



Supporting Qualitative Analysis with Large Language Models: Combining Codebook with GPT-3 for Deductive Coding

Ziang Xiao
ziangxiao@microsoft.com
Microsoft Research
Montreal, Quebec, Canada

Xingdi Yuan
eric.yuan@microsoft.com
Microsoft Research
Montreal, Quebec, Canada

Q. Vera Liao
veraliao@microsoft.com
Microsoft Research
Montreal, Quebec, Canada

Rania Abdelghani
rania.abdelghani@inria.fr
Inria
France

Pierre-Yves Oudeyer
pierre-yves.oudeyer@inria.fr
Inria
France

ABSTRACT

Qualitative analysis of textual contents unpacks rich and valuable information by assigning labels to the data. However, this process is often labor-intensive, particularly when working with large datasets. While recent AI-based tools demonstrate utility, researchers may not have readily available AI resources and expertise, let alone be challenged by the limited generalizability of those task-specific models. In this study, we explored the use of large language models (LLMs) in supporting deductive coding, a major category of qualitative analysis where researchers use pre-determined codebooks to label the data into a fixed set of codes. Instead of training task-specific models, a pre-trained LLM could be used directly for various tasks without fine-tuning through prompt learning. Using a curiosity-driven questions coding task as a case study, we found, by combining GPT-3 with expert-drafted codebooks, our proposed approach achieved fair to substantial agreements with expert-coded results. We lay out challenges and opportunities in using LLMs to support qualitative coding and beyond.

CCS CONCEPTS

• **Human-centered computing** → **HCI design and evaluation methods**; • **Computing methodologies** → **Natural language processing**.

KEYWORDS

Qualitative Analysis, Deductive Coding, Large Language Model, GPT-3

ACM Reference Format:

Ziang Xiao, Xingdi Yuan, Q. Vera Liao, Rania Abdelghani, and Pierre-Yves Oudeyer. 2023. Supporting Qualitative Analysis with Large Language Models: Combining Codebook with GPT-3 for Deductive Coding. In *28th International Conference on Intelligent User Interfaces (IUI '23 Companion)*, March 27–31, 2023, Sydney, NSW, Australia. ACM, New York, NY, USA, 4 pages. <https://doi.org/10.1145/3581754.3584136>

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

IUI '23 Companion, March 27–31, 2023, Sydney, NSW, Australia

© 2023 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-0107-8/23/03.

<https://doi.org/10.1145/3581754.3584136>

1 INTRODUCTION

Qualitative coding is a method of qualitative analysis that is used to identify patterns and categories in qualitative data, such as social media posts, open-ended survey responses, and field notes. Often, scientists will use coded data to derive theory or build models to understand the observed phenomenon further [5]. Qualitative coding can be a challenging process because it requires the researcher to understand and analyze data that is often complex, nuanced, and open to multiple interpretations. Multiple researchers need to spend a significant amount of time and effort to review the data, develop a codebook that can accurately describe each label, and iteratively code the data until reaching a reasonable inter-rater agreement. It is particularly hard when working with large datasets. In this study, we explore a novel approach that leverages large language models (LLMs) to facilitate qualitative coding.

Researchers have built AI-based tools to assist qualitative analysis [7, 10–12]. These tools use natural language processing (NLP) and machine learning (ML) algorithms to help researchers identify patterns and themes in qualitative data. Two categories of models are often used, 1) unsupervised models, e.g., topic models, to help researchers discover themes, or 2) supervised models, e.g., logistic regression, to classify data into labels. However, both methods face their own challenges. Unsupervised models are difficult to steer. It is difficult for researchers to use those models for their complex or nuanced research questions. And supervised models often require high-quality large datasets or computing resources to achieve reasonable performance. Therefore, most of today's qualitative coding still relies on manual effort.

The recent advent of LLMs (e.g., GPT-3 [2], PaLM [3], OPT [16]) offers new capabilities, including creative writing [14], programming¹, etc. Unlike traditional task-specific models, LLMs accept natural language prompts as input to perform various tasks [8]. For example, LLMs could perform classification tasks if the prompt specifies a set of labels as the output space [4]. Compared to unsupervised models, the prompt could include specific instructions and examples (e.g., a codebook) to increase LLMs' performance in a new task with unseen data in a zero-shot/few-shot fashion. Prior work shows that LLMs can be prompted to boost an NLG system's performance by selecting better outputs from sampled candidates, according to some pre-defined metrics [15]. Since LLMs operate

¹<https://github.com/features/copilot>

effectively with natural language input and do not require training datasets, they lower the barriers for researchers without extensive AI expertise or resources to leverage AI capabilities in data analysis (although many of today’s LLMs are proprietary).

Although recent studies have demonstrated LLMs utility in many domains, studies have shown LLMs are error-prone and have limited capability in capturing language structure and nuanced meanings [6], which are crucial in qualitative analysis. Therefore, in this preliminary study, we asked two research questions,

- **RQ1:** To what extent does our LLM-based approach agree with experts in a deductive coding task?
- **RQ2:** How do different prompt designs affect the coding results?

We examined LLMs’ capability in facilitating a deductive coding task where we combined GPT-3 with expert-developed codebooks to analyze children’s curiosity-driven questions in terms of question complexity and syntactic structure. We found our proposed approach achieved fair to substantial agreements with experts (Question complexity: Cohen’s $\kappa = 0.61$; Syntactic Structure: Cohen’s $\kappa = 0.38$). Based on our preliminary findings, we present challenges and opportunities for utilizing LLMs in qualitative analysis.

2 DEDUCTIVE CODING TASK

We selected a deductive coding task as our starting point. The goal of deductive coding is to label the data based on a codebook. It is a top-down process where the researchers start by developing a codebook with an initial set of codes along with descriptions and examples based on the research focus or theory [5]. Note that a coding process often involves both inductive coding (e.g. developing the code book) and deductive coding (coding all the data according to a code book). Our approach can be used to complete the second part or combined with the first part to help coders rapidly iterate on their codebook.

2.1 Case Study: Curiosity-driven questions Analysis

We examined our approach in analyzing children’s curiosity-driven questions. Understanding children’s ability to ask curiosity-driven questions informs psychologists of a child’s learning stage. We looked at two dimensions of a curiosity-driven question, question complexity and syntactic structure. The question complexity looks at if the answer to a question is a simple fact (e.g., “How big is a dinosaur?”) or requires explaining a mechanism, a relationship, etc.(e.g., “Why were dinosaurs so big?”) [13]. The syntactic structure has four categories, 1) ‘closed’ or declarative questions (e.g., “Dinosaurs were big?”), 2) questions with questioning words in the middle of the sentence (e.g., “The dinosaurs were how big?”), 3) questions without an interrogative formulation (e.g., “Why the dinosaurs are big?”), and 4) questions with a questioning word at the beginning of the sentence that has interrogative syntax (e.g., “Why are dinosaurs big?”).

We collected a dataset with 668 children’s questions in French [1]

² A team of psychologists has developed a codebook and coded each

²For question complexity, we first used GPT-3 to translate questions into English. For the syntactic structure, we kept questions in French to preserve its syntactic structure.

question on the dimension of question complexity and syntactic structure. Our chosen dimensions cover two main categories of deductive coding, binary coding, and multi-level coding, with one focusing on the semantic meaning and the other looking at the syntactic structure. Additionally, the dataset and codebook have never been published online which is unseen by LLMs.

3 GPT-3 SETUP AND PROMPT DESIGN

We chose GPT-3 (davinci-text-002) with a temperature of 0.0 during the prompting process, because it was the most advanced version of GPT-3 that was publicly available when we conducted the experiments and 0.0 temperature (greedy decoding) guarantees the reproducibility of this study.

We explored two design dimensions of the prompt,

- **Codebook-centered vs. Example-centered:** This dimension regards the structure of a prompt. In the codebook-centered prompt, we designed the prompt similar to how researchers read a codebook. The prompt follows the structure of [Code/ Description/ Examples]. For example, **Code: HIGH; Description: the answer to this question is not a simple fact but requires explaining a mechanism, a relationship, etc.; Examples: Why were dinosaurs so big?** The example-centered approach is inspired by the in-context learning in recent LLM works where the prompt explains the rationale behind each example [8]. For example, **“Why were dinosaurs so big?” is an example of “HIGH” because the answer to this question is not a simple fact but requires explaining a mechanism, a relationship, etc.** The code, examples, and descriptions are the same for both designs.
- **Zero-shot vs. One-shot vs. Few-shots:** Since recent work showed conflicting results on the number of examples in a prompt [8], we explored different prompt settings. In the Zero-shot setting, we give only Code and Description in Codebook-centered prompts ³. For the One-shot setting, we provided only one example for each code. And for the few-shot setting, we provided five examples for each code.

For all prompt variants, we included an identity modifier, “I am a developmental psychologist who has expertise in linguistics.” and an instruction to constrain output space, “Choose from the following candidates: [Code Set]”.

4 RESULTS

We measured the performance of our GPT-3 based approach with Cohen’s Kappa [9]. Cohen’s Kappa measures inter-rater reliability, which indicates how two coders agree with each other. We computed two sets of Cohen’s Kappa, between GPT-3 with the expert’s final coding results and between two experts who originally coded the dataset using the *same* codebook.

For RQ1, our results suggest that it is feasible to use GPT-3 with an expert-developed codebook for deductive coding. When analyzing curiosity-driven questions, our GPT-3-based approach achieved fair (Syntactic Structure: Cohen’s $\kappa = 0.38$) to substantial (Question complexity: Cohen’s $\kappa = 0.61$) agreement with expert rating (Cohen’s κ ’s interpretation is based on [9]), see Fig. 1. However, there

³Since the example-centered approach requires at least one example, we did not have the zero-shot setting for the example-centered approach

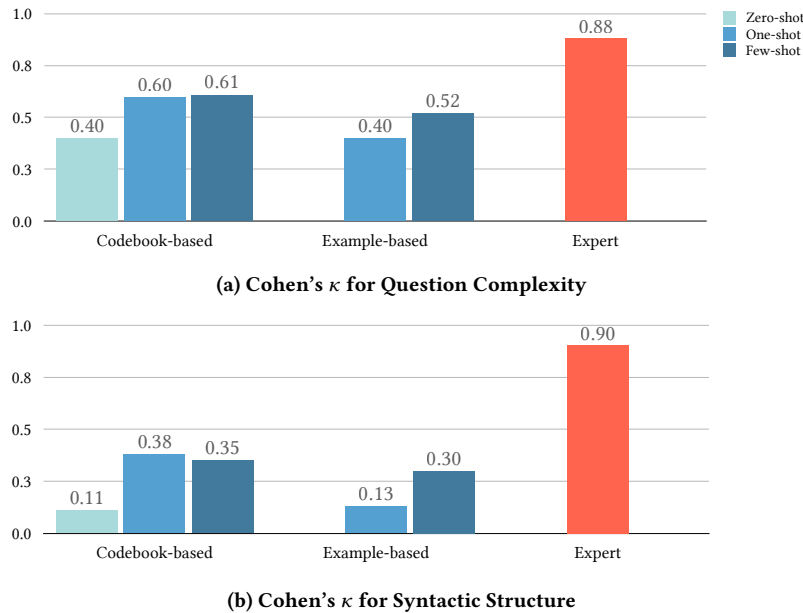


Figure 1: The Cohen’s κ between GPT-3 and experts shows substantial agreement in Question Complexity coding and fair agreement in Syntactic Structure coding. In general, Codebook-centered prompts with examples achieves the highest agreement.

is a gap between experts’ agreement with our approach and the agreement among experts (Question complexity: Cohen’s $\kappa = 0.88$; Syntactic Structure: Cohen’s $\kappa = 0.90$).

For RQ2, we examined the different prompt designs. We found the codebook-centered design performs better than the example-centered designs, see Fig. 1. And examples play an important role. We observed the largest performance gain when shifting from a zero-shot to a one-shot setting. However, the performance between one-shot and few-shot settings did not differ much.

5 OPPORTUNITIES AND CHALLENGES

Our preliminary findings indicate the feasibility of using LLMs for qualitative analysis. In a deductive coding task, by combining GPT-3 and a codebook, our LLM-based approach achieved fair to substantial agreement with experts. Considering the accessibility and flexibility of LLMs, we believe this approach has the potential to effectively help researchers to analyze qualitative data, especially for increasingly large datasets. We lay out a few challenges and opportunities for our future studies,

- **Interrogate Model Capability:** In this preliminary exploration, we only measured the level of agreement. To better understand model capability, we ought to conduct more detailed error analyses on disagreed items. Also, it is unclear if the LLM-based method could be extended to different contexts and different coding schemes where the coder needs to capture more nuanced signals.
- **Design for Appropriate Reliance:** Although our preliminary results show fair to substantial agreement with expert rating, the model produces incorrect labels. When deploying such an imperfect AI system, we must design for appropriate

reliance to prevent over-trusting and misuse. For example, when designing the interface, we could consider explainable AI methods to calibrate user trust over time.

- **Design Codebook for LLMs:** We constructed our prompts using the codebook for experts. Although it provides transparency and explicit control, it may limit the model's performance. Future study is required to understand how to design a more effective codebook for task performance and model understanding.
- **Support Inductive Coding:** We demonstrated the feasibility of using LLMs in deductive coding. However, for inductive coding, where the process is more bottom-up and exploratory, novel interaction and Human-AI collaboration diagrams are required. We should study interaction techniques and controls to let researchers use LLMs more effectively in the qualitative analysis.

REFERENCES

- [1] Rania Abdelghani, Pierre-Yves Oudeyer, Edith Law, Catherine de Vulpillieres, and H  lene Sauz  on. 2022. Conversational agents for fostering curiosity-driven learning in children. *arXiv preprint arXiv:2204.03546* (2022).
- [2] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Gishri Sarsh, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (Eds.), Vol. 33. Curran Associates, Inc., 1877–1901. <https://proceedings.neurips.cc/paper/2020/file/1457cd6bfcb4967418bf8ba412f6da-Paper.pdf>.
- [3] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2022. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311* (2022).

- [4] Tianyu Gao, Adam Fisch, and Danqi Chen. 2021. Making Pre-trained Language Models Better Few-shot Learners. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Association for Computational Linguistics, Online, 3816–3830. <https://doi.org/10.18653/v1/2021.acl-long.295>
- [5] Hsiu-Fang Hsieh and Sarah E Shannon. 2005. Three approaches to qualitative content analysis. *Qualitative health research* 15, 9 (2005), 1277–1288.
- [6] Diane M Korngiebel and Sean D Mooney. 2021. Considering the possibilities and pitfalls of Generative Pre-trained Transformer 3 (GPT-3) in healthcare delivery. *NPJ Digital Medicine* 4, 1 (2021), 1–3.
- [7] Jasy Suet Yan Liew, Nancy McCracken, Shichun Zhou, and Kevin Crowston. 2014. Optimizing features in active machine learning for complex qualitative content analysis. In *Proceedings of the ACL 2014 Workshop on Language Technologies and Computational Social Science*. 44–48.
- [8] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *arXiv preprint arXiv:2107.13586* (2021).
- [9] Mary L McHugh. 2012. Interrater reliability: the kappa statistic. *Biochemia medica* 22, 3 (2012), 276–282.
- [10] Michael Muller, Shion Guha, Eric PS Baumer, David Mimno, and N Sadat Shami. 2016. Machine learning and grounded theory method: convergence, divergence, and combination. In *Proceedings of the 19th international conference on supporting group work*. 3–8.
- [11] Pablo Paredes, Ana Rufino Ferreira, Cory Schillaci, Gene Yoo, Pierre Karashchuk, Dennis Xing, Coye Cheshire, and John Canny. 2017. Inquire: Large-scale early insight discovery for qualitative research. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*. 1562–1575.
- [12] Tim Rietz and Alexander Maedche. 2021. Cody: An AI-Based System to Semi-Automate Coding for Qualitative Research. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (Yokohama, Japan) (CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 394, 14 pages. <https://doi.org/10.1145/3411764.3445591>
- [13] William W Wilen. 1991. Questioning skills, for teachers. What research says to the teacher. (1991).
- [14] Ann Yuan, Andy Coenen, Emily Reif, and Daphne Ippolito. 2022. Wordcraft: Story Writing With Large Language Models. In *27th International Conference on Intelligent User Interfaces*. 841–852.
- [15] Xingdi Yuan, Tong Wang, Yen-Hsiang Wang, Emery Fine, Rania Abdelghani, Pauline Lucas, Hélène Sauzéon, and Pierre-Yves Oudeyer. 2022. Selecting Better Samples from Pre-trained LLMs: A Case Study on Question Generation. *arXiv preprint arXiv:2209.11000* (2022).
- [16] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. 2022. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068* (2022).