> can be trivially coded minimalist-style as
>
> I drank a lot --> I got super happy (Source ID: Steve; Quote: I drank a lot and so I got super happy)

That's it.

Causal maps

Crucially, we can then display the coded claims for individuals as a graphical causal map, and we can also display the entire map for all individuals and/or maps filtered in different ways to answer different questions. There is a handful of other applications (Ackermann et al., 1996) (Laukkanen, 2012) for causal mapping which also do this; but as far as we know, only Causal Map also allows direct QDA-style causal coding of texts.

Data structure

Although we have the option of creating additional tags associated with each link (where many approaches would for example code the polarity of a link) this is not central to our approach.

We don't use a separate native table for factor labels: they are simply derived on the fly from whatever labels happen to be used in the current table of links. This makes data processing simpler and also suggests an ontological stance: causal factors only exist in virtue of being part of causal claims.

We do however have an additional table for source metadata including the IDs of sources, which can be joined to the links table in order, for example, to be able to say "show me all the claims made by women".

Causal powers

We adopt an explicitly realist understanding of causation, because we think that's what people mean. The outcome occurred in virtue of the causal powers: drinking a lot causally influenced the super happiness in virtue of its causal powers to do so; that's what makes it a causal claim rather than just a remark on a co-occurrence or a sequence of events.

Causal influence, not determination

We believe that it's rare for people to make claims about causal determination: someone can say that the heavy drinking made them super happy and then also agree that the music had a lot to do with it too, without this feeling like a contradiction.

Not even polarity

We differ even from most other approaches which are explicitly called "causal mapping" in that we do not even "out of the box" record polarity of links (to do so would involve making assumptions about the nature of the "variables" at each end of the link as well the function from one to the other).

The Focus on Cognition

In the minimalist approach, **we are quite clear that what we are trying to code is the speaker's surface cognitions and causal thinking**, while the actual reality of the things themselves is simply bracketed off at this stage, either to be revisited later (because we are indeed interested in the facts beyond the claims) or not (because we are anyway interested in the cognitions).

Staying on the surface

At Causal Map, we rarely make any effort to get beneath the surface, to try to infer hidden or implicit meanings. This is particularly well-suited to coding at scale and/or with AI. Our colleagues at BathSDR do this a bit differently, spending more effort to read across an entire source to work out what the source *really* meant to say.

Closer to the cognitive truth

It's really easy to code statements like (S) using minimalist coding. The trouble with trying to use more sophisticated frameworks is that they are nearly always *ontologically under-determined*. For example, even a simple approach like Causal Loop Diagramming is strongly functional and requires at least a monotonic relationship between the variables: something like, *the more* I drink, *the happier* I get (in addition to which we have to code the actual, factual claims: I did drink a lot, and: I did get super happy). But is that what the speaker meant? How do we know if the speaker has say a continuous or Boolean model of "drinking"? If Boolean, what is the opposite of drinking a lot? Drinking only a little? If continuous, how do we know what kind of function they use in their own internal model?

We'd say: nonsense. To code most causal claims as meaning some functional relationship between variables is mostly over-specified and psychologically wrong. Trying to apply such non-minimalist models means that even the trivially easy 90% of causal coding becomes suddenly hard. Of course, you can just declare that we are going to use a particular kind of non-minimalist coding for everything, but which? If we code "I got really tired because I have Long Covid", we could perhaps code both cause and effect as Boolean variables, but what about "I got really tired because it was really hot", and "I got really tired because it was really cold" how are we going to code "it was really hot" and "it was really cold"? Is there a moderate temperature which does not have this effect? How moderate? Does this variable pass through zero and come out the other side into minus temperatures? ((Ragin, 2008)) If what want to do is model a system, we can pick any solution we want. But if we want to model *cognition*, any of these answers is usually over-specified.

Unclear counterfactuals

More formal, non-minimalist coding has clear counterfactuals. These may often be Boolean or continuous (the volume depends on the position of the volume dial; it's a 10, so the volume is maximum, if it had been at 5, the volume would have been about half as loud, and so on). Minimalist coding arguably implies some kind of naked counterfactual, but it is not always clear exactly *what*.

General versus specific

Minimalist coding focuses primarily on **factual causal claims** which also warrant the inference that both X and Y actually happened / were the case.

Most causal claims in the kinds of texts we have dealt with (interviews and published or internal reports in international development and some other sectors) are factual, about the present or past. Sometimes we see general claims, and we often just code these willy-nilly. In any case, the distinction between general claims and claims about specific events that actually happen is often fractal and difficult to maintain completely when modelling ordinary language.

Minimalist coding as "qualitative causal" coding

Minimalist coding may be reasonably also called **Qualitative Causal Coding**. It shares characteristics with some forms of coding within Qualitative Data Analysis (QDA), in particular demonstrating an asymmetry between presence and absence.

We don't code absences

We do not code absences unless they are specified within the text. While codes may be counted, the concept of a *proportion* of codes is challenging because the denominator is often unclear. So if families are talking about reasons for family disputes, and family F mentions social media use, and family G mentions homework, we do not usually assume that family F does *not* think that homework can also be a cause of family disputes.

The labels do all the work

At Causal Map Ltd, our canonical methodology initially involves in vivo coding, using the actual words in the text as factor labels. This initial process generates hundreds of overlapping factor labels. This part is really easy (and is easy to automate with AI). Obviously, hundreds (or hundreds of thousands) of overlapping factor labels are not very useful, so we need to somehow consolidate them. Arguably, minimalist coding makes the initial coding easy but it just defers some of the challenges to the recoding phase. We can:

- Use human- or AI-powered clustering techniques to consolidate the codes according to some theory
- Use AI-powered clustering techniques to consolidate the codes according to automated, numerical encoding of their meanings
- "Hard-recode" the entire dataset using a newly agreed codebook (see above)
- "Soft-recode" the dataset on the fly using embeddings to recode raw labels into those codebook labels to which they are most similar

None of this really answers all the questions raised above about problematic cases such as what to do with "I got really tired because it was really hot", and "I got really tired because it was really cold" or any other case where we have different factor codes which have shared information. At first blush, this isn't a problem, we can simply code "it was really hot" and "it was really cold" separately, but how to parse the contents to reflect the fact that these two are related? Or, how to parse the contents of "Improved health behaviour (hand washing)" and "Improved health behaviour (using a mosquito net)" to reflect the fact that they are somehow neighbours? We do have some tricks for this, but that would take us beyond the present discussion.

See also:

(Powell et al., 2024)

(Powell & Cabral, 2025)

(Britt et al., 2025)

(Powell et al., 2025)

(Remnant et al., 2025)

References

Ackermann, Jones, Sweeney, & Eden (1996). *Decision Explorer: User Guide.* https://banxia.com/pdf/de/DEGuide.pdf.

Axelrod (1976). *Structure of Decision: The Cognitive Maps of Political Elites.* Princeton university press.

Barbrook-Johnson, & Penn (2022). *Participatory Systems Mapping*. In *Systems Mapping: How to Build and Use Causal Models of Systems*. https://doi.org/10.1007/978-3-031-01919-7_5.

Britt, Powell, & Cabral (2025). *Strengthening Outcome Harvesting with AI-assisted Causal Mapping*. https://5a867cea-2d96-4383-acf1-7bc3d406cdeb.usrfiles.com/ugd/5a867c_ad000813c80747baa85c7bd5ffaf0442.pdf.

Laukkanen (2012). *Comparative Causal Mapping and CMAP3 Software in Qualitative Studies*. https://doi.org/10.17169/fqs-13.2.1846.

Powell, Copestake, & Remnant (2024). *Causal Mapping for Evaluators*. https://doi.org/10.1177/13563890231196601.

Powell, & Cabral (2025). *AI-assisted Causal Mapping: A Validation Study*. Routledge. https://www.tandfonline.com/doi/abs/10.1080/13645579.2025.2591157.

Powell, Cabral, & Mishan (2025). *A Workflow for Collecting and Understanding Stories at Scale, Supported by Artificial Intelligence*. SAGE PublicationsSage UK: London, England. https://doi.org/10.1177/13563890251328640.

Ragin (2008). *Measurement Versus Calibration: A Set-Theoretic Approach*. https://doi.org/10.1093/oxfordhb/9780199286546.003.0008.

Remnant, Copestake, Powell, & Channon (2025). *Qualitative Causal Mapping in Evaluations*. In *Handbook of Health Services Evaluation: Theories, Methods and Innovative Practices*. https://doi.org/10.1007/978-3-031-87869-5_12.