JOHAN LINÅKER

Qualitative Data Analysis

- Part of Theory and methods of Software Engineering

About making sense of the data

- A systematic approach to analyse qualitative content
- The goal is to make sense and meaning of the studied phenomena,
- Translating raw data and observations into abstract interpretations and generalisations
- In the end, creating knowledge that answers the defined research questions.
- "*Quantitative analysis calculates the mean. Qualitative analysis calculates meaning.*" – *Saldana, 2016*

Photo by Júnior Ferreira | https://unsplash.com/photos/person-catching-light-bulb-7esRPTt38nI

Same thing, different name

- Thematic analysis, Content analysis, Qualitative Data Analysis, or plain coding…
  - in essence, the same thing!
- General (and highly iterative) process
  - Processing and structuring of data – preparing the data for analysis, and the combination and cross-analysis with additional data-points
  - First-cycle coding– adding initial codes (tags and labels) to the data, creating the initial links and understanding of the studied phenomena
  - Second-cycle coding– identification of themes (grouping of codes into categories and sub-categories) making generalizations across the data points
  - Conceptualization and theorization – synthesising the main conclusions and knowledge statements based on the identified themes and strands of the data , connecting to the research questions

RI.
SE

Define your strategy up-front

- The choices you make up-front will define your analysis process down the road. Better define the strategy yourself, but
  - acknowledge that it will most probably change!
- Ensures that
  - The right data is collected
  - That the data is processed and structured correctly
  - That the necessary meta-data is available
  - …
  - That you can answer your research questions!

Photo by JESHOOTS.COM | https://unsplash.com/photos/depth-of-field-photography-of-man-playing-chess-fzOITuS1DIQ

Aligning with goal of the study

- Are you exploring a new phenomena?
- Does it build on top of previous work?
- Are you validating previous work, or any hypotheses coming out of that work?
- Are you extending and building on top of that work?

Photo by Gerrie van der Walt | https://unsplash.com/photos/round-black-decor-Cx4PzId7LUU

{De | In | Ab}ductive coding

- Aligning with the goal of the study, you need to decide how informed you are (or want to be)? How do you relate to extant knowledge?
- Different ends on a spectrum
- Being informed by literature and initial hypotheses predefines a general set of themes or assumptions to validate or explore
- Focusing on the empirical data allows for an open mind and interpretation
- Typically, you will end up moving between the two

Photo by Susan Q Yin | https://unsplash.com/photos/red-and-blue-arrow-sign-surrounded-by-brown-trees-BiWM-utpVVc

{De | In | Ab}ductive coding

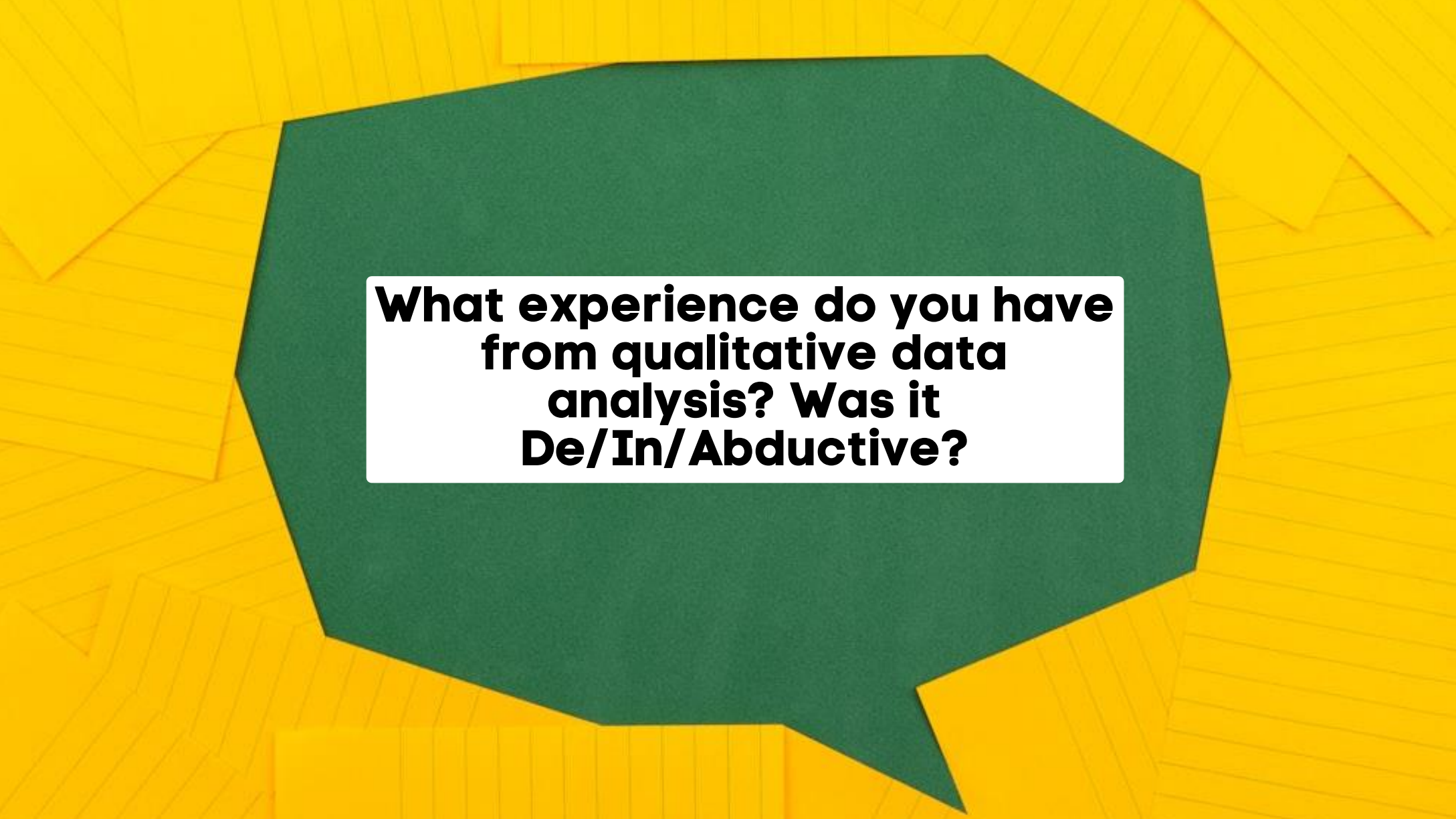
- Aligning with the goal of the study, you need to decide how informed you are (or want to be)? How do you relate to extant knowledge?
- Different ends on a spectrum
- Being informed by literature and initial hypotheses predefines a general set of themes or assumptions to validate or explore
- Focusing on the empirical data allows for an open mind and interpretation
- Typically, you will end up moving between the two

Photo by Susan Q Yin | https://unsplash.com/photos/red-and-blue-arrow-sign-surrounded-by-brown-trees-BiWM-utpVVc

The Code book

- The *a priori* codebook contains predefined codes and themes grounded in extant knowledge and hypotheses
  - For purely inductive studies, this will be empty
- The *a posteriori codebook* will contain the evolution of the *a priori* codes based on the analysed data

Photo by Sinziana Susa | https://unsplash.com/photos/opened-book-on-white-textile-SNHsMunOPME


What experience do you have from qualitative data analysis? Was it De/In/Abductive?

Various forms of data

- Scientific, grey and white literature
- Internal or public documents and communication
- Interview transcripts and open-ended survey responses
- Field notes and summaries, e.g., from observations, meetings or field visits
- Participant-generated data from workshops or diaries
- Visuals such as drawings, images, videos
- AI-generated data, e.g., through defined prompts
- …

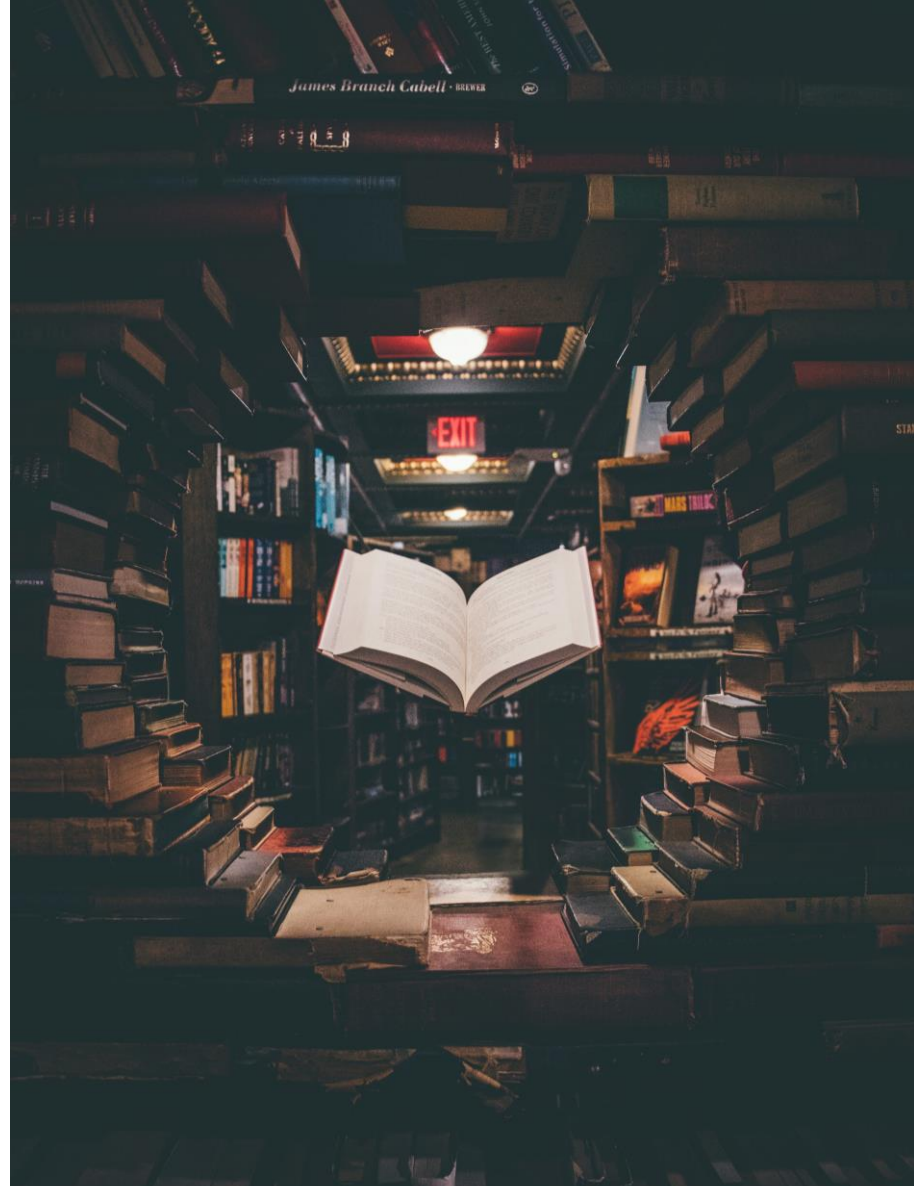


Photo by Jaredd Craig | https://unsplash.com/photos/view-of-floating-open-book-from-stacked-books-in-library-HH4WBGNyltc

What types of qualitative data sources have you analysed?

Structuring the data

- Need to enable
  - Systematic approach for analysis
  - Cross analysis and comparison between data sources
  - Audit trail from raw data across various levels of analysis and interpretation
  - Collaborative analysis and validation of the analysis
  - Intermediate interpretations of the data

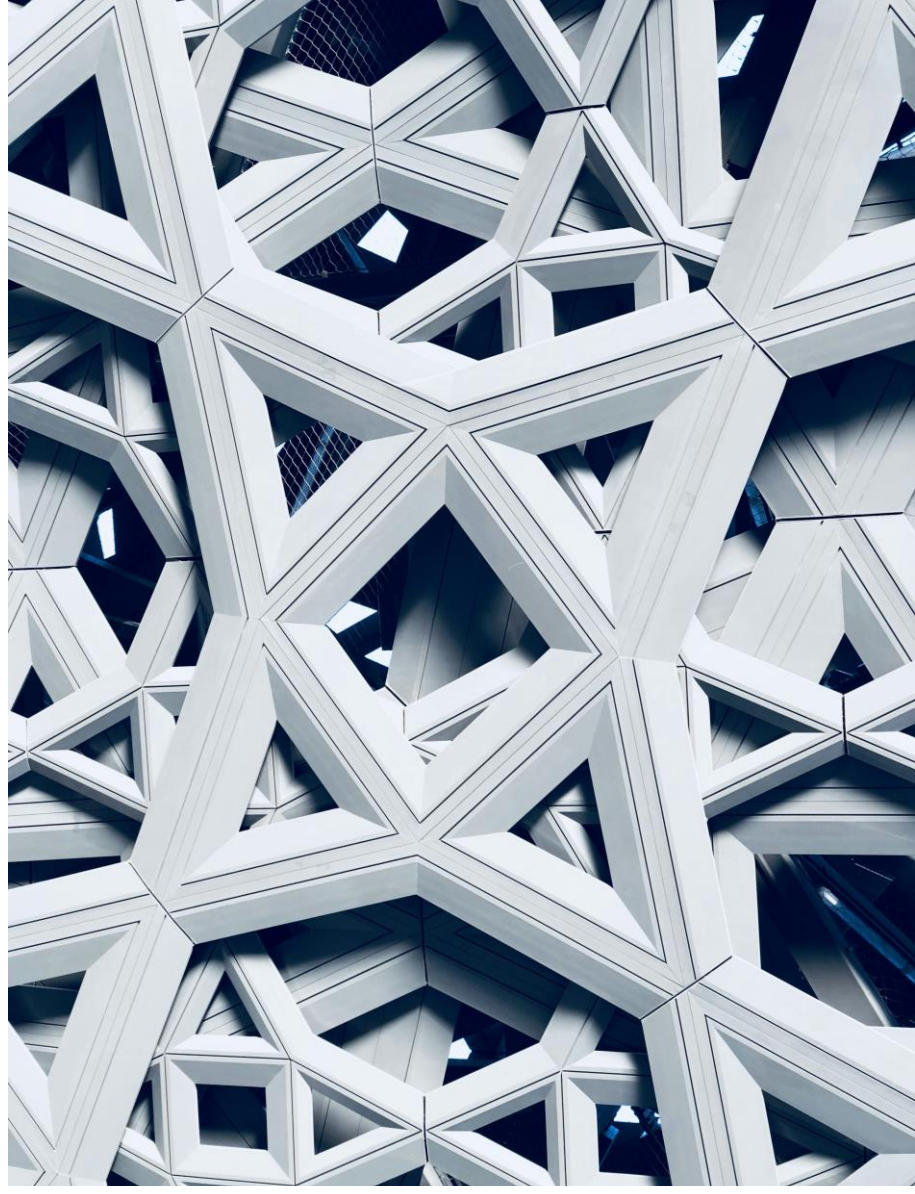


Photo by Alvaro Pinot | https://unsplash.com/photos/closeup-photo-of-white-decor-czDvRp5V2b0

How did you structure your data?


Example

Iterative process

- Analysis process starts synchronously with the data collection
- Reflections and thoughts when taking in the data provide input to the analysis
  - Should be recorded in what ever way possible and incorporated in continued analysis.
- Each data point should be analysed and incorporated in an intermediate analysis
- Will help to steer and inform future data collection

Photo by Abhipsa Pal | https://unsplash.com/photos/a-group-of-gears-sitting-on-top-of-a-metal-surface-8id1zGnmJ08

First cycle coding

- Codes are tags or labels capturing the essence of data excerpts through words or shorter sentences
- Essence comes by summarizing condensing, distilling, reducing, and enriching the key information of the excerpt
- Excerpts can range between sentences, paragraphs, and full pages
- Codes can emerge from data or from literature based on approach
- Code book continuously revised and evolved through analysis process per data point.
- Important to iterate to ensure new meaning of code, or occurrence of newly derived code

Photo by Patrick Perkins | https://unsplash.com/photos/assorted-notepads-ETRPjvb0KM0


Example

Coding for patterns

- Patterns links data through common characteristics, including
  - Similarities
  - Differences
  - Frequencies
  - Sequences
  - Causation
- Important to not only look for meaning through patterns, also need to consider the anomalies and incoherences

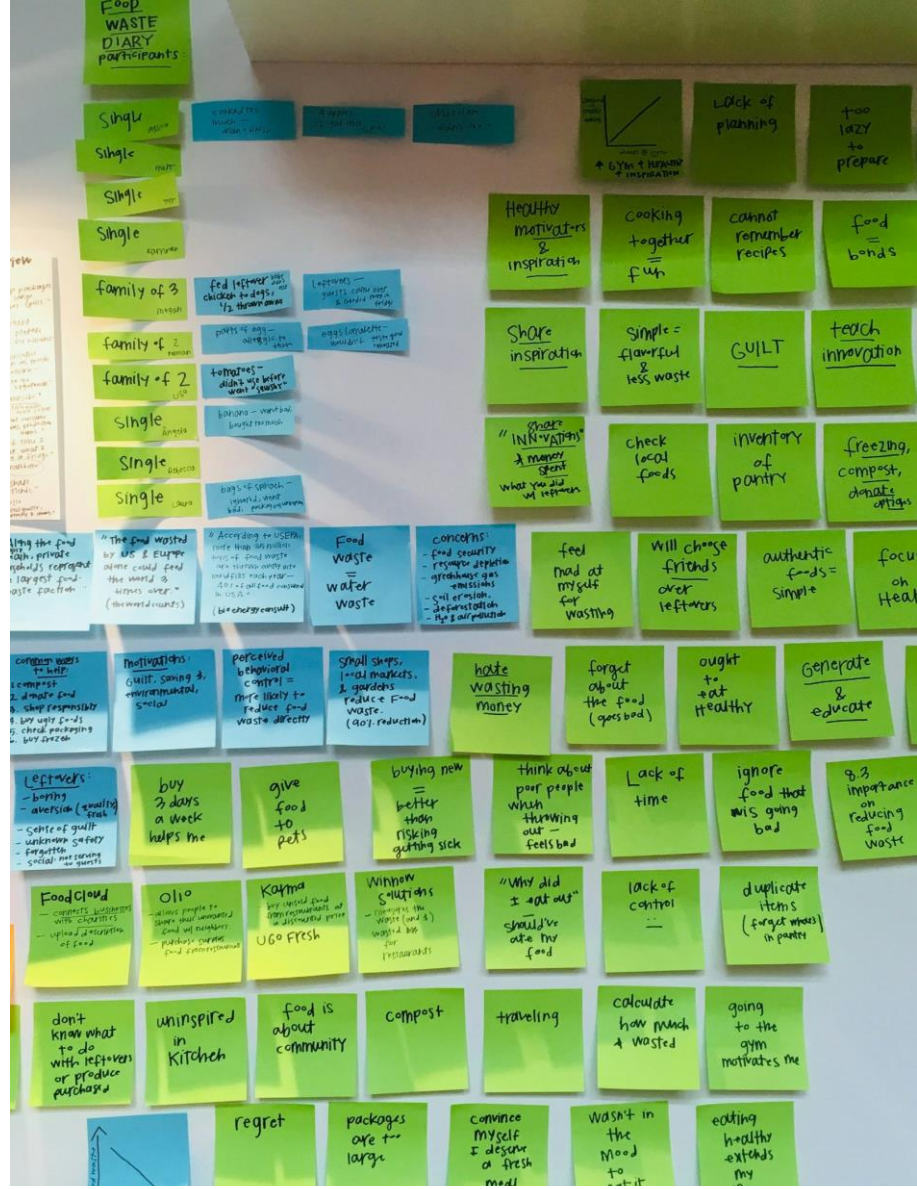


Photo by Hugo Rocha | https://unsplash.com/photos/assorted-color-sticky-notes-qFpnvZ_j9HU

Coding comes in different glasses

- Pending on the coding technique, you will get different patters and understanding of the data
- Coding technique, or lens, should align with your goals and research questions
- Can also provide ways of exploring and understanding the data from various perspectives

Photo by Bud Helisson | https://unsplash.com/photos/person-holding-brown-eyeglasses-with-green-trees-background-kqguzgvYrtM

Example (first cycle) coding methods

- Structural coding
  - Structures excerpts per topics or questions (e.g., RQs or from Questionnaire, enabling further detailed analysis (e.g., requirements engineering, testing, community management)
- Descriptive coding
  - Highlights topics and scope talked about in the data excerpts (e.g., toxicity, turnover, contribution process)
- In vivo coding
  - Adopts the terms used in the data excerpts, e.g., by the interviewee, capturing their personal reflection, not your own (e.g., stressed, burned-out)
- Process coding
  - Gerunds (“-ing” words) describing actions performed or described in the data excerpts (e.g., fuzz-testing, reviewing, debugging)

RI.
SE

Example (first cycle) coding methods

- Values coding
  - Reflects the values, attitudes or beliefs expressed in the data excerpt (e.g., opinion on whether a certain practice is good, bad, efficient or not)
- Versus coding
  - Identify dichotomous or binary terms that stand out in relation to each other (e.g., open source vs proprietary software)
- Initial coding (open coding)
  - An inductive approach to coding – adopting, e.g., one of the previous methods, and iteratively developing codes from the data
- Hypothesis coding
  - A deductive approach to coding – adopting, e.g., one of the previous methods, and iteratively applying codes generated by theory, conceptual frameworks, or hypotheses
- + many more… See **Saldana (2016)** for more details.

RI.
SE

What kind of coding method(s) have you used?

Choosing the right method(s)

- See what methods fits the type of data, type of study and your research questions
- Start of with a generic approach, but be open to trial different options
- Starting out from a clean sheet with different methods may provide new insights
  - Pilot tests can help determine suitability
- Combining methods can add value and strengthen synthesis, but don't over complicate it

Photo by Victoriano Izquierdo | https://unsplash.com/photos/man-on-front-of-vending-machines-at-nighttime-JG35CpZLfVs

Continuous reflection and analytic memos

- Thoughts and reflections throughout the analysis process
  - Notes and reflections during and after interviews
  - Summaries and reflections on data excerpts during coding process
  - Working case synthesis and state of understanding
- Provides a core or input to general synthesis, conceptualization, theorization and paper writing
- Critical to capture your running thought process. Risk of loosing valuable insights otherwise.
- Helps to condense, distil and summarize data, and provides a as-critical tool as the codes
- Provides the necessary context needed, while avoiding getting drowned in the data

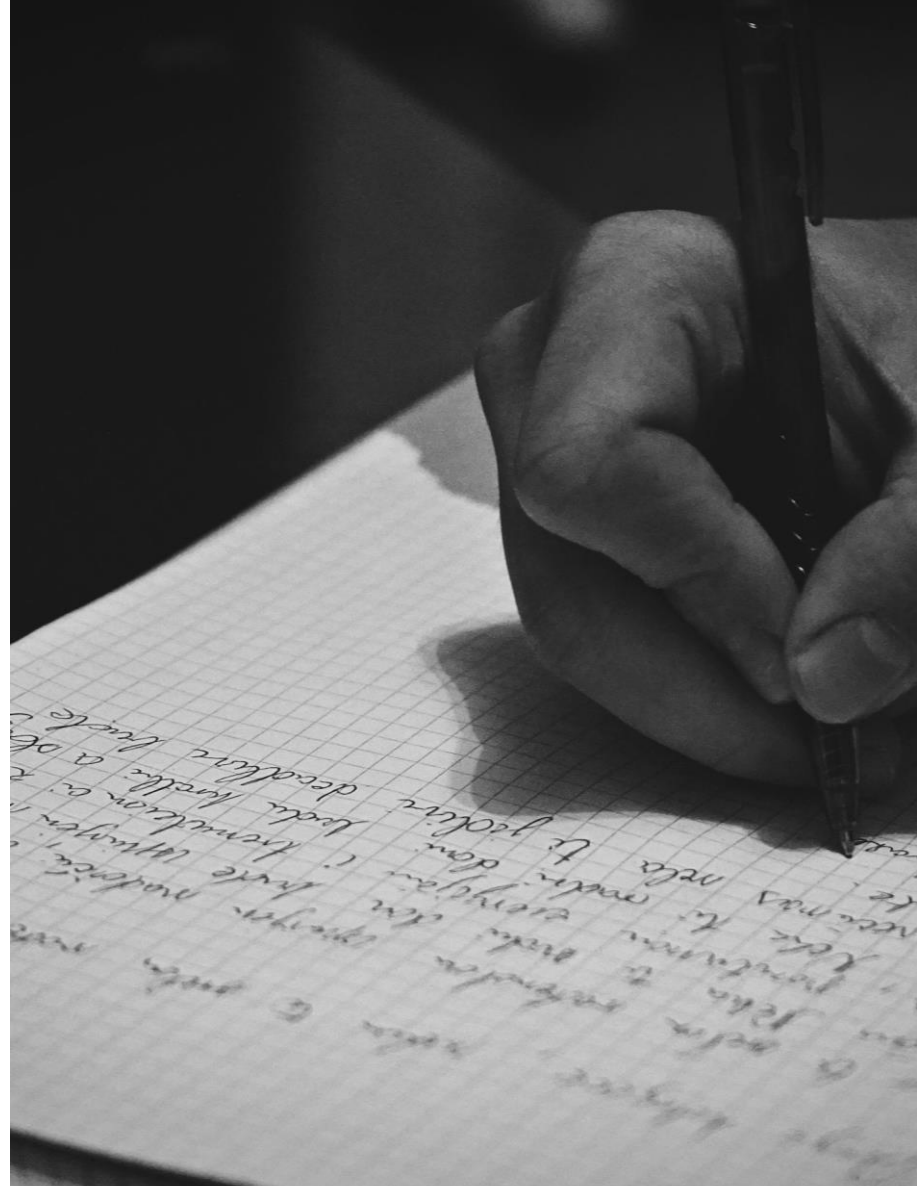


Photo by Nedim | https://unsplash.com/photos/a-person-writing-on-a-notebook-with-a-pen-6CQy1sklbRs

Coding exercise: How can toxicity in open source development be characterized?

Coding exercise

- RQ: How can toxicity in open source development be characterized?
- You have chosen to study the Linux kernel project and its mailing list, which has a reputation for having a harsh tone.
- You are starting with an inductive approach, iteratively analysing mail threads from a larger sample.
- Study the mail thread and apply a suitable coding method

Photo by Patrick Perkins | https://unsplash.com/photos/assorted-notepads-ETRPjvb0KM0

What kind codes and insight did you elicit?

Coding exercise

- Now continue with a deductive approach
- You leverage the Framework of characteristics of open source toxicity developed by Miller et al., as an a-priori code book.

"Did You Miss My Comment or What?" Understanding Toxicity in Open Source Discussions

Courtney Miller, Sophie Cohen, Daniel Klug, Bogdan Vasilescu, Christian Kästner
Carnegie Mellon University Wesleyan University
{courtneymiller, dklug, vasilescu}@cmu.edu

ABSTRACT

Online toxicity is ubiquitous across the internet and its negative impact on the people and that online communities that it effects has been well documented. However, toxicity manifests differently on various platforms and toxicity in open source communities, while frequently discussed, is not well understood. We take a first stride at understanding the characteristics of open source toxicity to better inform future work on designing effective intervention and detection methods. To this end, we curate a sample of 100 toxic GitHub issue discussions combining multiple search and sampling strategies. We then qualitatively analyze the sample to gain an understanding of the characteristics of open-source toxicity. We find that the pervasive forms of toxicity in open source differ from those observed on other platforms like Reddit or Wikipedia. In our sample, some of the most prevalent forms of toxicity are entitled, demanding, and arrogant comments from project users as well as insults arising from technical disagreements. In addition, not all toxicity was written by people external to the projects; project members were also common authors of toxicity. We also discuss the implications of our findings. Among others we hope that our findings will be useful for future detection work.

ACM Reference Format:
Courtney Miller, Sophie Cohen, Bogdan Vasilescu, Christian Kästner. 2022. *"Did You Miss My Comment or What?"* Understanding Toxicity in Open Source Discussions. In *44th International Conference on Software Engineering (ICSE '22), May 21–29, 2022, Pittsburgh, PA, USA.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3510003.3510111

volunteering their time, talk openly about how sometimes interactions with others in open source can be toxic, rude, mean, or unkind [e.g., 3, 24, 44, 54, 108]. Toxicity, defined here as "rude, disrespectful, or unreasonable language that is likely to make someone leave a discussion"[1] is a huge problem online [26]. Virtually all online platforms recognize the threat that toxicity, or the various types of behavior under its umbrella, poses on the health and safety of online communities. As a result, a number of prevention and mitigation policies and interventions have been proposed, including codes of conduct, moderation, counterspeech, shadow banning, or just-in-time guidance to authors.

Expectedly, open source communities are not immune to toxicity. While the term "toxicity" as defined above has only recently started being used in the open-source literature [17, 79, 86], the presence of behaviors "likely to make someone leave" have long been documented by researchers and practitioners in this space. For example, the Linux Kernel Mailing List is notorious for having discussions with a tone that "tends to discourage people from joining the community" [24].[2] Generally, in open source the tone of project discussions is something newcomers pay attention to when deciding to join projects [76] and it can also act as a barrier to onboarding [93]. In an attempt to discourage abuse and harassment and to set acceptable behavior norms, many open source projects have adopted codes of conduct [55, 98].

Contributor disengagement, especially when precipitated, is of major concern in open source [46]. With many important open source projects being maintained by one or two volunteers [4],

How were the a-priori codes applied? How did they match/differ with your inductive codes?

Second cycle coding

- Continued condensation, distillation and summative compilation of extant codes
- Goal is to create (re)organize first level abstractions into higher level categories, themes and conceptual structures with relations between
- The process requires equal parts logic and creativity

Photo by FORTYTWO | https://unsplash.com/photos/person-holding-yellow-sticky-notes-MDu-53qRVr4

Example (second cycle) coding methods

- Pattern coding
  - Summarizes and clusters first cycle codes and summaries to higher categories, looking to explain or show emergent themes from across the data
- Focused coding
  - Looks specifically to develop the most saliant themes, typically from a “grounded theory” and inductive PoV.
- Axial coding
  - House cleaning of the first cycle codes, removing redundant codes, merging similar, and keeping the codes best representing the data
- Longitudinal coding
  - Captures change over time as expressed in the data, e.g., increase/emergence, surge/turning point, decrease/cease, constancy/consistency.

RI.
SE

Coding exercise

- Now continue with the second-cycle, considering both you inductive and deductive codes

Photo by Patrick Perkins | https://unsplash.com/photos/assorted-notepads-ETRPjvb0KM0

How can toxicity in open source development be characterized?

Working synthesis

- Captures ongoing analysis and provides a “state-of-understanding”
- Beneficial when multiple data sources and researchers involved
- Provides trace-data for how understanding evolves
- Especially critical for longitudinal studies, e.g., with recurrent interviews of same case org

Photo by Hannah Olinger | https://unsplash.com/photos/a-person-writing-on-a-piece-of-paper-with-a-pen-8eSrC43qdro

Conceptualization and theorization

- Theoretical coding (or selective coding)
  - Connects and explains all higher-level codes and categories around a central or core category, with the goal of creating a theory (per grounded theory methodology)
- Categorizing, condensing and distilling to the highest yet practical level possible, Answering the research questions defined
- Explaining the relationships between categories, e.g., causes, depends, increases…
- Arriving at a technological rule, capturing general knowledge mapping between a problem-solution pair. Can be phrased as: “to achieve <Effect> in <Situation> apply <Intervention>”

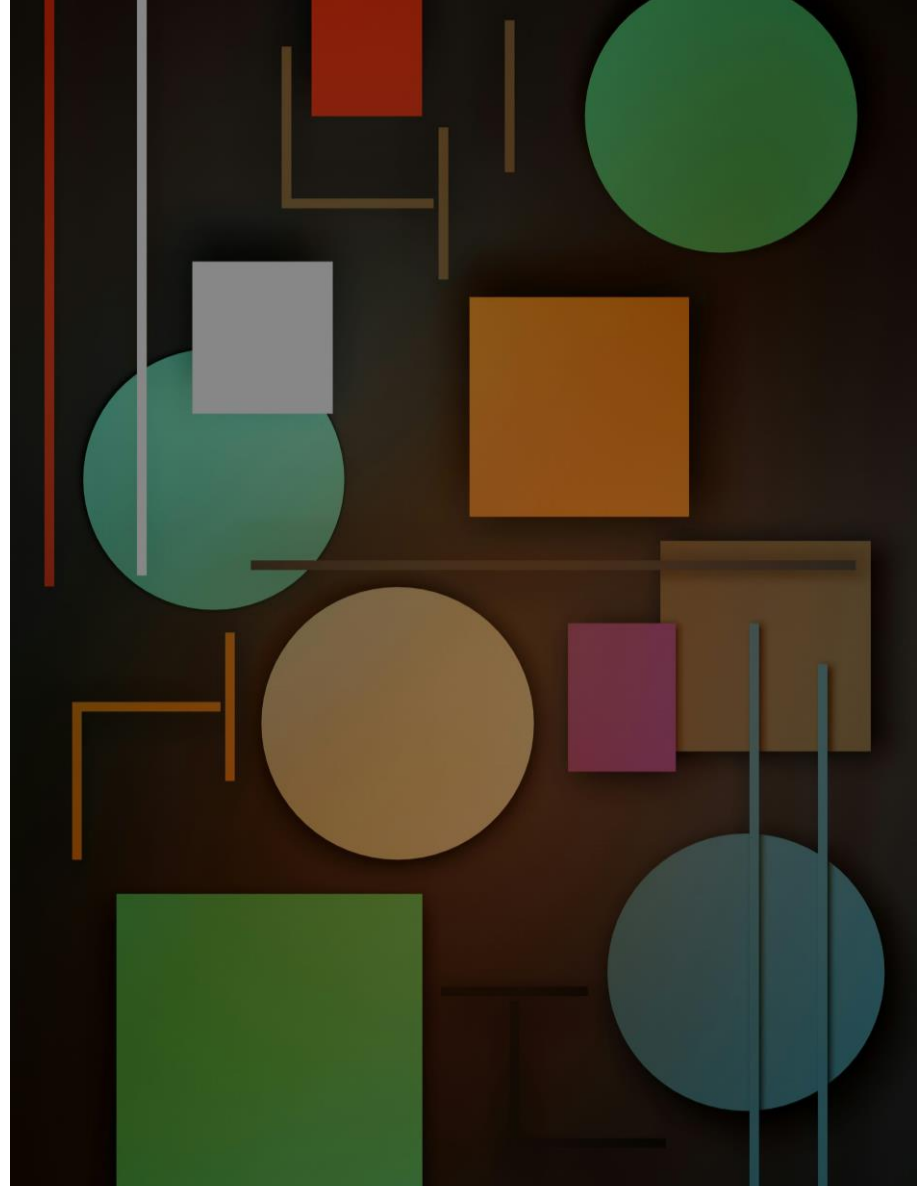


Photo by Sufyan | https://unsplash.com/photos/a-group-of-different-colored-shapes-on-a-black-background-_Mpa9FoBAKU

Further ways of making sense

- Top 10 most relevant codes
- Top 3 – Venn-diagram
- Code weaving
- Network graphs and other data visualization techniques
- See Miles, Huberman and Saldana (2020)

Photo by Deng Xiang | https://unsplash.com/photos/graphical-user-interface--WXQm_NTK0U

Saturation

- No new information, explanations, codes or categories emerge from the data
- Consider mainly from the higher-level codes and themes,
  - On first code-level, new ones can be added constantly
- Not always a black or white decision
- Consider similarity or differences in data sources, and how they overlap. Collect more data until satisfactory
- Ties back to scope of study, e.g., exploratory or confirmatory?

Photo by Jana Knorr | https://unsplash.com/photos/red-and-white-no-smoking-sign-d-JNCzsRDFE

How have you determined saturation?

Strengthening Validity

- Peer-coding/researcher triangulation
  - Full or partial
  - Confirmatory review
  - Settling and reporting disputes
- Member-checking
  - Transcript and/or interview summary
  - First/mid-way synthesis and/or coding
  - Final reporting
- Data-triangulation
- Increasing and/or diversifying sampling
- Contextual descriptions and quotes

Photo by VARUN RADHAKRISHNAN | https://unsplash.com/photos/a-green-dumbbell-sitting-on-top-of-a-metal-bench-irRQoTtwGDQ

Further reading

- Milea, M., Huberman, M., & Saldaña, J.(2020). Qualitative data analysis: A Method sourcebook, 4th edition. Sage.
- Saldaña, J. (2021). The coding manual for qualitative researchers. Sage.
- Melegati, J., Conboy, K., & Graziotin, D. (2024). Qualitative Surveys in Software Engineering Research: Definition, Critical Review, and Guidelines. *IEEE Transactions on Software Engineering*.

Photo by Christin Hume | https://unsplash.com/photos/person-picking-white-and-red-book-on-bookshelf-k2KcwkandwgI