# Automating Protocol Analysis with Generative AI: Classifying Questions in Design Review Meetings Using GPT-4

by

Ahmed Shahriar Sakib

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Applied Science
in
Management Science and Engineering

Waterloo, Ontario, Canada, 2024

© Ahmed Shahriar Sakib 2024

## Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

## Statement of Contributions

Ahmed Shahriar Sakib was the sole author of this thesis which was written under the supervision of Dr. Ada Hurst and Dr. Frank Safayeni. Preliminary results from this research were previously published in the Design Computing and Cognition 2024 Conference proceedings.

Citation:

Sakib, A. S., Hurst, A., & Safayeni, F. (2024, July). Towards AI-Assisted Protocol Analysis in Design Research: Automating Question Labelling with GPT-4 According to Eris'(2004) Taxonomy. In *International Conference on-Design Computing and Cognition* (pp. 38-55). Cham: Springer Nature Switzerland. DOI: 10.1007/978-3-031-71918-9_3

## Abstract

This study investigates the extent to which generative AI can classify question utterances in verbal protocols of design according to Eris' (2004) taxonomy utilizing in-context learning (ICL). Specifically, it examines the impact of various factors on classification, including the size of the prompt data, contextual information inclusion, prompt engineering strategies, cross-dataset applicability, and cross-model evaluation. The findings of this research could pave the way for more widespread adoption of AI in design research, transforming how protocols are analyzed and interpreted, ultimately leading to more efficient and accurate insights into cognitive processes during design activities.

A series of experiments was conducted to evaluate the performance of GPT-4, a state-of-the-art generative AI model, in this context. The experiments involved utilizing ICL by providing the AI model on a dataset of human-labeled questions and testing its ability to classify new questions according to predefined categories. The findings indicate that GPT-4 performs reasonably well in categorizing stand-alone question utterances, achieving alignment with human-sourced labels in many cases. Moreover, GPT-4 also shows promising generalization capability across the datasets used in this study. The study also highlights the potential of using the less expensive proprietary LLM, Claude 3.5, for similar tasks without a significant drop in performance, making it a more accessible option. The results also imply that classification accuracy depends on the dataset quality, indicating that performance may improve with a higher-quality dataset. However, the results also reveal that providing additional context does not always enhance, and in some cases even diminishes, the model's performance, highlighting the challenges of context-dependent classification tasks.

The implications of these findings suggest that while generative AI shows promise as a tool for automating protocol analysis, there are significant limitations that must be addressed to fully leverage its capabilities in design research. In conclusion, this research contributes to the growing body of knowledge on the application of AI in design research, proposing several directions for future research aimed at refining the use of artificial intelligence in qualitative analysis.

**Acknowledgements**

First and foremost, I would like to thank my supervisors, Frank Safayeni and Ada Hurst, for their limitless patience and priceless effort in shaping my understanding and my study. I truly appreciate their time for all the lengthy discussions that brought me a more intellectual point of view, though this thesis presents only a humble proportion of it.

I would like to thank P. Robert Duimering and Olga Vechtomova for their valuable feedback. I appreciate their efforts to read my lengthy study carefully.

I would like to express my heartfelt gratitude to my beloved fiance for her unwavering support, encouragement, and steadfast belief in me throughout this journey.

I am also deeply thankful to my dear friends, Fuat and Aniket, for their companionship, kindness, and encouragement, which have made a significant impact on my academic journey. I also wish to express my sincere appreciation to my friends, Sharleen, Nazmus Sakib, Shovon Das, Jami Ahmed, Ruyel Miah, Ahmed Borle, Jhuma, Khaled Shah, Shawon Arefin, Daniel Yu, Amina, Ramisha, Nayem, Akib, Ashik, Jim, Lima, Punno, Saad, and Musab. Your collective presence has been invaluable, and I am forever grateful.

I also want to thank my friends and their families living in Ontario who supported me when adapting to this new environment. I truly feel thankful to them for their succor.

Lastly, I want to acknowledge Eiichiro Oda for his manga Wan Pīsu, which has been both a source of inspiration and a much-needed escape throughout my thesis journey. I began this research alongside the first volume and am delighted to conclude it having finally caught up to the latest chapter. His work reminded me to stay persistent and find joy even in challenging moments. Just as the Straw Hat crew sails toward their dreams despite the odds, I too have navigated the waves of this journey with the belief that each step brings me closer to my own aspirations. And with this thesis completed, I can proudly say I have concluded my academic journey before the release of GTA 6—a milestone I'll always look back on with a smile.

## Dedication

This thesis is dedicated to my beloved parents, my dear sibling, and the courageous souls of the Bangladesh Quota Reform Movement, whose resilience and sacrifices I hold in utmost respect.

# Table of Contents

**Author's Declaration** ii

**Statement of Contributions** iii

**Abstract** iv

**Acknowledgements** v

**Dedication** vi

**List of Figures** x

**List of Tables** xi

**1 Introduction** 1

**2 Background & Related Work** 5

2.1 Overview of Generative AI . . . . . 5

2.2 Applications of Generative AI in Qualitative Research . . . . . 7

2.3 Protocol Analysis in Design Research . . . . . 9

2.4 Automation in Protocol Analysis . . . . . 10

2.5 Using GPT-4 for Protocol Analysis . . . 12
2.6 Summary of the Literature . . . 13

**3 Methodology** **15**
3.1 Scope of the Study . . . 15
3.2 Dataset . . . 19
3.3 Tool . . . 21
3.4 Tasks . . . 23
3.4.1 Preliminary Experiments . . . 23
3.4.2 Extended Experiments . . . 24

**4 Experiments & Results** **26**
4.1 Preliminary Experiments . . . 26
4.1.1 Experiment 1 . . . 27
4.1.2 Experiment 2 . . . 30
4.1.3 Experiment 3 . . . 31
4.1.4 Experiment 4 . . . 35
4.2 Extended Experiments . . . 46
4.2.1 Experiment 5 . . . 47
4.2.2 Experiment 6 . . . 53
4.2.3 Experiment 7 . . . 56
4.2.4 Experiment 8 . . . 60

**5 Discussion** **64**
5.1 Classification accuracy and sensitivity . . . 65

5.2 The role of context in the question classification task . . . . . 71
5.3 Broader considerations in using LLMs in design protocol analysis . . . . . 73
5.4 Potential Applications of LLMs in Broader Qualitative Research Domains . 74

**6 Conclusion** **76**

**Bibliography** **79**

**APPENDICES** **95**

**A System Prompt** **96**

A.1 Complete System Prompt . . . . . 96
A.2 System Prompt With Only Definitions . . . . . 100

**B Experiment 5** **104**

B.1 Subgroup Question Distribution . . . . . 104
B.2 Label Alignment Distribution . . . . . 105

**C Experiment 6** **108**

C.1 User Prompt With No Prompt Examples . . . . . 108
C.2 Nemenyi Test Results . . . . . 109

# List of Figures


4.1 Complete transcript with Qs identified & labeled . . . . . 36

4.2 Complete transcript with Qs identified, but unlabeled . . . . . 36

4.3 labeled question, with "chunk" of context surrounding it . . . . . 40

4.4 Unlabeled question, with "chunk" of context surrounding it . . . . . 40

4.5 labeled Qs, with utterance immediately after . . . . . 43

4.6 Unlabeled Qs, with utterance immediately after . . . . . 43

4.7 Alignment comparison across different samples . . . . . 51

# List of Tables


3.1 Question categories in Eris' (2004) Taxonomy . . . . . 17

3.2 Dataset $d_1$ characteristics. . . . . 20

4.1 Experiment 1 results. . . . . 28

4.2 Experiment 2 results . . . . . 32

4.3 Experiment 3 results . . . . . 34

4.4 Experiment 4b results . . . . . 41

4.5 Experiment 4c results . . . . . 45

4.6 Experiment 5 results - performance across categories. . . . . 50

4.7 Experiment 5 results - performance across subcategories. . . . . 52

4.8 Experiment 6 results. . . . . 55

4.9 Experiment 7a results - performance on $d_2$. . . . . 58

4.10 Experiment 7b results - performance on $d_1$. . . . . 59

4.11 Experiment 8a results - GPT-4 vs. Claude 3.5 on $d_1$ . . . . . 61

4.12 Experiment 8b results - GPT-4 vs. Claude 3.5 on $d_2$ . . . . . 62

B.1 Subgroup Question Distribution . . . . . 104

B.2 Label Alignment Summary of Categories . . . . . 105

B.3 Label Alignment Summary of Subcategories . . . . . 105
B.4 Subcategory Label Alignment Breakdown for Six 10-Question Subsets . . . 106
B.5 Subcategory Label Alignment Breakdown for Single 60-Question Set . . . . 107

C.1 Experiment 6 - Nemenyi Test Results . . . . . 109

# Chapter 1

# Introduction

Generative Artificial Intelligence (AI) represents a transformative shift in artificial intelligence, characterized by its ability to create new content, including text, images, and music, by learning from existing data. Unlike traditional AI systems designed for specific tasks, generative AI models have the capacity to generate human-like text and creative outputs, making them highly versatile tools. The evolution of generative AI can be traced through various key milestones, starting with early rule-based systems and progressing to advanced machine learning models. Notable advancements include the development of Generative Adversarial Networks (GANs) (Goodfellow et al., 2014) and transformer models (Vaswani et al., 2017), culminating in sophisticated language models such as GPT-4 (Achiam et al., 2023). The development of Large Language Models (LLMs) represents a major advancement in generative AI. Progressing from earlier versions to the latest models, LLMs have increased in scale and improved in their ability to interpret and generate text. These models are notable for their enhanced contextual interpretation, diverse language task capabilities, and improved coherence and relevance in outputs, achieved through extensive training on diverse datasets. A particularly powerful feature of modern LLMs is in-context learning (ICL) (Brown et al., 2020), which allows the model to understand and respond to new tasks and contexts based on a few examples provided within a prompt (few-shot prompting), without requiring retraining. This feature significantly enhances the flexibility and adaptability of LLMs, making them even more effective in generating contextually appro-

priate and detailed responses. Consequently, LLMs are proving to be versatile tools across various applications, including design research (Dai et al., 2023; Eschrich and Sterman, 2024; Dunivin, 2024).

AI has emerged as a promising tool in qualitative research. Recent studies have highlighted AI's effectiveness in improving qualitative research methodologies, such as grounded theory and thematic analysis (Sinha et al., 2024; Perkins and Roe, 2024). AI has been shown to significantly enhance coding accuracy and efficiency, surpassing human performance in some cases, and facilitating more complex analyses (Arlinghaus et al., 2024; Torii et al., 2024; Lennon et al., 2021). As AI continues to evolve, its integration into qualitative research promises to push the boundaries of what is possible in data analysis and interpretation.

Protocol analysis (Ericsson and Simon, 1984) is a qualitative research method extensively used in design studies to capture and analyze the cognitive processes of designers during the design process. This method involves recording participants as they work on tasks and then analyzing these recordings to understand their thought processes, decision-making strategies, and problem-solving approaches (Ericsson and Simon, 1993). Protocol analysis has been pivotal in uncovering insights into how designers think and work, contributing significantly to our understanding of design cognition. While the methods of protocol analysis have provided valuable insights into design cognition, they come with several limitations. These methods are time-consuming and labor-intensive, often requiring extensive manual transcription and coding of verbal data (Cross et al., 2001). The manual nature of these tasks introduces a significant potential for human error and variability, which can affect the reliability and reproducibility of the results (Ericsson, 2006). Furthermore, the scalability of these methods is limited, making it challenging to analyze large datasets or conduct longitudinal studies that require the continuous collection and analysis of data over extended periods.

Given the labor-intensive nature of the protocol analysis, there is a compelling need for automation to increase efficiency and accuracy. Advanced Natural Language Processing (NLP) models like GPT-4 offer a promising solution by automating the transcription and coding processes, thereby reducing the time and effort required. GPT-4's ability to interpret and generate human-like text can be leveraged to transcribe and analyze design

protocols more efficiently, enabling researchers to handle larger datasets (Pelaez et al., 2024) and minimize the potential for human error (Ericsson and Simon, 1993). Previous research has demonstrated the effectiveness of earlier GPT models in tasks such as text classification and summarization, highlighting their potential in managing large volumes of textual data (Radford et al., 2019). The integration of GPT-4 into protocol analysis not only streamlines the workflow but also opens up new possibilities for real-time analysis, providing immediate feedback and insights during design sessions (Achiam et al., 2023). This research is motivated by the broader research goal of determining the extent to which LLMs can assist in the process of analysis of verbal protocols of design. Specifically, this study aims to address this goal within the context of one existing framework for analyzing verbal protocols of design: Eris (2004) question asking taxonomy. This taxonomy categorizes questions into Low-Level Questions (LLQs), Deep Reasoning Questions (DRQs), and Generative Design Questions (GDQs). It has previously been used in design research to study idea generation (Cardoso et al., 2016) and feedback in design reviews (Cardoso et al., 2014; Hurst et al., 2021). Additionally, the deductive nature of the coding task is considered a good match for LLMs like GPT-4. This is because deductive tasks require applying clear rules and logic to classify information, which aligns well with the strengths of LLMs (Brown et al., 2020). Within this scope, the primary research objective of this study is to evaluate the performance of GPT-4 (the latest GPT variant[1]) in classifying questions within verbal protocols of design, using Eris' (2004) taxonomy as a framework. The specific research questions are:

- **RQ-1**: How does the ICL influence the labeling accuracy of GPT-4 in automated question classification?
- **RQ-2**: How consistently does GPT-4 classify questions in design protocols across different instances?
- **RQ-3**: Does incorporating contextual information enhance GPT-4's performance in automated question classification?

[1] https://platform.openai.com/docs/models/gpt-4o, accessed on 15th July 2024

- **RQ-4**: How generalizable are the prompts used for question classification across different datasets or when used with other LLMs?

By addressing these research questions, this study provides a comprehensive examination of how GPT-4 can be optimized for automated question classification. This study explores the impact of ICL methods on model accuracy, assesses the consistency of classifications across different instances, investigates the role of contextual information in enhancing performance, and evaluates the generalizability of classification prompts across datasets and models. This analysis sheds light on the specific factors that influence GPT-4's effectiveness in this application, offering valuable insights for researchers aiming to refine AI-driven tools for protocol analysis and ensuring their adaptability and reliability across diverse contexts.

The rest of the paper is structured as follows. The following chapter begins with a review of the literature, which includes an examination of key concepts and methodologies in protocol analysis, the role of AI in design research, and the development and capabilities of LLMs. The third chapter details the research methodology used to achieve the research objectives. It describes the design of experiments, tools, datasets, and the automation process. Subsequently, a comprehensive account of the experiments conducted will be presented, including a step-by-step description of the experimental design, procedures followed, and specific configurations tested. The results of these experiments will be analyzed in depth, highlighting the performance, accuracy, and any notable observations. The final chapters offer a thorough discussion of the experimental results, providing insights into their implications, significance, and limitations of the study.

# Chapter 2

# Background & Related Work

The purpose of this literature review is to provide a comprehensive overview of the existing research related to the use of generative AI, specifically GPT-4, in design research and protocol analysis. This review aims to identify key studies, theories, and findings that have shaped the current understanding of these topics. Additionally, it seeks to highlight gaps in the current knowledge and propose areas where further research is needed. By synthesizing the relevant literature, this review sets the stage for the empirical research presented in subsequent chapters of this thesis.

## 2.1 Overview of Generative AI

Generative AI has evolved significantly since its inception, marked by several key milestones. The early days of generative AI were characterized by rule-based systems and simple probabilistic models. However, the advent of neural networks (NNs) and deep learning transformed the field. NNs are computational models inspired by the human brain, composed of interconnected layers of "neurons" or units that process information in complex, non-linear ways. Data is passed through these layers, with each layer extracting increasingly abstract features. The training process, known as backpropagation (Rumelhart et al., 1986), adjusts the weights of these connections to minimize errors between the

network's output and the actual result, enabling the network to learn complex patterns from large datasets. Deep learning, a subset of machine learning, utilizes multi-layered NNs to analyze vast amounts of data, enabling these models to automatically identify intricate patterns and generate high-quality content. One of the earliest breakthroughs was the introduction of GAN (Goodfellow et al., 2014), which employs two NNs in a competitive setting—a generator and a discriminator—to create realistic images.

Another significant milestone was the development of Transformers, a specific type of deep learning architecture, introduced by Vaswani et al. (2017), which excels in handling sequence-based data, such as text. They rely on a mechanism called "self-attention" to weigh the importance of each word in a sequence relative to others. This allows transformers to capture contextual relationships over long sequences, without the limitations of sequential processing found in earlier models like recurrent neural networks (RNNs) (Sherstinsky, 2020). Transformers use layers of self-attention and feed-forward NNs, enabling parallel processing and efficient handling of large datasets, making them particularly powerful in natural language processing tasks. This architecture laid the foundation for subsequent advancements in NLP and generative AI, including the development of the LLMs. Early models struggled with maintaining coherence over long texts and generating contextually relevant content. However, the introduction of the transformer architecture and the scaling of model parameters have significantly improved these aspects. In recent years, LLMs have gained immense attention due to their impressive performance on a wide range of NLP tasks. Models like BERT (Devlin et al., 2019) and GPT-3 (Brown et al., 2020) have demonstrated remarkable capabilities in generating human-quality text, translating languages, writing different kinds of creative content, and answering your questions in an informative way. While these advancements have significantly pushed the boundaries of generative AI, challenges remain. Issues such as bias, misinformation, and ethical considerations continue to be areas of active research and discussion (Zohny et al., 2023; Schlagwein and Willcocks, 2023; OpenAI, 2023b).

## 2.2 Applications of Generative AI in Qualitative Research

The integration of Generative AI, particularly LLMs, has begun to influence qualitative analysis. This influence is evident in various studies that highlight both the strengths and limitations of these technologies across different research contexts. A key area where LLMs have shown considerable potential is in enhancing traditional qualitative research methodologies. For example, Sinha et al. (2024) utilized GPT-4 in grounded theory analysis of a classroom transcript, where a teacher introduces students to scientific theory-building practices. Their study aimed to investigate how GPT-4 can aid the initial stages of grounded theory analysis and found that, while GPT-4 showed promise as a supportive tool for enriching qualitative analysis with nuanced insights, token size limitations, output variability, and the risk of shallow interpretation highlight the need for a cautious, hybrid approach involving close researcher involvement. Perkins and Roe (2024) analyzed AI policies using GPT-4 alongside manual review, highlighting that while GenAI accelerated theme identification, it lacked the nuanced depth of human analysis and posed challenges like hallucinations and replicability issues. This emphasizes GenAI's role as a supportive tool that complements, rather than replaces, human interpretation. Similarly, Arlinghaus et al. (2024) introduced an LLM-Assisted Inductive Categorization (LAIC) method which leverages GPT models to generate and assign categories more efficiently than traditional human coding. LAIC, particularly effective with GPT-4 at low temperature[1]settings, provide consistent, high-quality results, significantly saving time on large datasets, such as survey responses or interview statements, further advocating for the integration of LLMs to streamline qualitative coding procedures.

The role of LLMs in enhancing Human-Computer Interaction (HCI) research was explored by Torii et al. (2024), who established a framework for evaluating the performance of LLMs in qualitative data analysis. Their findings suggest that LLMs not only match the efficacy of traditional methods but also offer unique insights, paving the way for methodological innovation in qualitative research. Complementing this, Feuston and Brubaker

[1]The term "temperature" is discussed later in the thesis (see the Methodology chapter, subsection 3.3.)

(2021) provided a nuanced view of human-AI collaboration, emphasizing that the timing and perspective of AI integration significantly influence its acceptance and utility by qualitative researchers. This perspective is crucial for ensuring that AI tools are effectively tailored to augment rather than automate the analytic processes in qualitative research.

Gamieldien et al. (2023) explored how NLP and LLMs like GPT-3.5 can streamline qualitative data coding. By comparing traditional manual coding with NLP-assisted approaches (NLP Cluster Assisted and GPT-3.5-enhanced coding) on a dataset of 3,800 engineering students' reflections, the study found both NLP methods successfully identified similar themes as manual coding. GPT-3.5 provided more granular insights, potentially enhancing depth. Despite challenges, such as model biases and resource demands, the findings highlighted the advantages of incorporating NLP and LLMs into qualitative research. Christou (2023) suggested that while AI can be valuable in qualitative research, it requires clear guidelines for ethical and effective use. Researchers must actively engage with and review AI outputs, using their expertise to add depth and mitigate biases. This active involvement can enhance the rigor of AI-assisted analysis, especially in thematic and content analysis.

Additionally, Zhang et al. (2023) developed *QualiGPT*, a specialized tool built on GPT technology, designed to address the limitations in traditional qualitative coding and existing qualitative analysis software. *QualiGPT* enhances the efficiency, transparency, and credibility of the qualitative analysis process by providing an intuitive, accessible, and cost-effective solution. This tool served as a practical example of how LLMs can be adapted to meet specific research needs while addressing concerns about transparency and credibility in AI-driven analysis. By easing usability and operational barriers, *QualiGPT* opens opportunities for the broader adoption of AI in qualitative research. Moreover, the potential for automating qualitative analysis through AI was demonstrated by Lennon et al. (2021), who introduced *AQUA*, an automated qualitative assistant. *AQUA*'s ability to achieve high intercoder reliability and provide transparent, graph-theoretic topic extraction and clustering marks an important step toward more scalable and reliable qualitative analysis in qualitative research. The growing capabilities of these tools enhance the efficiency of managing large datasets and support more nuanced analyses, contributing to their utility in qualitative research.

## 2.3 Protocol Analysis in Design Research

Protocol analysis is a qualitative research method used to capture and analyze the cognitive processes of individuals engaged in complex tasks, such as design. This method involves recording participants as they verbalize their thoughts while working on a design task, followed by transcribing and coding these verbal protocols to uncover patterns and insights into their cognitive strategies (Ericsson and Simon, 1993). The importance of protocol analysis in design research lies in its ability to provide a detailed and nuanced understanding of the mental processes underlying design activities. This method has been instrumental in advancing theories of design cognition and informing design education and practice (Cross et al., 2001).

Generally, protocol analysis involves several steps, including the collection of verbal data, transcription of the recorded sessions, and coding of the transcriptions according to predefined categories or frameworks. The coding process typically involves segmenting the transcriptions into meaningful units and assigning codes that represent different cognitive activities, such as problem-solving, decision-making, and evaluation (Ericsson and Simon, 1993). These codes are then analyzed to identify patterns and draw conclusions about the cognitive processes involved in design. Several frameworks and coding schemes have been developed for protocol analysis in design research. For example, Gero's Function-Behavior-Structure (FBS) coding scheme categorizes design activities based on their function, behavior, and structure, providing a comprehensive view of the design process (Gero, 1990). Another widely used framework is the Design Problem Solving (DPS) model, which focuses on identifying and analyzing the different problem-solving strategies employed by designers (Dorst, 2008).

The methods of protocol analysis provide a rich and detailed account of the cognitive processes involved in design, enabling researchers to gain deep insights into how designers think and work. Additionally, protocol analysis allows for the systematic examination of design behavior, which can inform the development of more effective design tools and methods (Cross et al., 2001). However, it is also associated with several challenges. One of the main challenges is the labor-intensive and time-consuming nature of the process, particularly the transcription and coding stages. Manual transcription and coding are

prone to human error and variability, which can affect the reliability and validity of the results (Ericsson, 2006). Additionally, the need for extensive training and expertise in coding schemes can limit the accessibility and scalability of this method.

Protocol analysis plays a crucial role in design research methodologies by providing a means to study the cognitive processes underlying design activities in a structured and systematic manner. This method allows researchers to evaluate how designers think and behave during tasks and assess the impact of various interventions. Recent reviews have highlighted its broad application in studying conceptual design (Hay et al., 2017), problem framing (Kelly and Gero, 2022), and design learning (Litster and Hurst, 2021). The insights gained from protocol analysis have contributed to the development of theories and models of design cognition, which in turn inform design education and practice. In design education, protocol analysis has been used to assess the effectiveness of different teaching methods and curricula by examining how students approach and solve design problems (Cardella et al., 2008). For example, studies have used protocol analysis to compare the cognitive processes of novice and expert designers, providing insights into the development of design expertise and informing the design of instructional interventions (Song and Becker, 2014; Dixon and Bucknor, 2019).

## 2.4 Automation in Protocol Analysis

The pursuit of automating protocol analysis has been driven by the desire to alleviate the labor-intensive nature of traditional methods. Early attempts at automation focused on developing software tools to assist with the transcription and coding processes. These initial efforts included the use of speech recognition technologies to transcribe verbal data automatically (Matheson, 2007). However, the accuracy of these systems was often limited, and they required significant manual correction, which diminished their overall effectiveness. Another early approach involved the development of coding software that could assist researchers in organizing and analyzing large volumes of qualitative data. Tools such as *ATLAS.ti* and *NVivo* were designed to facilitate the coding process by allowing researchers to tag segments of text with predefined codes and search for patterns within

the data (Friese, 2019). While these tools improved the efficiency of data management, the actual coding still relied heavily on manual input from researchers, limiting the extent of automation (Walsh et al., 2015). Moreover, the complexity of design cognition posed a challenge for automated coding systems. The nuanced and context-dependent nature of design thinking required sophisticated understanding and interpretation, which early systems lacked. These systems often failed to capture the subtleties of cognitive processes and struggled with the high variability and ambiguity inherent in qualitative data (Seawright and Gerring, 2008).

Recent advances in artificial intelligence, particularly in NLP, have significantly improved the prospects for automating protocol analysis. More recently, LLMs have shown promise for revolutionizing the way researchers analyze and interpret textual data (Mogavi et al., 2024; Bano et al., 2023). Powered by advanced NLP techniques, LLMs can comprehend and generate human-like text, offering researchers a versatile means of exploring and understanding qualitative information. These models, often pre-trained on vast corpora of diverse language data, can capture intricate linguistic nuances, making them particularly adept at uncovering underlying patterns, sentiment, and context within qualitative data (Zhao et al., 2023).

AI integration in protocol analysis has taken several forms. One notable approach is the use of AI-powered transcription services that leverage advanced speech recognition and NLP algorithms to transcribe verbal data with high accuracy. Services like Google's *Speech-to-Text* and Microsoft's *Azure Speech* provide reliable transcription capabilities, significantly reducing the need for manual correction (Raffel et al., 2020). Another significant advancement is the application of AI for automated coding and analysis of transcribed data. Language models such as GPT-4 can be guided through ICL to recognize and categorize different cognitive activities based on predefined coding schemes. This allows for a more efficient and consistent coding process, as the AI can handle large volumes of data and maintain high levels of accuracy and reliability (OpenAI, 2023b).

Several studies have demonstrated the effectiveness of AI in automating protocol analysis. For instance, a study by Zhang et al. (2024a) applied AI-powered tools to automate the identification and coding of protocols. The results showed that the AI system could accurately categorize cognitive activities and streamline the analysis process, significantly

reducing the time and effort required for manual analysis. Another study by Li et al. (2024a) explored the use of AI in analyzing students' reasoning activities. The AI system was able to identify key activities based on the linguistic features of students' think-aloud transcripts. The success of these studies highlights the potential of AI to transform protocol analysis in qualitative research. By automating the transcription and coding processes, AI could enable researchers to handle larger datasets, conduct more comprehensive analyses, and derive deeper insights into design cognition.

## 2.5 Using GPT-4 for Protocol Analysis

As we delve deeper into the landscape of language models, one standout example is ChatGPT (Chat Generative Pre-trained Transformer), a conversational variant of LLMs built by OpenAI. Its ability to engage in dynamic and context-aware conversations opens new avenues for researchers to interact with and explore qualitative data in a more conversational manner. ChatGPT relies on a pre-trained generative language model that generates responses by identifying patterns provided by the user as input (Ray, 2023). Hence, the language model essentially assigns probabilities to each word within a vocabulary that could follow a specified input sequence. These word embeddings (Mikolov et al., 2013) are developed through artificial neural networks, learning a probability distribution from provided texts in an unsupervised manner, i.e., without the need for additional human input or labeling. The production of the output sequence takes into account the tokens from the input sequence, their positions, and the previously generated output. This process is referred to as autoregressive generation (Jurafsky and Martin, 2023). Large sets of training data are required to train these LLMs. As these models have grown in sophistication and depth, their ability to generate coherent and contextually relevant text has advanced significantly.

GPT variants (e.g., GPT-3.5, GPT-4) are now being tested in various inductive and deductive qualitative analysis applications with moderate success. Hamilton et al. (2023) explored the use of ChatGPT for performing thematic analysis on interview data and found some overlap between human and AI-generated themes. Similarly, Paoli (2023)

demonstrated how GPT-3.5 can be used to partially replicate the steps in inductive thematic analysis prescribed by Braun and Clarke (2006). Xiao et al. (2023a) explored LLMs for deductive analyses and used GPT-3 in conjunction with an expert-drafted codebook to complete a coding task and report "fair to substantial agreement" with expert results. Early reports suggest that AI performs better on deductive than inductive analyses (Siiman et al., 2023). Key features of GPT-4 that are relevant to protocol analysis include its advanced language modeling capabilities, its ability to handle large volumes of text data, and its proficiency in identifying and categorizing various cognitive activities. These capabilities enable GPT-4 to automate the transcription, coding, and analysis of verbal protocols, thereby streamlining the traditionally labor-intensive process of protocol analysis (Achiam et al., 2023).

## 2.6 Summary of the Literature

The literature review has provided a comprehensive overview of the existing research related to the use of GPT-4 and other generative AI technologies in qualitative research. It has highlighted the evolution and advancements in generative AI, demonstrating how these technologies have been increasingly integrated into qualitative research to enhance usability, efficiency, and innovation. The review discussed the significant capabilities of GPT-4, particularly in automating protocol analysis. Studies have shown that LLMs such as GPT-4 can effectively transcribe and code verbal data, and generate concise summaries. Despite these advancements, the review identified several limitations in the current research, including the dependency on large datasets, challenges in integrating AI with expert knowledge, and a lack of long-term impact studies.

The findings from the literature review have several implications for the current research. First, the demonstrated capabilities of GPT-4 in automating protocol analysis provide a strong foundation for exploring its use in this context further. By leveraging GPT-4's advanced NLP capabilities, the research aims to enhance the efficiency and accuracy of protocol analysis in design studies, addressing one of the critical gaps identified in the literature (Rodgers et al., 2023). Additionally, the review underscores the importance

of developing methods to integrate AI tools seamlessly into qualitative research.

# Chapter 3

# Methodology

## 3.1 Scope of the Study

This study explores the potential of LLMs in analyzing verbal protocols of design, focusing on Eris' question-asking taxonomy (Eris, 2004). Eris' interest in studying question asking in design is based on the premise that question asking during designing influences how designers think, including how they think creatively, make decisions, and learn (Eris, 2004, p. 11). His working definition of a question in a design context is *"a verbal utterance related to the design tasks at hand that demand an explicit verbal and/or nonverbal response"* (Eris, 2004, p. 36). His taxonomy builds on prior existing taxonomies of question asking, primarily work by Lehnert (1978) and Graesser et al. (1988); Graesser and Person (1994), and classifies questions into one of three broad categories:

- Low-Level Questions (LLQs), are posed to clarify information gaps in the design project or its processes during discussions (e.g., "Is this manual pushing or motor pushing?"). This type of question can be broken down into eight subcategories: verification, disjunctive, definition, example, feature specification, concept completion, quantification, and comparison. These questions encourage convergent thinking, where the respondent can recall the correct answer.

- Deep Reasoning Questions (DRQs), where the asker is seeking to establish causal explanations of phenomena (e.g., "How was the candy being ejected from your machine?"). They can be further divided into the subcategories of interpretation, goal orientation, causal antecedent, causal consequence, expectations, procedural, enablement, and judgemental. The premise of DRQs, as in the case of LLQs, is that the answer to the question exists and is known by the question receiver.

- Generative Design Questions (GDQs), aim to facilitate the generation of many possible answers, regardless of how likely they are to be deemed feasible or true. Proposal/negotiation, method generation, scenario creation, ideation, and enablement are subcategories that fall under this category. This type of questioning, which stimulates divergent thinking, invites the respondent to evaluate potential responses and pinpoint those that meet defined criteria. According to Eris (2004), this iterative process of design activity along with the process of exploration, comparison, and evaluation constitute a crucial component of design cognition. For instance, when seeking to generate alternative options for addressing a goal or obstacle, a designer might ask, "How do we animate this?"

Table 3.1 presents the question categories and sub-categories in Eris' (2004) taxonomy. This taxonomy provides a useful lens for uncovering patterns of convergent (LLQ, DRQ) and divergent (GDQ) thinking in design. LLQ and DRQ share the common premise that a specific answer, or a specific set of answers, exists, facilitating convergent thinking processes aimed at information seeking (i.e., LLQ) or establishing causal explanations of phenomena (i.e., DRQ) (Cardoso et al., 2016). One of the primary reasons for selecting Eris' taxonomy is its proven effectiveness in previous design research studies. This taxonomy has been widely used as a coding scheme of verbal protocols of design, to study various phenomena and processes in design research.

Building on the foundational work of Eris (2004), Aurisicchio et al. (2010) and Aurisicchio et al. (2013) demonstrated that design questions reveal core activities in design by distinguishing between information acquisition, which supports design thinking, and information processing, which drives reasoning and problem-solving. In the context of

Table 3.1: Classification of the types of questions by Eris' (2004) Taxonomy.

| **Question Class** | **Sub-category** |
|---|---|
| Low-level questions (LLQ) | Verification |
| | Definition |
| | Example |
| | Feature specification |
| | Concept completion |
| | Quantification |
| | Disjunctive |
| | Comparison |
| | Judgemental |
| Deep reasoning questions (DRQ) | Interpretation |
| | Rationale/Function |
| | Causal antecedent |
| | Causal consequent |
| | Expectational |
| | Procedural |
| | Enablement |
| Generative design questions (GDQ) | Proposal/Negotiation |
| | Enablement |
| | Method generation |
| | Scenario creation |
| | Ideation |

idea generation, Cardoso et al. (2016) analyzed how high-level questions within a multidisciplinary student group led to "inflection moments"—reflective shifts that prompted creative thinking. Similarly, Leung and Lo (2024) explored the use of AI chatbots in idea generation, finding that students often preferred a question guide based on Eris' taxonomy and the *SCAMPER* technique (Boonpracha, 2023), which encouraged more innovative, perspective-shifting questions over the generic suggestions provided by ChatGPT. This

taxonomy has also been applied in studying design reviews. Cardoso et al. (2014) examined student-instructor interactions, noting that instructors tended to provide directives rather than using questions to foster deeper student analysis and self-reflection. Additionally, Hurst et al. (2021) used Eris' taxonomy to analyze questions posed by students during asynchronous, written peer critiques in a virtual learning environment. The study highlighted that GDQ—expanding the design space—was most common in this setting, underscoring asynchronous, written critiques as an effective mode for enhancing meaningful feedback exchanges among student peers. In the context of peer feedback, Cardoso et al. (2020) applied Eris' taxonomy to categorize question types in peer-to-peer design reviews, highlighting that high-performing teams used more High-Level questions (DRQ, GDQ), which promoted deeper learning. Lastly, in examining the role of expertise in feedback, Hurst et al. (2023) used Eris' taxonomy to compare the questioning styles of academic and practitioner tutors in an engineering design studio. The findings indicated that academic tutors favored convergent thinking to guide students in refining ideas, while practitioner tutors encouraged divergent thinking, promoting a broader exploration of design possibilities. Together, these studies underscore the versatility of Eris' taxonomy in capturing cognitive and collaborative processes across diverse design contexts.

Additionally, the deductive nature of the coding task was deemed well-suited to the GPT tool due to the inherent strengths of language models like GPT-4 in processing structured information and applying predefined categories to textual data (Chew et al., 2023; Xiao et al., 2023b). Deductive coding involves applying a predetermined set of codes to qualitative data based on existing theories or frameworks, which in this case is Eris' question-asking taxonomy. This approach aligns well with the capabilities of GPT-4 (Achiam et al., 2023) because the model can be guided through ICL to recognize and classify text according to the specific categories defined by the taxonomy. This study focused on classifying questions into these three broad categories. By concentrating on these overarching categories—LLQ, DRQ, and GDQ—this study aimed to assess the effectiveness of GPT-4 in distinguishing between these fundamental types of inquiry. This approach allowed us to explore the model's capabilities in a focused yet comprehensive manner, ensuring that the classification process remained manageable and directly relevant to the core cognitive processes in design thinking.

In this chapter, the dataset used in the study and the configurations of GPT-4 employed for the classification tasks will first be described. Following this, an overview of the experiments will be provided.

## 3.2 Dataset

This study utilized two datasets, $d_1$ and $d_2$, collected from two prior studies (Hurst et al., 2023; Cardoso et al., 2020). All experiments utilized the dataset $d_1$, with the exception of Experiments 7 and 8, which employed both $d_1$ and $d_2$.

For dataset $d_1$, the study involved teams of two to three students enrolled in a graduate engineering design program who met with their tutor weekly over six weeks to discuss a term-long design project. In this design project, student teams worked on developing an enhanced automatic candy-wrapping machine. Each session was audio and video recorded. The audio recordings were transcribed with an AI-based tool (Otter). The generated transcripts were manually checked to ensure quality. A Python script was created to extract all sentences ending with question marks, followed by a manual review to identify any mislabeled question sentences, correcting false positives and false negatives. Then, two research assistants independently classified each question into one of three categories—LLQs, DRQs, and GDQs—and their subcategories using Eris' taxonomy (Eris, 2004). Any coding disagreements were resolved with the assistance of a third, more experienced, researcher. The overall agreement between the two coders was substantial (Cohen's $k = 0.72$), when calculated at the broader level of the three main question classes (LLQs, DRQs, and GDQs). The dataset (Table 3.2) consisted of a total of 31 transcribed tutoring sessions, with each speaker's utterances labeled and timestamped. In addition, all question utterances (2036 in total) were identified and coded (labeled) as LLQ, DRQ, or GDQ, which also included the complete transcripts of the dialogues in which the questions were uttered.

Table 3.2: Dataset $d_1$ characteristics.
For each session (S) and group (G), session length in mins and (words), and distribution of questions by category – LLQ, DRQ, GDQ are included. E.g., the cell S1, G1 denotes the first session details of group 1: 67.50 minutes, 6472 words, and the question distribution – 95 LLQ, 9 DRQ, and 25 GDQ.

| | S1 | S2 | S3 | S4 | S5 | S6 |
|---|---|---|---|---|---|---|
| G1 | 67.50<br>(6472)<br>[95, 9, 25] | 38.03<br>(1945)<br>[45, 7, 11] | 20<br>(255)<br>[5, 1, 5] | 94.93<br>(10299)<br>[131, 11, 44] | 72.90<br>(4852)<br>[77, 8, 23] | n/a |
| G2 | 65.37<br>(19369)<br>[51, 0, 32] | 72.22<br>(21112)<br>[55, 2, 36] | 66<br>(18991)<br>[50, 5, 42] | 83.93<br>(28539)<br>[70, 4, 36] | 73.92<br>(9191)<br>[49, 5, 20] | n/a |
| G3 | 45.80<br>(10712)<br>[54, 6, 13] | 23.58<br>(2346)<br>[27, 1, 8] | 38.22<br>(5187)<br>[36, 6, 18] | 31.20<br>(3838)<br>[26, 8, 15] | 30.07<br>(3110)<br>[29, 4, 11] | n/a |
| G4 | 50.45<br>(26600)<br>[60, 15, 12] | 66.92<br>(21760)<br>[68, 1, 22] | 66.58<br>(8994)<br>[64, 15, 8] | 58.27<br>(4048)<br>[45, 15, 4] | 72.17<br>(21551)<br>[58, 13, 25] | n/a |
| G5 | 99.15<br>(13690)<br>[29, 1, 28] | 50.52<br>(3364)<br>[25, 5, 10] | 66.13<br>(3392)<br>[23, 2, 13] | 78.10<br>(8241)<br>[26, 3, 17] | 70.45<br>(14274)<br>[22, 14, 19] | 68.68<br>(7454)<br>[17, 2, 19] |
| G6 | 42.62<br>(2023)<br>[38, 5, 2] | 38.30<br>(1878)<br>[17, 1, 13] | 22.82<br>(269)<br>[11, 2, 1] | 10.75<br>(382)<br>[12, 2, 2] | 11.75<br>(559)<br>[7, 1, 2] | n/a |

For the second dataset $d_2$, the study was conducted within a capstone design course, where student projects fell into two main categories: Operations Research (OR), focused on improving client operations through analytics, and Information Systems (IS), aimed at enhancing or automating existing processes. The course involved 15 student teams, each participating in three 90-minute design review meetings. These meetings were structured around the exchange of peer feedback, with the teams grouped into five *super-groups* based on the similarity of their project types. During the meetings, each team presented their

progress for approximately 15 minutes, followed by a feedback session where each team submitted a reflection document that contained the feedback from the other teams in their *super-group*, primarily in the form of questions and their overall quality rating of the feedback. Additionally, they were asked to identify the feedback (specific comments or questions) they found most valuable and to provide a written explanation of why they considered it valuable. Later, only data from review meetings for seven of the fifteen projects, all within the operations research category, was used, with questions categorized using Eris' question-asking taxonomy (Eris, 2004). This subset was selected because the feedback content in review meetings can vary significantly depending on the project type, especially between OR and IS categories. For instance, feedback on IS projects frequently focuses on new product features, a topic less common in OR projects. The coding process involved two independent coders with substantial agreement (Cohen's $k = 0.65$) at the broader question classification level (three categories) and moderate agreement (Cohen's $k = 0.55$) at the more granular level (subcategories). Of the 265 questions comprised in the feedback utterances, 199 (75.1%) of the questions asked were LLQ, 46 (17.36%) were DRQ, and 20 (7.54%) were GDQ.

It is worth noting that the second dataset $d_2$ was considered to be of better quality due to a key difference: the students had taken note of these questions on their own, therefore, any low-quality questions were likely filtered out before documentation. Moreover, the study included a perceived value metric, allowing participants to evaluate the quality and effectiveness of the questions and feedback they received. This additional layer of evaluation provided nuanced insights into how different types of questions were valued and the impact they had on fostering learning and critical design thinking. Collectively, these aspects contributed to a richer, more reliable dataset.

## 3.3 Tool

At the time of this study (June 2024), GPT-4 was the latest iteration in OpenAI's lineup of generative pre-trained transformers, representing an improvement not only in its scale but also in its underlying architecture and capabilities. The experiments were carried

out utilizing the official Python Library for the OpenAI chat completion API (OpenAI, 2023). GPT-4 provides text outputs in response to its inputs which are also referred to as "prompts". The prompt is divided into two segments: *system message* and *user message*. The system message is used to set the behavior or context for the conversation. It helps set the tone for the interaction and provides high-level context that the model can use to better understand the user's intent or the desired outcome. On the other hand, user messages are the inputs or prompts given by the user to the model. GPT-4 Turbo (OpenAI, 2023b) model was used for all the experiments. In addition to GPT-4, the final experiment (Experiment 8) of this study also employed Claude 3.5 Sonnet (Anthropic, 2024a) to compare its performance against GPT-4. Claude 3.5 Sonnet has demonstrated highly competitive performance over GPT-4 across a range of benchmarks, including graduate-level reasoning (GPQA), undergraduate-level knowledge (MMLU), and coding proficiency (HumanEval), highlighting its strengths in handling complex tasks and offering valuable insights into areas where it may outperform GPT-4 in specific applications (Anthropic, 2024b; Huang et al., 2024). This inclusion provides a broader perspective on the generalizability and effectiveness of different large language models in the context of the study's objectives. Additionally, it's important to note that both the OpenAI API and the Anthropic API, used for GPT-4 and Claude 3.5 Sonnet respectively, maintain data privacy by default and do not use the data to train their models (OpenAI, 2023; Anthropic, 2024c).

The system message A.1 provided an overview of the chosen taxonomy, including both high-level definitions of each of the major categories (LLQ, DRQ, GDQ), as well as definitions and examples of each of the sub-categories, copied from Appendix 1 in (Cardoso et al., 2016). The user prompts varied depending on the experiment and expected outcomes. The system was further fine-tuned through the *seed* and *temperature* parameters. The *seed* parameter (OpenAI, 2023) controls the system configuration, ensuring that the system will make the best effort to sample deterministically, such that repeated requests with the same seed and parameters should return the same result. However, determinism is not guaranteed due to the frequent updates made to model configurations and system settings by OpenAI. The *temperature* parameter (OpenAI, 2023) controls the variation in the output text generation. A higher *temperature* (e.g., 0.8) leads to more random and diverse responses, while a lower *temperature* (e.g., 0.2) produces more deterministic and fo-

cused outputs. In summary, while the seed parameter promotes consistency in outputs by aiming for repeatability, the temperature parameter adjusts the creativity and variability of the model's responses. The *seed* was set to a specific number across all the experiments and the *temperature* was set to 0 so that the model would adhere closely to the input instructions and generate output with minimal variation.

## 3.4 Tasks

A series of experiments were designed and conducted to systematically evaluate the effectiveness of GPT-4 in classifying questions within design protocols. These experiments were crafted to address the specific research questions outlined earlier and to explore the various factors that could influence the model's performance. The experiments are categorized into preliminary experiments, which focus on establishing baseline performance and understanding foundational elements, and extended experiments, which examine scalability and generalizability.

### 3.4.1 Preliminary Experiments

The preliminary experiments are aimed at setting the initial benchmarks for GPT-4's performance in automated question classification. These experiments focus on understanding how factors like ICL, prompt data size, consistency, and context influence the model's ability to accurately classify questions.

- **Experiment 1: Baseline Performance.** This experiment is designed to determine the baseline performance of GPT-4 in classifying questions. Specifically, it seeks to answer **RQ-1**. The experiment involves classifying a test set of stand-alone question utterances, both with and without labeled examples (zero-shot and few-shot settings), to evaluate the impact of having prior labeled examples (annotated examples included in the user prompt) on classification accuracy.

- **Experiment 2: Impact of In-context Examples Count.** This experiment extends the findings of Experiment 1 by systematically varying the number of in-context labeled examples to determine how this influences GPT-4's classification performance. It directly addresses **RQ-1** by exploring whether using a large number of in-context examples improves the model's ability to accurately classify questions.

- **Experiment 3: Consistency Across Instances.** This experiment is designed to explore **RQ-2**. By running the same classification task multiple times under identical conditions, this experiment assesses the consistency of GPT-4's performance, ensuring that the results are reliable and repeatable.

- **Experiment 4: Role of Context in Classification.** This experiment investigates **RQ-3**. It examines how the inclusion of surrounding context (e.g., prior and subsequent dialogue) affects the performance of GPT-4 in classifying a given question. The goal is to determine whether providing more context enables the model to make more informed and accurate classifications or if it leads to worse results.

### 3.4.2 Extended Experiments

The extended experiments build on the preliminary findings by examining the scalability and generalizability of the GPT-4-based classification system. These experiments focus on evaluating the impact of increased test set sizes, variations in prompt structures, and cross-model applicability.

- **Experiment 5: Scaling Up Test Set.** This experiment is a continuation of the investigation into **RQ-1**. It evaluates GPT-4's performance on larger test sets by increasing the test size to 60 questions and running the classification tasks across all possible mutually exclusive test samples of the dataset $d_1$. This experiment is critical for understanding how well GPT-4 scales and maintains accuracy when dealing with more extensive and diverse datasets.

- **Experiment 6: Extended Baseline Experiment.** Further exploring **RQ-1**, this experiment tests GPT-4's performance across four prompt settings. It includes

scenarios with no system message or labeled examples (zero-shot setting), only a system message with definitions, a full system message without labeled examples, and a full system message with labeled examples (few-shot setting). This experiment aims to determine how different prompt configurations impact question classification accuracy, offering insights into the optimal setup for using GPT-4 in this context.

- **Experiment 7: Generalizability Across Datasets.** This experiment addresses **RQ-4**. In this experiment, GPT-4 was provided with in-context examples (labeled questions) from dataset $d_1$ and evaluated on test questions from $d_2$, and vice versa. Thus, this experiment assesses the robustness and transferability of the model's learned classification capabilities across different but related datasets.

- **Experiment 8: Generalizability Across LLMs.** This final experiment also explores **RQ-4** by evaluating the performance of GPT-4 in comparison with another proprietary LLM, Claude 3.5 Sonnet. The experiment involves evaluating both models on datasets $d_1$ and $d_2$ to determine how well the prompts and classification strategies generalize across different LLMs. Both models are evaluated on datasets $d_1$ and $d_2$ using few-shot learning to assess how well the prompts and classification strategies generalize across different LLMs.

Each experiment and their results are detailed in the following chapter.

# Chapter 4

# Experiments & Results

This chapter presents the detailed experiments conducted to evaluate the performance of GPT-4 in question labeling tasks. The experiments are systematically categorized into preliminary and extended studies to establish initial benchmarks and explore the scalability and generalizability of the approach. Preliminary experiments focus on foundational aspects such as the impact of ICL, the number of in-context examples, result sensitivity, and contextual influence. Extended experiments further investigate the robustness of GPT-4 across various configurations, including larger test sets, different prompt settings, and generalizability across datasets and large language models.

## 4.1 Preliminary Experiments

The preliminary experiments serve to establish initial performance benchmarks for GPT-4 in the context of automated question classification. These experiments are specifically designed to address **RQ-1** (impact of the number of in-context examples on classification performance), **RQ-2** (consistency of classification results), and **RQ-3** (effect of contextual information on performance). By exploring these foundational aspects, the preliminary experiments provide critical insights into the model's baseline capabilities, which are essential for guiding the subsequent extended experiments.

### 4.1.1 Experiment 1

The goal of experiment 1, aligned with **RQ-1**, was to provide a baseline of GPT-4's effectiveness in labeling questions and investigate the value of in-context examples in this task. Dataset $d_1$ was used for this experiment.

In the first condition, the tool was provided with 90 stand-alone human-labeled questions as in-context examples (90-shot learning), uniformly distributed among the LLQ, DRQ, and GDQ types. In the second condition, no labeled examples (zero-shot) were provided. The classification task was conducted on a new set comprised of 30 question utterances (testing set). The questions included in the testing set were randomly selected from the data, with the constraint that they were also uniformly distributed among the LLQ, DRQ, and GDQ types. The experiment was run once using the system prompt described in Appendix A.1. The exact user prompt is provided below:

> *Classify each of the questions below, delimited by triple backticks, using the taxonomy proposed by Eris. Label each question with one of the three categories: Low-level questions, Deep Reasoning Questions, or Generative Design Questions. Delimit the label with triple backticks. State your reasoning for the assigned label. Format the result as a markdown table.*
>
> ``` < *test set* > ```
>
> *To help you categorize the questions above, here are some examples delimited by triple backticks, each line contains an example that has two segments - question and category separated by colon (:)*
>
> ``` < *in-context examples* > ```

Table 4.1 contrasts the results of the labeling task under the two conditions (without and with the in-context examples) with the human-labeled set. GPT-4 generated labels aligned more closely with the human labels when a prompt dataset was provided: 83% (25 out of 30) compared to 60% (18 out of 30) alignment. Errors unique to the no-few shot learning condition highlighted the GPT-4's reliance on general patterns when specific prompt data is absent, leading to misclassifications in more complex or domain-specific scenarios like

Table 4.1: Experiment 1 results.
Labels assigned by: "H" → Human; "Zero-shot" → GPT-4, without in-context examples; "Few-shot" → GPT-4, with 90 in-context examples.

| # | Test set: Question Utterance | H | Zero-shot | Few-shot |
|---|---|---|---|---|
| 1 | So how do you know how much do? | DRQ | LLQ | DRQ |
| 2 | ...what are the possible contributors? | GDQ | GDQ | GDQ |
| 3 | Why would you put that on there if you could swap in the new roll in a minute? | DRQ | DRQ | DRQ |
| 4 | What about cleats, right cleated conveyor cleats? | GDQ | GDQ | GDQ |
| 5 | But so it's full 15 or whatever, probably right? | LLQ | LLQ | LLQ |
| 6 | Could you take the candy and slide it on the wrapper? | LLQ | GDQ | GDQ |
| 7 | Why would I do that? | DRQ | DRQ | DRQ |
| 8 | How they wrap candy currently? | DRQ | LLQ | DRQ |
| 9 | How does the very last one behave? | GDQ | LLQ | DRQ |
| 10 | But this whole thing has to rotate, right? | LLQ | LLQ | LLQ |
| 11 | So when they're wrapping in one of these candies, have you seen the whole process? | LLQ | LLQ | LLQ |
| 12 | Yeah, but I think, isn't it the same? | GDQ | LLQ | LLQ |
| 13 | So is the main assumption right now that the wrapper will stick? | LLQ | LLQ | LLQ |
| 14 | How are you going to intersection things that are exactly the same...? | GDQ | GDQ | GDQ |
| 15 | How are you going to keep rolling and pulling it? | GDQ | GDQ | GDQ |
| 16 | How was the candy being ejected from your machine. | DRQ | LLQ | DRQ |
| 17 | Can I do this stuff? | LLQ | LLQ | LLQ |
| 18 | Makes sense? Right? | LLQ | LLQ | LLQ |
| 19 | What kind of processes out there now? | DRQ | LLQ | DRQ |
| 20 | Because these paper will come on to the stage right? | LLQ | LLQ | LLQ |
| 21 | ...Why did you put grid? | DRQ | DRQ | DRQ |
| 22 | What function is providing? | DRQ | LLQ | LLQ |
| 23 | What's gonna happen? | GDQ | DRQ | GDQ |
| 24 | So what's happening here at the end of this conveyor belt, tell me a little bit more ...What would happen in terms of how this thing gets pushed in the plastic? | DRQ | LLQ | DRQ |
| 25 | ...What is the purpose of this box? | DRQ | LLQ | LLQ |
| 26 | And is this manual pushing? Or .. a motor pushing? | LLQ | LLQ | LLQ |
| 27 | ...How are you going to then finalize the closure? | GDQ | GDQ | GDQ |
| 28 | So how do we animate this? | GDQ | GDQ | GDQ |
| 29 | Or is it something that we can outsource to a company? | GDQ | LLQ | GDQ |
| 30 | Do you want to a machine to produce 15 to 20 pieces of candy per min and currently they're doing what is it? | LLQ | LLQ | LLQ |

in this case (Brown et al., 2020; OpenAI, 2023b). As such, all future preliminary labeling tasks (Experiments 2, 3, and 4) use in-context examples (few-shot setting).

The five questions (6, 9, 12, 22, and 25) for which the AI-assigned labels didn't match the human label were further scrutinized. For all those questions, it is believed that the AI-assigned labels are correct when considering the question in isolation, as written. For instance, consider question 6: *"Could you take the candy and slide it on the wrapper?"* Here is the same question (highlighted) surrounded by two utterances (one before and one after) where an utterance was defined as any text uttered by a participant, located between two turn-taking activities in the conversation.

> *so we need to calculate the timing of the?*
>
> *I think so what do you guys think? Is this is a valid concern? Yeah. Right. I, honestly, I don't know. Right? I, you know, how sticky is the candy?* ***Could you take the candy and slide it on the wrapper?*** *Do you have one? We had you ate them all, didn't you.*
>
> *Candy that from the cold storage. So they are not as sticky as it would be*

When the question was judged within the conversation (context), this question belonged to the LLQ category, specifically as a *Verification* type, because it sought to verify the possibility of an action related to the design task. However, when isolated from the context, this question could fall into the GDQ category, specifically under the *Proposal/Negotiation* subcategory. This is because, in isolation, the question can be interpreted as proposing an idea or solution, inviting the listener to consider the action of sliding the candy on the wrapper as a potential option. This discrepancy of mismatching labels occurred may be due to a number of reasons, beyond a simple deficiency in the tool or a mistake in the labeling by the human. In their task, in making their judgment, the human coders were also considering the larger context of the conversation, including what is discussed prior to the question, and the answer that immediately follows it. The impact of context is further explored in Experiment 4.

Also, where there was misalignment, there was often ambiguity in the correct classification for the question. For example, in some cases, the two different human coders disagreed

about the correct category in the initial coding, or we disagreed with the human-assigned category. This ambiguity highlights the inherent challenges in categorizing certain questions, which may contribute to the discrepancies observed between AI and human labels. For instance, GPT-4 labeled question 9 - *"How does the very last one behave?"* as DRQ. One of the human coders also labeled it as DRQ. However, the final human-assigned category was selected as GDQ. This discrepancy illustrates the subtle distinctions between question types and the potential for different interpretations based on context or individual understanding.

Moreover, these ambiguities highlight the importance of considering multiple perspectives when evaluating AI performance in natural language tasks. While misalignments between AI and human labels are important to identify and address, it's equally crucial to recognize that some level of disagreement may be inherent to the task itself, rather than a shortcoming of the AI system.

### 4.1.2 Experiment 2

The goal of Experiment 2, again addressing **RQ-1**, was to determine the effect of the size of the prompt dataset on the accuracy of labeling by GPT-4. Using the dataset $d_1$, the labeling task described in Experiment 1 was repeated under varying conditions, adjusting the number of in-context examples from 0 (i.e., no labeled examples) up to 300 pre-labeled questions, in increments of 30. The test set was kept constant and identical to the one in Experiment 1. The system prompt A.1 and the user prompt was kept the same as Experiment 1.

Table 4.2 presents the labeling task results for each question and each condition based on the number of in-context examples (zero-shot to few-shot setting) in the prompt. Overall, it appears that while using in-context examples improves alignment with human-generated labels, no accuracy improvements are achieved when the size is increased past 90 questions. This observation is consistent with findings by Li et al. (2023b), where adding more ICL demonstrations caused the accuracy curve of LLM to drop rapidly, and by Chen et al. (2023), who reported performance degradation with the inclusion of additional in-context examples addressing the weakness of LLMs. Similar trends were observed in the studies

by Wang et al. (2024) where adding more in-context examples appeared to have no more positive impact on the performance and by Min et al. (2022), where increasing the number of ICL demonstrations beyond a certain point brought only marginal improvements, further illustrating the challenges of scaling up the number of in-context examples for LLMs.

The lack of improvement in performance with increasing in-context examples was attributed to several factors. Li et al. (2023b) suggested that this decline could be due to the model's limitations in handling longer input sequences and the constraints imposed by the model size. Similarly, Min et al. (2022) pointed to sequence length limitations, which caused performance to plateau as the number of examples increased. Chen et al. (2023) highlighted that spurious correlations among multiple demonstrations could mislead the model, thereby reducing accuracy. On the other hand, Wang et al. (2024) argued that there is a performance upper bond for the model, which could only be surpassed through additional training, not by ICL. These findings highlight the need for further exploration and development of advanced ICL algorithms or strategies to overcome these limitations in many-shot ICL. Moreover, this plateau in performance, though observed in a different context, mirrors the findings of Kaplan et al. (2020), where neural language models trained on larger datasets reach a point of diminishing returns, with additional data providing progressively smaller performance gains. This suggests that in this experiment, GPT-4 might have sufficiently learned the necessary patterns from the initial 60 or 90 examples, rendering further increases in the number of in-context examples redundant for improved performance. By recognizing that the model's performance plateaued after 60 or 90 examples, the subsequent experiments were designed to maximize efficiency and effectiveness without unnecessarily expanding the number of in-context examples, as further increases would only add additional cost and time (e.g., few-shot setting with 180 in-context examples) without yielding significant improvements in accuracy.

### 4.1.3 Experiment 3

Although utilizing a fixed set of seed and temperature is expected to produce consistent output, responses can still differ due to the inherent non-deterministic nature of the model (OpenAI, 2023a). The goal of Experiment 3 was thus to determine the sensitivity

Table 4.2: Experiment 2 results

AI-assigned label matches human-assigned label

AI-assigned label does not match human-assigned label

| | **Number of in-context examples** | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | 0 | 30 | 60 | 90 | 120 | 150 | 180 | 210 | 240 | 270 | 300 |
| Q1 | | | | | | | | | | | |
| Q2 | | | | | | | | | | | |
| Q3 | | | | | | | | | | | |
| Q4 | | | | | | | | | | | |
| Q5 | | | | | | | | | | | |
| Q6 | | | | | | | | | | | |
| Q7 | | | | | | | | | | | |
| Q8 | | | | | | | | | | | |
| Q9 | | | | | | | | | | | |
| Q10 | | | | | | | | | | | |
| Q11 | | | | | | | | | | | |
| Q12 | | | | | | | | | | | |
| Q13 | | | | | | | | | | | |
| Q14 | | | | | | | | | | | |
| Q15 | | | | | | | | | | | |
| Q16 | | | | | | | | | | | |
| Q17 | | | | | | | | | | | |
| Q18 | | | | | | | | | | | |
| Q19 | | | | | | | | | | | |
| Q20 | | | | | | | | | | | |
| Q21 | | | | | | | | | | | |
| Q22 | | | | | | | | | | | |
| Q23 | | | | | | | | | | | |
| Q24 | | | | | | | | | | | |
| Q25 | | | | | | | | | | | |
| Q26 | | | | | | | | | | | |
| Q27 | | | | | | | | | | | |
| Q28 | | | | | | | | | | | |
| Q29 | | | | | | | | | | | |
| Q30 | | | | | | | | | | | |
| **Alignment** (%) | 60 | 67 | 83 | 83 | 80 | 83 | 83 | 73 | 70 | 73 | 83 |

of the results across multiple "runs" of the experiment, thereby addressing **RQ-2**. Using 60 and 90 in-context examples, as determined from Experiment 2, the labeling task was repeated once again on the dataset $d_1$, independently run 50 times, for each case. The system prompt A.1 and the user prompt was kept the same as Experiment 1. The number of in-context examples 60 and 90 were chosen for this experiment because they achieved the highest labeling accuracy of 83% in Experiment 2 and GPT-4's performance plateaued after 60 or 90 examples. On top of that, a large number of in-context examples with similar performance would add additional cost and time.

Table 4.3 presents the aggregated labeling results; for example, a score of 0.5 indicates that the tool selected that particular label in half of the 50 runs (i.e., 25 times). For instance, Question 15 was labeled as GDQ by the human, but when GPT-4 was provided a set of 60 examples, it labeled the question as DRQ in 48% of the runs and as GDQ in 52% of the runs. However, when the number of the in-context examples increased to 90, GPT-4 consistently labeled it as GDQ in all runs. Any labels that do not match the human label are highlighted in red. In the case of "split" labels, if one of the two labels matches the human label, they are highlighted in orange.

The results demonstrate the probabilistic nature of the labeling, with many questions being labeled differently on different runs. Of note is that the uncertainty is much larger in the task with the smaller number of in-context examples (9 questions with split labels) compared to the task with the larger number of in-context examples (only 1 question with a split label). It was thus determined that while the labeling accuracy was the same with 60 and 90 in-context examples in Experiment 2, this experiment demonstrated that the larger set of 90 in-context examples provides more robust results across multiple runs of the experiment. This robustness implies that with a larger set of in-context examples, the GPT-4's predictions are more consistent, reducing the variability across different runs. However, although the larger set of in-context examples such as with 150, 180, or 300 examples exhibited the highest performance in Experiment 2, the same performance was achieved with only 60 or 90 in-context examples. As discussed earlier in Experiment 2, this suggests that after reaching a certain threshold in the number of in-context examples, the model's performance did not improve with additional data, but rather, the benefits began to diminish. This pattern aligns with previous findings that highlight the challenges of

Table 4.3: Experiment 3 results

| # | **Human label** | | | **GPT-4, 60-shot** | | | **GPT-4, 90-shot** | | |
|---|---|---|---|---|---|---|---|---|---|
| | LLQ | DRQ | GDQ | LLQ | DRQ | GDQ | LLQ | DRQ | GDQ |
| Q1 | 1 | | | 1 | | | 1 | | |
| Q2 | | | 1 | | | 1 | | | 1 |
| Q3 | | 1 | | | 1 | | | 1 | |
| Q4 | | | 1 | | | 1 | | | 1 |
| Q5 | 1 | | | 1 | | | 1 | | |
| Q6 | 1 | | | | 0.24 | 0.76 | | | 1 |
| Q7 | | 1 | | | 1 | | | 1 | |
| Q8 | | 1 | | | 1 | | | 1 | |
| Q9 | | | 1 | | 1 | | | 1 | |
| Q10 | 1 | | | 0.86 | 0.14 | | 1 | | |
| Q11 | 1 | | | | 1 | | 1 | | |
| Q12 | | | 1 | 1 | | | 1 | | |
| Q13 | 1 | | | 0.62 | 0.38 | | 1 | | |
| Q14 | | | 1 | | 0.48 | 0.52 | | | 1 |
| Q15 | | | 1 | | 0.48 | 0.52 | | | 1 |
| Q16 | | 1 | | | 1 | | | 1 | |
| Q17 | 1 | | | 1 | | | 1 | | |
| Q18 | 1 | | | 1 | | | 1 | | |
| Q19 | | 1 | | | 1 | | | 1 | |
| Q20 | 1 | | | 1 | | | 1 | | |
| Q21 | | 1 | | | 1 | | | 1 | |
| Q22 | | 1 | | 0.66 | 0.34 | | 1 | | |
| Q23 | | | 1 | | 0.34 | 0.66 | | 1 | |
| Q24 | | 1 | | | 1 | | | 1 | |
| Q25 | | 1 | | 0.66 | 0.34 | | 0.78 | 0.22 | |
| Q26 | 1 | | | | 1 | | 1 | | |
| Q27 | | | 1 | | 0.48 | 0.52 | | | 1 |
| Q28 | | | 1 | | | 1 | | | 1 |
| Q29 | | | 1 | | | 1 | | | 1 |
| Q30 | 1 | | | 1 | | | 1 | | |
| **Total Split Labels** | | | | **9** | | | **1** | | |

scaling up the number of in-context examples for LLMs, as the performance gains become progressively smaller or detrimental (Li et al., 2023b; Min et al., 2022; Chen et al., 2023).

Thus, while a larger set of in-context examples can contribute to greater consistency and robustness, it is important to balance this with the recognition that beyond a certain point, expanding it might yield only minimal improvements. Therefore, the diminishing returns in accuracy, coupled with the increased cost and time associated with using a larger set of in-context examples, made the 90-shot prompt setting a more practical choice for balancing performance and efficiency. Thus, the number of in-context examples for stand-alone question classification was consistently maintained at 90 in the extended experiments (Experiments 5-8).

### 4.1.4 Experiment 4

In all experiments described so far, the labeling task was focused on questions isolated from their context in the conversation. However, **RQ-3** prompts us to explore whether incorporating contextual information can enhance the AI's performance. Human coders use the context of the conversation to make judgments for the appropriate labels, including both the discussion before a question is posed, and the response immediately after. For example, consider the question: *"What does the client want?"* Here is the same question (highlighted) surrounded by two utterances (one before and one after). An utterance is defined here exactly as described in Experiment 1.

> *Essentially, concept one, two, and three is just three different problem scopes. So we either we're thinking of doing each of us designing three concepts, basically one concept or just the problem space is this much, then this much.*
>
> ***What does the client want?***
>
> *He's open to the first two.*

When the question was judged within the conversation (context), this question belonged to the LLQ category, specifically as a *Concept Completion* type, because it sought to complete the understanding of the client's requirements within the design task. This example illustrates how context helps clarify the intent of the question and guides human coders in their labeling decisions, a nuance that the model might miss when questions are

isolated from their conversational background. The goal of Experiment 4 was to determine whether the GPT-4 can also use context in the labeling task and whether this results in labels that are more similar to those assigned by humans.

### Experiment 4a

In this experiment, the entire conversation in the session was used for context in both the in-context examples and test sets (both drawn from dataset $d_1$). The in-context example consisted of the entire transcript of one complete session (G1, S1), with the questions in the transcript identified, and their human-assigned labels provided (Figure 4.1). The test set consisted of the entire transcript of a separate session (G1, S2), with the questions (but not their labels) identified (Figure 4.2).

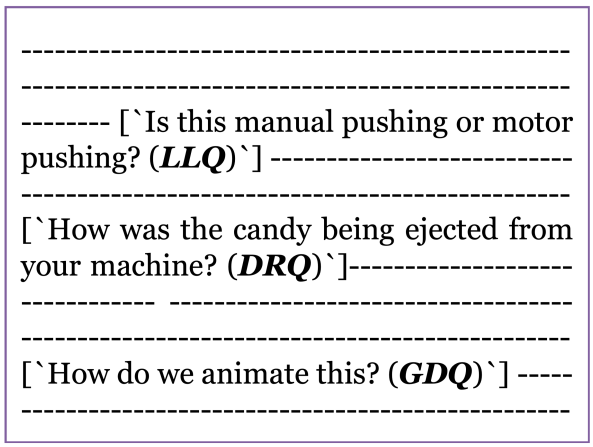



Figure 4.1: Complete transcript from one session with Qs identified & labeled

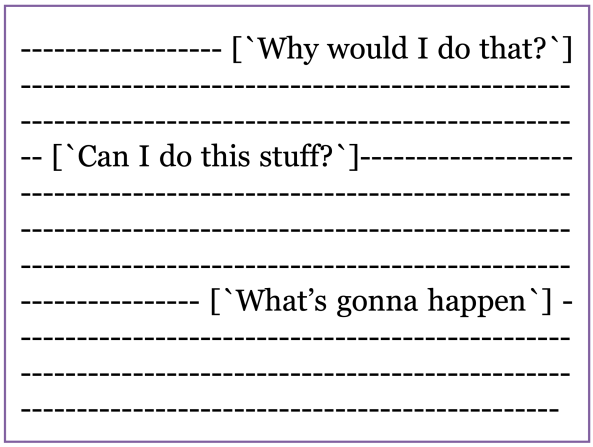



Figure 4.2: Complete transcript from a separate session with Qs identified, but unlabeled

The in-context examples and test sets were selected from the same group of participants so that the GPT-4 could learn not only from the labeled examples but also from the participants' speaking styles. The system prompt A.1 was kept the same and the exact user prompt in Experiment 4a was:

*Classify all the questions identified in the conversation below, using the taxonomy proposed by Eris. Label each question with one of the three categories: Low-level questions, Deep Reasoning Questions, or Generative Design Questions, and present the results in JSON format that contains only questions and labels. Each question is identified and kept inside a square bracket `[]` at the end of an utterance. Each line (separated by newline character "\n") represents an utterance from a person in the conversation. There can be multiple questions from a single utterance. Label as you go through the conversation. In determining an appropriate label, consider the context of the conversation, including what has been discussed before the question, and the response that follows it. Here is the conversation -*

``` < *test set* > ```

*To help you classify the questions, an example is given below which is a similar discussion between multiple subjects on the same topic. In this example, each line contains an utterance from a person in the conversation. The lines are separated by the newline character "\n". The appropriate question is identified and kept inside a square bracket and a label can be found on that question inside parentheses. If there are multiple questions from a single utterance, then those questions and associated labels are listed inside the square brackets separated by a comma. The example -*

``` < *in-context examples* > ```

The results of Experiment 4a were unexpected, revealing significant challenges when utilizing the entire conversation from a session as context in both in-context examples and test sets. Despite the clear instructions in the prompt, GPT-4 failed to label all the identified questions in the testing set, and some AI-labeled questions were mistakenly mixed with the in-context examples. These findings suggest that the inclusion of extensive contextual information, while theoretically beneficial, may introduce complexities that current LLMs struggle to manage. This outcome aligns with previous research indicating that while context is crucial for understanding and generating accurate responses, excessive context can lead to confusion in language models (Liu et al., 2024).

### Experiment 4b

Next, to address the limitations of the previous attempt, a different experiment was designed. The idea in the new experiment was to provide the GPT-4 with a smaller "chunk" of context surrounding the question. It was hypothesized that this chunk of context would improve the labeling accuracy, inspired by recent studies on LLMs. According to An et al. (2024), the *"lost-in-the-middle"* (Liu et al., 2024) problem is a significant challenge for LLMs, where crucial information within a long context can be overlooked. The authors propose that by focusing on fine-grained information within shorter segments of the context, the model's ability to utilize this information effectively is enhanced. Inspired by this approach, it was hypothesized that providing the GPT-4 with a smaller, more manageable context would mitigate some of the difficulties observed in Experiment 4a and lead to better performance.

In this experiment, the in-context examples consisted of examples of "chunks" of text from different sessions (Figure 4.3). Each chunk contained a labeled question, with a specified amount of text both immediately before and after it. Initially, it was believed that including one to two utterances before and after the question would be appropriate. An utterance is defined here exactly as described in Experiment 1. However, it was later determined that given the large variety of utterances (which could vary from one word to several paragraphs in length), a different approach was needed. Thus, approximately 50-80 words were included both before and after each question in such a way that complete sentences were included. A Python script was developed to automate the whole process. An example of one such chunk is provided below:

> *...it's basically think of two plates close. So you could attach it to a threaded bolt. And then where the bottom one or the top. Either one is fixed, and the other one rides up and down on the threaded bolt. So how do you know how much do we haven't come up with the final design of that? [`So how do you know how much do?`]*
>
> *So there's gonna well or there are parameters and then by test testing the system, you'll determine what those parameters are operational parameters are right. like how much to grip and everything...*

Two different versions of the in-context example sets were used: one with 60, and one with 90 pre-labeled questions (from dataset $d_1$). In both cases, these were the same 60/90 questions used in Experiment 3. The testing set comprised similarly constructed chunks of text (Figure 4.4), enveloping the same 30 questions used in Experiment 1, 2, and 3. The system prompt A.1 was kept the same as before and the following user prompt was used:

> *Classify the question within the 30 conversation segments listed below, using the taxonomy proposed by Eris. Label each question with one of the three categories: Low-level questions, Deep Reasoning Questions, or Generative Design Questions—and present the results in JSON format that contains both questions and labels. Each question is identified and kept inside a square bracket `[]` at the end of an utterance. Each line represents an utterance from a person in the conversation. In determining an appropriate label, consider the context of the conversation, including what has been discussed before the question, and the response that follows it.*
>
> ``` < *test set* > ```
>
> *To help you classify the questions, here are some examples. 60 discussion segments between multiple subjects separated using <example> tag are provided below. In these examples, each line contains an utterance from a person in the conversation. The appropriate question is identified and kept inside a square bracket just like before followed by the label for that question inside parentheses.*
>
> ``` < *in-context examples* > ```

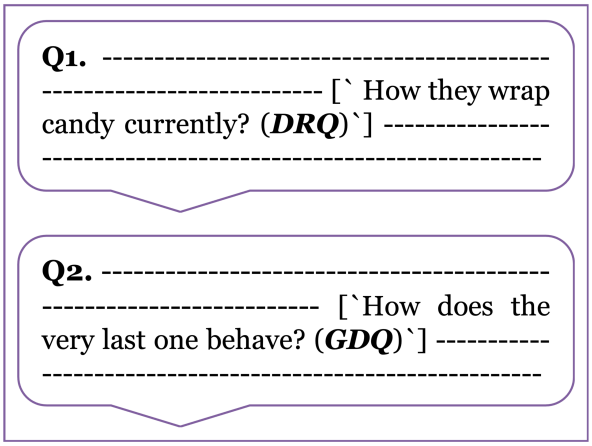



Figure 4.3: labeled question, with "chunk" of context surrounding it

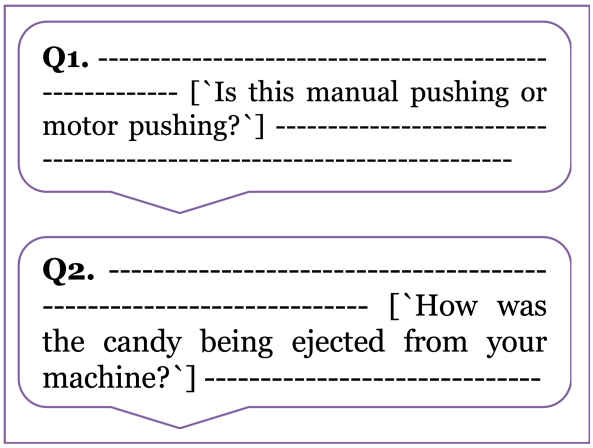



Figure 4.4: Unlabeled question, with "chunk" of context surrounding it

In both cases, approximately half of the GPT-4 generated labels were incorrect (see Table 4.4), resulting in a significantly degraded performance compared to the case when no context around the questions was provided. The results were surprising, as providing context was expected to improve labeling accuracy, yet it led to significantly degraded performance. This outcome, while unexpected, might be explained by the inconsistent length and relevance of the contextual utterances provided, which likely confused the model. The variability in the context might have overwhelmed the model, leading to incorrect label assignments. This suggests that context management is critical, and more context does not necessarily equate to better performance (Hosseini et al., 2024). Moving forward, experiments should focus on refining how context is provided, ensuring it is concise, relevant, and manageable for the model.

Table 4.4: Experiment 4b results

| **ID** | **Human Label** | **GPT-4, context NOT provided** | | **GPT-4, context provided** | |
|---|---|---|---|---|---|
| | | 60-shot | 90-shot | 60-shot | 90-shot |
| Q1 | DRQ | DRQ | DRQ | LLQ | LLQ |
| Q2 | GDQ | GDQ | GDQ | GDQ | GDQ |
| Q3 | DRQ | DRQ | DRQ | DRQ | DRQ |
| Q4 | GDQ | GDQ | GDQ | GDQ | GDQ |
| Q5 | LLQ | LLQ | LLQ | LLQ | LLQ |
| Q6 | LLQ | GDQ | GDQ | DRQ | DRQ |
| Q7 | DRQ | DRQ | DRQ | DRQ | DRQ |
| Q8 | DRQ | DRQ | DRQ | LLQ | LLQ |
| Q9 | GDQ | DRQ | DRQ | DRQ | DRQ |
| Q10 | LLQ | LLQ | LLQ | DRQ | DRQ |
| Q11 | LLQ | LLQ | LLQ | DRQ | LLQ |
| Q12 | GDQ | LLQ | LLQ | LLQ | LLQ |
| Q13 | LLQ | LLQ | LLQ | DRQ | DRQ |
| Q14 | GDQ | GDQ | GDQ | DRQ | DRQ |
| Q15 | GDQ | GDQ | GDQ | DRQ | DRQ |
| Q16 | DRQ | DRQ | DRQ | DRQ | LLQ |
| Q17 | LLQ | LLQ | LLQ | GDQ | GDQ |
| Q18 | LLQ | LLQ | LLQ | LLQ | LLQ |
| Q19 | DRQ | DRQ | DRQ | LLQ | LLQ |
| Q20 | LLQ | LLQ | LLQ | LLQ | LLQ |
| Q21 | DRQ | DRQ | DRQ | DRQ | DRQ |
| Q22 | DRQ | LLQ | LLQ | DRQ | DRQ |
| Q23 | GDQ | GDQ | GDQ | GDQ | GDQ |
| Q24 | DRQ | DRQ | DRQ | DRQ | DRQ |
| Q25 | DRQ | LLQ | LLQ | DRQ | DRQ |
| Q26 | LLQ | LLQ | LLQ | DRQ | LLQ |
| Q27 | GDQ | GDQ | GDQ | DRQ | DRQ |
| Q28 | GDQ | GDQ | GDQ | GDQ | GDQ |
| Q29 | GDQ | GDQ | GDQ | DRQ | DRQ |
| Q30 | LLQ | LLQ | LLQ | DRQ | LLQ |
| **Alignment** (%) | **n/a** | **83.33** | **83.33** | **46.67** | **53.33** |

**Experiment 4c**

Given the challenges in the previous two attempts, the experimental task was further pared down. Specifically, Experiment 4c sought to determine if the label accuracy could be improved by providing the GPT-4 utterance that immediately follows the question. The reasoning was that when humans perform the labeling task, they are trained to use the answer to a question for important clues about the intention of the question. An example is provided below -

> *[`**What's your first impression when you look at this?**`]*
>
> *This one? Oh, thank you, well it's, first of all, it's difficult to put the candy at the right place to start the process. And it has to be manual. Like the machine cannot load the candy on the cello paper automatically get that*

In this example, the question *"What's your first impression when you look at this?"* is immediately followed by an utterance that provides a detailed response about the difficulty in placing the candy correctly to start the process. The content of the answer reveals that the question is probing for an evaluation or assessment of the process, which could be beneficial for determining the correct label. By including the following utterance, the GPT-4 was expected to better infer the intent behind the question, similar to how human coders do. Using the same prompt datasets (sizes 60 and 90) from the dataset $d_1$ that were used in Experiment 2, the labeling task was repeated once again on the same test set of 30 questions, with one key difference: for both the prompt dataset (Figure 4.5) and test sets (Figure 4.6), the GPT-4 was also provided with the utterance that immediately followed the question, which was presumed to contain the response to the question. A Python script was developed to automate the whole process.

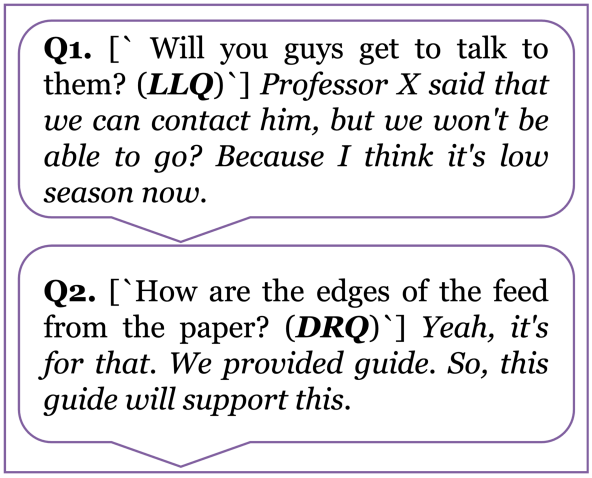



Figure 4.5: labeled questions, with utterance immediately after

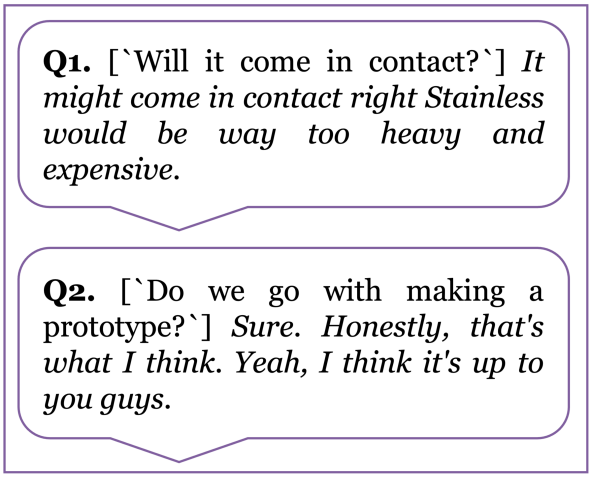



Figure 4.6: Unlabeled questions, with utterance immediately after

An utterance is defined here exactly as described in Experiment 4b. The system prompt A.1 was kept the same as before and the following user prompt was used:

> *Classify the question within the 30 conversation segments listed below, using the taxonomy proposed by Eris. Label each question with one of the three categories: Low-level questions, Deep Reasoning Questions, or Generative Design Questions—present the results in JSON format that contains both questions and labels. Also, state your reasoning for the assigned label. Each question is identified and kept inside a square bracket `[]`. Each line (separated by a newline character "\n") represents an utterance from a person in the conversation. In determining an appropriate label for the question, consider the utterance that follows it in the conversation, which is provided in the following line.*
>
> ``` < *test set* > ```
>
> *To help you classify the questions, here are some examples. 90 discussion segments between multiple subjects are provided below which are separated using <example> tag. In each example, the first line contains the question kept inside a square bracket just like before followed by the label for that question inside*

*parentheses. The second line provides the utterance that follows the question in the conversation. The lines are separated by the newline character "\n".*

``` < *in-context examples* > ```

The experiment results are presented in Table 4.5, which contrasts the label assignments in this experiment ("GPT-4, answer provided") to those in Experiment 2 ("GPT-4, answer NOT provided"), as well as to the human-assigned labels. It was observed that supplying the GPT-4 with the answer to the question results in worse alignment with human-generated labels, compared to not providing the answer. While increasing the number of in-context examples from 60 to 90 questions resulted in improved alignment (50% to 60%), it still failed significantly below the performance when the question answers were not provided (83% alignment).

These results were surprising, as it was initially hypothesized that providing the immediate answer to the question would enhance the GPT-4's ability to infer the intent behind the question, leading to more accurate label assignments. However, the findings suggest that the inclusion of the answer may introduce additional complexity or ambiguity that the model struggles to interpret correctly. One possible reason for this could be that the model is overly influenced by the content of the answer. This can cause it to misinterpret the question's intent, particularly when the answer is long or includes several different ideas. In such cases, the model might struggle to focus on the core of the question, leading to classification errors. For instance, consider question 1: *"So how do you know how much do?"* Here is the same question (highlighted) followed by an utterance. An utterance is defined here exactly as described in Experiment 1.

*[`**So how do you know how much do?**`]*

*So there's gonna well or there are parameters and then by test testing the system, you'll determine what those parameters are operational parameters are right. like how much to grip and everything.*

GPT-4 labeled this question as DRQ in isolation which aligned with the human label. The reasoning from the GPT-4 response was that the question sought an explanation of
```

Table 4.5: Experiment 4c results

| **ID** | **Human Label** | **GPT-4, answer NOT provided** | | **GPT-4, answer provided** | |
|---|---|---|---|---|---|
| | | 60-shot | 90-shot | 60-shot | 90-shot |
| Q1 | DRQ | DRQ | DRQ | LLQ | LLQ |
| Q2 | GDQ | GDQ | GDQ | LLQ | GDQ |
| Q3 | DRQ | DRQ | DRQ | DRQ | DRQ |
| Q4 | GDQ | GDQ | GDQ | LLQ | LLQ |
| Q5 | LLQ | LLQ | LLQ | LLQ | LLQ |
| Q6 | LLQ | GDQ | GDQ | LLQ | GDQ |
| Q7 | DRQ | DRQ | DRQ | DRQ | DRQ |
| Q8 | DRQ | DRQ | DRQ | LLQ | LLQ |
| Q9 | GDQ | DRQ | DRQ | DRQ | DRQ |
| Q10 | LLQ | LLQ | LLQ | LLQ | LLQ |
| Q11 | LLQ | LLQ | LLQ | LLQ | LLQ |
| Q12 | GDQ | LLQ | LLQ | LLQ | LLQ |
| Q13 | LLQ | LLQ | LLQ | LLQ | LLQ |
| Q14 | GDQ | GDQ | GDQ | LLQ | GDQ |
| Q15 | GDQ | GDQ | GDQ | LLQ | GDQ |
| Q16 | DRQ | DRQ | DRQ | LLQ | LLQ |
| Q17 | LLQ | LLQ | LLQ | LLQ | LLQ |
| Q18 | LLQ | LLQ | LLQ | LLQ | LLQ |
| Q19 | DRQ | DRQ | DRQ | LLQ | LLQ |
| Q20 | LLQ | LLQ | LLQ | LLQ | LLQ |
| Q21 | DRQ | DRQ | DRQ | DRQ | DRQ |
| Q22 | DRQ | LLQ | LLQ | LLQ | LLQ |
| Q23 | GDQ | GDQ | GDQ | LLQ | DRQ |
| Q24 | DRQ | DRQ | DRQ | LLQ | LLQ |
| Q25 | DRQ | LLQ | LLQ | LLQ | LLQ |
| Q26 | LLQ | LLQ | LLQ | LLQ | LLQ |
| Q27 | GDQ | GDQ | GDQ | LLQ | GDQ |
| Q28 | GDQ | GDQ | GDQ | GDQ | GDQ |
| Q29 | GDQ | GDQ | GDQ | GDQ | GDQ |
| Q30 | LLQ | LLQ | LLQ | LLQ | LLQ |
| **Alignment** (%) | **n/a** | **83.33** | **83.33** | **50.00** | **60.00** |

the basis for a quantity, which implied a "rationale" or "interpretation". But with additional context, GPT-4 labeled it as LLQ in the current and the previous experiment

(in Experiment 4b). The reasoning of GPT-4 from the response was that the question sought a specific amount of information regarding operational parameters which was not true for this question. This shift is influenced by the surrounding discussion, which highlighted considerations of measurements and parameters in a design context, steering the interpretation toward seeking specific information rather than exploring deeper causal or rationale-based inquiries. This outcome underscores the challenge of context management in LLMs like GPT-4 (Li et al., 2024b). While human coders can use the answer to help guide their labeling decisions, LLMs may not have the same nuanced understanding (Bender et al., 2021). This makes it difficult for LLMs to accurately interpret how the answer relates to the question, potentially leading to misclassification. This suggests that the LLMs' decision-making process might benefit more from concise, focused context rather than extended contextual information (lengthy text), which can sometimes be more of a hindrance than a help (Hosseini et al., 2024).

## 4.2 Extended Experiments

Building on the findings from the preliminary experiments (Experiment 1, 2, and 3), which suggested that 90 in-context examples are sufficient for effective classification accuracy, the extended experiments (beginning with Experiment 5) utilized a refined version of this in-context examples set as opposed to randomly sampled in-context examples used in the preliminary experiments. Recent studies (Liu et al., 2022; Rubin et al., 2022; Chang and Jia, 2023) emphasized that selecting the right ICL examples is crucial for achieving optimal performance when dealing with LLMs. Also, as demonstrated by Zhao et al. (2021), LLMs like GPT-3 are biased toward predicting labels that are frequent in the prompt. Therefore, the in-context examples from dataset $d_1$ were modified by filtering out less effective examples and replacing them with more representative ones.

As part of this refinement, duplicate questions were removed from dataset $d_1$. These duplicates consisted of identical questions presented in different conversational contexts (either within the same session or across different sessions), which sometimes resulted in inconsistent labeling. Since the subsequent experiments would focus exclusively on isolated

questions without surrounding context (as seen in Experiment 4), removing duplicates ensured that each question in dataset $d_1$ was assigned a unique label. This step was taken to prevent potential confusion for the language model by avoiding scenarios where identical questions were inconsistently labeled within the same test set or across different test sets. Nevertheless, it is important to acknowledge that some questions may not fit precisely into one of the three primary categories (LLQ, DRQ, GDQ), which may pose challenges for consistent classification.

After this cleaning process, a total of 1,995 questions remained in dataset $d_1$, from which 90 questions were selected for ICL through uniform sampling at the category level (30 LLQs, 30 DRQs, and 30 GDQs) as in previous experiments, to ensure a balanced representation across categories. Care was also taken to ensure that the questions selected were concise, avoiding overly elaborate phrasing that could complicate the classification process as seen from the Experiment 4 that long context might overcomplicate for language model's understanding (An et al., 2024; Hosseini et al., 2024). Also, the training set from $d_1$ was created in a way that it consisted of all questions from subcategories as opposed to randomly sampled training examples used in the preliminary experiments. These adjustments to the in-context examples set were intended to optimize the model's performance in the subsequent experiments, where the scalability and generalizability of the approach were further tested.

### 4.2.1 Experiment 5

Experiment 5 aimed to further explore **RQ-1**, which investigates how the ICL influences the labeling accuracy of GPT-4 in automated question classification. Specifically, this experiment tested the scalability and robustness of GPT-4 by increasing the test set size from 30 to 60 questions. A larger test set would allow for a more comprehensive assessment of the model's accuracy and consistency across a broader range of scenarios, helping to reduce the impact of anomalies or outliers that might disproportionately affect the outcomes in a smaller test set. Following the exclusion of the ICL examples from dataset $d_1$, 1,905 questions (1266 LLQs, 139 DRQs, 500 GDQs) remained available for further analysis. These remaining questions were used to construct the test sets. To ensure a representative

distribution of the different question categories, the test sets were stratified as follows: 40 LLQs, 4 DRQs, and 16 GDQs. Although the distribution is not balanced across categories, this stratification reflects the proportions present in the remaining dataset. In all subsequent experiments, the test set size was consistently maintained at 60 questions, with the stratification described above.

From Experiment 4, it was observed that the length of the context plays a significant role in the model's classification performance. Although GPT-4 maintains a longer context window (128k input token limit), its ability to effectively utilize extended context remains limited (Li et al., 2024b; Levy et al., 2024). This observation raised questions about whether breaking down the test samples into smaller, more manageable subsets could mitigate any adverse effects of longer contexts. Specifically, the experiment examined whether processing smaller chunks of data (subsets of the test sample with a small sample size) at a time would alleviate any strain on the model compared to classifying the entire test sample at once. In other words, this experiment aimed to determine whether the GPT-4's ability to generate accurate responses is influenced by the size of the test sample being processed simultaneously. Therefore, each 60-question test sample was further subdivided into six distinct, mutually exclusive sets, each containing 10 questions with a stratified distribution across categories (see Appendix B.1 for the detailed distribution). By breaking down the 60-question test sample into these six sets, the goal was to explore the potential impact on the GPT-4's performance across these stratified subsets. The system prompt used for this experiment was A.1, the user prompt was the same as Experiment 1, and the experiment was run once for all 10 test samples.

The results presented in Table 4.6 demonstrated that the average label alignment for the six 10-question subsets was 69.50% ($SD = 5.63\%$), while the single 60-question set had an average label alignment of 73.17% ($SD = 5.35\%$). This overall alignment (%) for both conditions was worse than that of Experiment 1, where a single test set of 30 questions yielded an alignment of 83% while GPT-4 was provided with same-sized examples. One possible reason could be due to the increased size of the test set; GPT-4 likely encountered questions that were not well-represented in the in-context examples i.e., it was exposed to more variability and outlier questions that it may not have learned to handle effectively, a common issue dealing with machine learning models (Lones, 2024). Additionally, it is

important to note that, unlike Experiment 1 where the test data distribution was uniform across categories (10 questions for each category), in this case, the distribution was stratified. The average alignment across test sets for the condition "single 60-question" seemed higher than for the "six 10-question subsets" condition, with an average alignment of 73.17% compared to 69.50% for the subsets as illustrated in Figure 4.7. Now, to determine whether the label alignment (%) difference occurred between these two test conditions due to a random chance a non-parametric *Wilcoxon Signed-Rank Test* was conducted given the small sample size ($n = 10$). The test indicated that the difference in performance between the two test conditions was not statistically significant ($W = 12.0$, $p = 0.212 > 0.05$), i.e., the observed differences in label alignment could be attributed to random chance.

Additionally, from the results, it can be observed that in both conditions, GPT-4 performed worse in labeling GDQ compared to LLQ and DRQ (see Appendix B.2 for the aggregated label alignment results). To further investigate this observation, statistical tests were conducted to assess whether the differences in labeling accuracy across the three categories (LLQ, DRQ, GDQ) were statistically significant. In the case of the six 10-question subsets, the results indicated a significant difference in GPT-4's performance across the categories (*Friedman Test*, $\chi^2(2) = 13.40$, $p < 0.05$). The post-hoc analysis confirmed that GPT-4's alignment in labeling GDQ was significantly lower than in labeling both LLQ (*Nemenyi Test*, $p = 0.001 < 0.05$) and DRQ (*Nemenyi Test*, $p = 0.001 < 0.037$). Similarly, for the single 60-question set, the results also indicated a significant difference in performance across the categories (*Friedman Test*, $\chi^2(2) = 10.95$, $p = 0.004 < 0.05$). The post-hoc analysis confirmed that GPT-4's alignment in labeling GDQ was significantly lower than in labeling LLQ (*Nemenyi Test*, $p = 0.005 < 0.05$). On the other hand, the difference between DRQ and GDQ accuracy is not statistically significant, but it's close to the significance threshold ($p = 0.065 > 0.05$).

The responses from the GPT-4 were further manually analyzed to check the performance of the subcategories in both conditions and the results are presented in Table 4.7, where the subcategory alignment results are broken down by question types. In both conditions, GPT-4, on average, correctly labeled only less than half of the question utterances within their respective subcategories. A detailed breakdown of the subcategory label alignment for each condition is provided in Appendix B.4 and B.5. The analysis revealed that,

Table 4.6: Experiment 5 results - performance across categories.

Heading explanation: *"Sample (S)"* → Ten test samples; *"Six 10-Question Subsets"* → Aggregated results from six different subsample with size 10; *"Single 60-Question Set"* → Single sample with size 60. For each sample (S) a total number of correctly aligned labels were included, accuracy (%), and count of correctly labeled by category – LLQ, DRQ, GDQ. E.g., the cell *S1, Six 10-Question Subsets* denotes total correctly aligned labels with humans - 45, overall alignment with human labels 75%, and the label alignment distribution - 31/40 LLQ, 4/4 DRQ, 10/16 GDQ.

| **Sample (S)** | **Six 10-Question Subsets** | **Single 60-Question Set** |
|---|---|---|
| S1 | 45<br>(75.00%)<br>[36, 2, 7] | 41<br>(68.33%)<br>[34, 2, 5] |
| S2 | 40<br>(66.67%)<br>[31, 3, 6] | 48<br>(80.00%)<br>[37, 4, 7] |
| S3 | 44<br>(73.33%)<br>[33, 4, 7] | 41<br>(68.33%)<br>[30, 4, 7] |
| S4 | 39<br>(65.00%)<br>[28, 3, 8] | 41<br>(68.33%)<br>[31, 4, 6] |
| S5 | 40<br>(66.67%)<br>[33, 2, 5] | 43<br>(71.67%)<br>[33, 2, 8] |
| S6 | 49<br>(81.67%)<br>[37, 4, 8] | 49<br>(81.67%)<br>[35, 4, 10] |
| S7 | 43<br>(71.67%)<br>[34, 2, 7] | 45<br>(75.00%)<br>[36, 2, 7] |
| S8 | 37<br>(61.67%)<br>[31, 2, 4] | 48<br>(80.00%)<br>[33, 3, 12] |
| S9 | 41<br>(68.33%)<br>[33, 1, 7] | 40<br>(66.67%)<br>[33, 2, 5] |
| S10 | 39<br>(65.00%)<br>[30, 2, 7] | 43<br>(71.67%)<br>[32, 2, 9] |
| **Average Alignment (%)** | **69.50** | **73.17** |

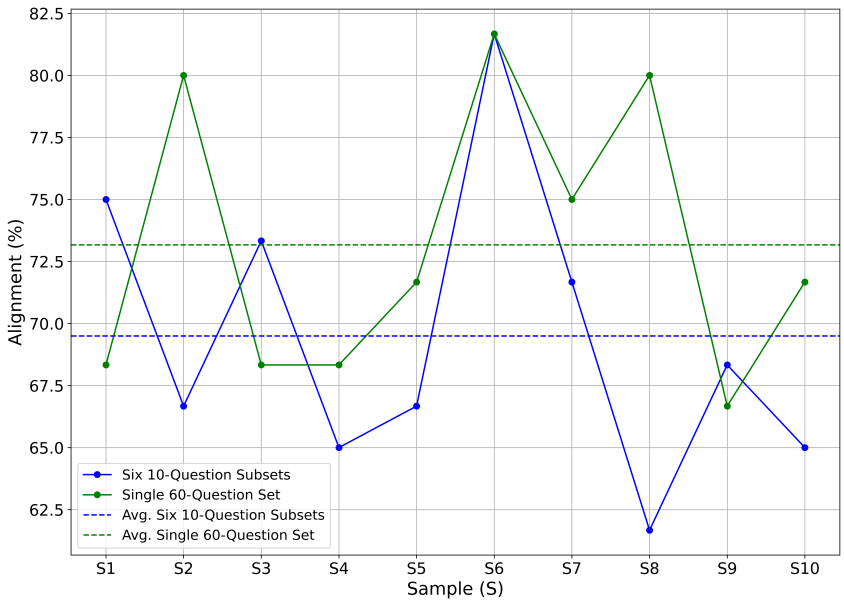



Figure 4.7: Alignment (%) comparison across different samples for the two test conditions.

in both conditions, out of 16 subcategories present in the ten sets, only 5 categories-LLQ (Quantification, Verification), DRQ (Definition, Comparison, Instrumental/Procedural), GDQ (Proposal/ Negotiation)—achieved more than 50% alignment with human labels. Notably, GPT-4 consistently failed to correctly label any questions in the 'Enablement' subcategory across both conditions. Similar to previous findings, the performance difference in both conditions was found to be non-significant (*Wilcoxon Signed-Rank Test*, $W = 12.0$, $p = 0.213 > 0.05$).

These statistical analyses confirmed that variations in ICL conditions did not significantly impact GPT-4's overall performance. However, across both scenarios, GPT-4 consistently struggled more with labeling GDQs than with LLQs and DRQs, highlighting a persistent area of weakness. One possible explanation for this pattern could be the complexity and open-ended nature of GDQs, which involve divergent thinking and often require synthesizing multiple design considerations and generating novel solutions (Cardoso et al.,

Table 4.7: Experiment 5 results - performance across subcategories.

Heading explanation: *"Sample (S)"* → Ten test samples; *"Six 10-Question Subsets"* → Result aggregated from six different subsamples with size 10; *"Single 60-Question Set"* → Result from a single sample with size 60. Each sample (S) includes the total number of correctly aligned subcategory labels, overall subcategory alignment (%), and count of correctly aligned subcategories under the three main categories (LLQ, DRQ, GDQ). E.g., the cell *S1, Six 10-Question Subsets* denotes total correctly aligned subcategory labels with humans - 32/60, overall alignment with human labels - 53.33%, and the subcategory label alignment distribution - 25/40 LLQ, 1/4 DRQ, 6/16 GDQ.

| **Sample (S)** | **Six 10-Question Subsets** | **Single 60-Question Set** |
|---|---|---|
| S1 | 32<br>(53.33%)<br>[25, 1, 6] | 25<br>(41.67%)<br>[22, 1, 2] |
| S2 | 27<br>(45.00%)<br>[20, 3, 4] | 31<br>(51.67%)<br>[23, 2, 6] |
| S3 | 26<br>(43.33%)<br>[20, 3, 3] | 26<br>(43.33%)<br>[20, 3, 3] |
| S4 | 21<br>(35.00%)<br>[11, 3, 7] | 21<br>(35.00%)<br>[13, 3, 5] |
| S5 | 23<br>(38.33%)<br>[17, 2, 4] | 23<br>(38.33%)<br>[16, 2, 5] |
| S6 | 34<br>(56.67%)<br>[27, 3, 4] | 39<br>(65.00%)<br>[29, 3, 7] |
| S7 | 31<br>(51.67%)<br>[26, 0, 5] | 33<br>(55.00%)<br>[26, 0, 7] |
| S8 | 26<br>(43.33%)<br>[21, 2, 3] | 38<br>(63.33%)<br>[25, 3, 10] |
| S9 | 25<br>(41.67%)<br>[21, 0, 4] | 21<br>(35.00%)<br>[18, 0, 3] |
| S10 | 30<br>(50.00%)<br>[25, 1, 4] | 31<br>(51.67%)<br>[24, 1, 6] |
| **Average Alignment (%)** | **45.83** | **48.00** |

2016). Unlike LLQs, which tend to be fact-based, and DRQs, which are more reasoning-oriented—both of which facilitate convergent thinking processes—GDQs may demand a more nuanced understanding of design intent and creative exploration, areas where LLMs like GPT-4's training data or model structure may have limitations. This observation suggested that, in this experiment, while GPT-4 performed well on questions requiring factual or logical alignment, it appeared to face challenges with contexts involving higher levels of abstraction and creativity, such as GDQ labeling. GPT-4's consistent difficulty with GDQs across different test conditions highlighted an area where improvements in handling divergent thought processes could enhance its performance. Further exploration is needed to confirm and better understand this limitation.

### 4.2.2 Experiment 6

Experiment 6 continued to explore **RQ-1**, which investigates how ICL influences the accuracy of GPT-4 in automated question classification. While previous experiments (Experiments 1 and 2) examined the impact of the number of in-context examples, Experiment 6 shifted the focus to how different prompt conditions—ranging from minimal information (zero-shot) to comprehensive prompt examples—affect the model's performance. By analyzing GPT-4's responses across these varying conditions, this experiment sought to determine the extent to which the structure and content of prompts could enhance or diminish the labeling accuracy of question classification, thereby providing a more nuanced understanding of the role of ICL in improving the model's effectiveness. Therefore, GPT-4's performance was tested across different prompt conditions using the first ten test samples from Experiment 5, with the same 90 in-context examples (labeled questions) used in that experiment. The different prompt conditions were:

A. Baseline: No content in the system message.

B. Definitions Only: The system message contained only the definitions of question categories (see Appendix A.2).

C. Definitions + Taxonomy Examples: The system message included definitions along with corresponding high-quality examples (see Appendix A.1).

D. Definitions + Taxonomy Examples + In-Context Examples: The system message included definitions and corresponding examples as condition C, while the user message contained in-context examples (90 labeled questions from Experiment 5).

For conditions A, B, and C the user prompt contained no labeled example (see Appendix C.1). Condition D used the same user prompt as in Experiment 1. The experiments were run once in all conditions for all 10 test samples.

The performance of the four prompt conditions (A, B, C, and D) was evaluated based on their label alignment (%) across ten samples (S1–S10), as presented in Table 4.8. The results indicated a significant difference in label alignment among the conditions (*Friedman test*, $\chi^2(3, N = 60) = 21.03$, $p < 0.05$). Furthermore, the post hoc *Nemenyi test* revealed that condition A differed significantly from both conditions C ($p = 0.001 < 0.05$), and D ($p = 0.001 < 0.05$), while no significant pairwise differences were observed between Condition B and any other condition, nor between Conditions C and D (see Appendix C.2 for details).

The significant differences observed between Condition A and the other conditions indicated that the lack of additional context or examples in Condition A contributed to lower label alignment. However, the absence of significant differences between the other conditions implied that the inclusion of definitions and examples (Conditions B, C, and D) might have similarly enhanced the model's performance, without any one condition being markedly superior.

This experiment also revealed insights into the impact of the ICL examples. In Experiment 1, the label alignment exceeded 80% when ICL examples were provided. However, in Experiment 5 which was more comprehensive, the highest alignment was on average 74%, with only 3 out of 10 samples reaching around 80% alignment. The test data used in Experiment 1 had a uniform distribution across categories in contrast to the stratified distribution of test data in Experiment 5. Additionally, the test set in Experiment 1 was half the size. These differences, combined with the small and non-significant difference between Condition C (without in-context examples) and Condition D (with in-context examples), implied that the impact of the in-context examples might be less conclusive than initially anticipated.

Table 4.8: Experiment 6 results.

A, B, C, and D present different prompt conditions for the experiment. For each sample (S) a total number of correctly aligned labels was included, label alignment (%), and the count of aligned labels by category – LLQ, DRQ, GDQ. E.g., the cell S1, A denotes total correctly aligned labels with humans - 35, overall alignment - 58.33%, and the accuracy distribution - 28/40 LLQ, 2/4 DRQ, 5/16 GDQ.

| **Sample (S)** | **A** | **B** | **C** | **D** |
|---|---|---|---|---|
| S1 | 35<br>(58.33%)<br>[28, 2, 5] | 41<br>(68.33%)<br>[34, 2, 5] | 41<br>(68.33%)<br>[34, 2, 5] | 41<br>(68.33%)<br>[34, 2, 5] |
| S2 | 32<br>(53.33%)<br>[21, 4, 7] | 40<br>(66.67%)<br>[31, 3, 6] | 38<br>(63.33%)<br>[29, 3, 6] | 48<br>(80.00%)<br>[37, 4, 7] |
| S3 | 33<br>(55.00%)<br>[28, 3, 2] | 42<br>(70.00%)<br>[33, 3, 6] | 43<br>(71.67%)<br>[33, 3, 7] | 41<br>(68.33%)<br>[30, 4, 7] |
| S4 | 30<br>(50.00%)<br>[22, 2, 6] | 40<br>(66.67%)<br>[32, 3, 5] | 40<br>(66.67%)<br>[35, 3, 2] | 41<br>(68.33%)<br>[31, 4, 6] |
| S5 | 31<br>(51.67%)<br>[26, 0, 5] | 40<br>(66.67%)<br>[33, 2, 5] | 42<br>(70.00%)<br>[34, 2, 6] | 43<br>(71.67%)<br>[33, 2, 8] |
| S6 | 40<br>(66.67%)<br>[32, 1, 7] | 50<br>(83.33%)<br>[38, 3, 9] | 51<br>(85.00%)<br>[37, 4, 10] | 49<br>(81.67%)<br>[35, 4, 10] |
| S7 | 35<br>(58.33%)<br>[32, 1, 2] | 40<br>(66.67%)<br>[36, 1, 3] | 46<br>(76.67%)<br>[37, 2, 7] | 45<br>(75.00%)<br>[36, 2, 7] |
| S8 | 34<br>(56.67%)<br>[26, 1, 7] | 42<br>(70.00%)<br>[33, 3, 6] | 43<br>(71.67%)<br>[34, 3, 6] | 48<br>(80.00%)<br>[33, 3, 12] |
| S9 | 38<br>(63.33%)<br>[32, 2, 4] | 38<br>(63.33%)<br>[33, 2, 3] | 42<br>(70.00%)<br>[33, 2, 7] | 40<br>(66.67%)<br>[33, 2, 5] |
| S10 | 35<br>(58.33%)<br>[27, 3, 5] | 43<br>(71.67%)<br>[32, 2, 9] | 43<br>(71.67%)<br>[33, 2, 8] | 43<br>(71.67%)<br>[32, 2, 9] |
| **Average Alignment (%)** | **57.17** | **69.33** | **71.50** | **73.17** |

It is also important to note that in condition A, where no system message was provided and the model relied solely on the user prompt without any in-context examples, GPT-4 achieved an alignment exceeding 50% across all samples. This was a significant observation given that the task involves labeling questions into one of three categories (LLQ, DRQ, GDQ), where random chance would yield an alignment of 33.33%. The fact that GPT-4's performance under these minimal conditions surpassed the baseline expectation of random chance suggests that the model might be leveraging implicit knowledge, potentially due to the mention of the Eris taxonomy in the prompt (user message). Even without explicit definitions or examples, this implicit reference may contribute to the model's ability to classify questions with a level of accuracy better than random guessing.

### 4.2.3 Experiment 7

LLMs like GPT-4 can recognize patterns even with a limited number of examples (Brown et al., 2020), which enables the model to adapt quickly to new tasks with minimal additional data (Wei et al., 2022). This is due to their extensive training on diverse, large-scale datasets, which equip them to handle inputs that differ from those seen in training. This capability allows LLMs like GPT-4 to perform a wide range of tasks, such as text summarization, translation, and question-answering, without extensive task-specific training (Achiam et al., 2023; Naveed et al., 2023). Moreover, LLMs are also effective at generalizing to out-of-distribution (OOD) data. Cheng et al. (2023) found that prompting GPT-3 can be more robust than supervised fine-tuning of smaller-scale language models like *BERT* and *RoBERTa* by conducting OOD tests on various datasets. Their experiment indicated that GPT-3 prompts can directly generalize to OOD test sets, where the test examples differ in distribution from those seen in the prompt demonstrations. A recent study by Yuan et al. (2024) introduced *BOSS*, a benchmark for OOD robustness evaluation, where the authors found that LLMs like GPT-3.5 Turbo exhibited better performance on most ODD datasets reinforcing the view that large pre-trained models possess strong generalization capabilities. This motivated the exploration of whether GPT-4 would exhibit similar performance when provided with ICL examples from one dataset $d_1$ and tested on a different dataset $d_2$ and vice versa. Both datasets share a common foundation, as

the questions were labeled using Eris' taxonomy (Eris, 2004). Therefore, it was hoped that GPT-4 would be able to maintain consistent classification performance across both datasets. Thus, the objective of this experiment, aligned with **RQ-4**, was to assess the generalizability of GPT-4's performance across different datasets. The motivation behind this experiment was to determine whether the prompts and in-context examples developed using one dataset would be effective when applied to a different dataset, thereby testing the robustness and adaptability of GPT-4 in varying contexts. The experiment involved two cases:

a. Using in-context examples from $d_1$ and test samples from $d_2$, and

b. Using in-context examples from $d_2$ and test samples from $d_1$.

In both cases, the system prompt was A.1, and the user prompt was identical to that used in Experiment 1. Also for both cases, the in-context examples consisted of 90 questions and each test sample included 60 questions (except for the last sample of $d_2$), consistent with Experiments 5 and 6. The experiments were run once for all the test samples.

In the first case, the same 90 in-context examples from dataset $d_1$ were used, which were previously utilized in Experiments 5 and 6. On the other hand, after 90 questions were excluded from dataset $d_2$ to be used as in-context examples, the remaining 175 questions were divided into three stratified samples and used as test sets. Due to the size constraints of $d_2$, each test sample contained 60 questions, except for the last sample, which had 55 questions. The category distribution among the test samples varied as well: the first two samples consisted of 45 LLQs, 10 DRQs, and 5 GDQs, while the last sample consisted of 41 LLQs, 10 DRQs, and 4 GDQs. In contrast, the $d_1$ test samples each had a distribution of 40 LLQs, 4 DRQs, and 16 GDQs. It should be noted that the decision to create three test samples in both cases was driven by the size constraints of dataset $d_2$.

The results from the first case are presented in Table 4.9, which demonstrates that when GPT-4 was provided in-context labeled examples from $d_1$, the average alignment with human-generated labels was 79.24% ($SD$ = 5.64%). In comparison, ICL on $d_2$ labeled examples resulted in an average alignment of 80.30% ($SD$ = 7.08%). The performance

was found to be very similar in both conditions (*Wilcoxon Signed-Rank Test*, $W = 1.00$, $p = 0.655 > 0.05$).

Table 4.9: Experiment 7a results - performance comparison on $d_2$ based on in-context examples.

For each test sample (S), the table provides - the number of correct predictions, the label alignment (%), and the distribution of predictions across three categories (LLQ, DRQ, and GDQ). E.g., the cell *S1, ICL on $d_1$ Examples* denotes the first test sample, provided with labeled examples from $d_1$ (size: 90) resulting in 50 correct predictions out of 60, resulting in an alignment of 83.33%, with a distribution of 40/45 LLQ, 9/10 DRQ, and 1/5 GDQ. The star(*) indicates the size of S3 is 55 with a distribution of 30/41 LLQ, 8/10 DRQ, and 2/4 GDQ .

| **Test Sample ($d_2$, Size: 60)** | **ICL on $d_1$ Examples** | **ICL on $d_2$ Examples** |
|---|---|---|
| S1 | 50<br>(83.33%)<br>[40, 9, 1] | 50<br>(83.33%)<br>[39, 9, 2] |
| S2 | 49<br>(81.67%)<br>[37, 10, 2] | 52<br>(86.67%)<br>[40, 10, 2] |
| S3* | 40<br>(72.72%)<br>[30, 8, 2] | 39<br>(70.91%)<br>[29, 8, 2] |
| **Average Alignment (%)** | **79.24** | **80.30** |

In the second case, a randomly generated stratified set of 90 questions from dataset $d_2$ was prepared to use as a prompt dataset. The model's performance was then tested on three sample sets derived from dataset $d_1$ (Same as Experiment 6), where each stratified sample (40 LLQ, 4 DRQ, and 16 GDQ) consisted of 60 questions. The results from the second case are presented in Table 4.10, which demonstrates that when GPT-4 was provided with labeled examples from $d_1$, the average alignment with human-generated labels was 72.22% ($SD$ =6.70%). In comparison, ICL on $d_2$ examples resulted in an average alignment of 71.66% ($SD$ =5.96%). Similar to the first case, the performance between these two conditions was also found to be very similar (*Wilcoxon Signed-Rank Test*, $W = 2.50$,

$p = 1.00 > 0.05$).

Table 4.10: Experiment 7b results - performance comparison on $d_1$ based on in-context examples.

For each test sample (S), the table provides - the number of correct predictions, the label alignment (%), and the distribution of predictions across three categories (LLQ, DRQ, GDQ). For instance, the cell S1 under "ICL on $d_1$ Examples" indicates that the first test sample, provided with labeled examples from $d_1$ resulted in 41 correct predictions out of 60, resulting in an accuracy of 68.33%, with a distribution of 34/40 LLQ, 2/4 DRQ, and 5/16 GDQ.

| **Test Sample ($d_1$, Size: 60)** | **ICL on $d_1$ Examples** | **ICL on $d_2$ Examples** |
|---|---|---|
| S1 | 41<br>(68.33%)<br>[34, 2, 5] | 40<br>(66.67%)<br>[35, 2, 3] |
| S2 | 48<br>(80.00%)<br>[37, 4, 7] | 47<br>(78.33%)<br>[37, 4, 6] |
| S3 | 41<br>(68.33%)<br>[30, 4, 7] | 42<br>(70.00%)<br>[33, 3, 6] |
| **Average Alignment (%)** | **72.22** | **71.66** |

Overall, the results indicated that GPT-4 was capable of maintaining consistent classification performance across datasets $d_1$ and $d_2$. This implied that in this scenario, GPT-4's generalization ability was well enough to handle different datasets without experiencing a significant decline in labeling accuracy. Also, from Experiment 6, it can be said that it would perform almost the same without in-context examples.

Based on the results from both cases, it was also observed that GPT-4 consistently performed better when tested on dataset $d_2$, regardless of whether it was provided labeled examples from dataset $d_1$ or $d_2$. However, this difference was found to be non-significant (*Mann-Whitney U test*, $U = 2.00$, $p = 0.20 > 0.05$). Although the difference is non-significant, the result implied that the classification performance improved when tested on

a better-quality dataset, which in this scenario was $d_2$. This is further discussed in the discussion (section 5.1).

### 4.2.4 Experiment 8

Following the evaluation of GPT-4's generalizability on a different dataset ($d_2$) in Experiment 7, Experiment 8 extended this inquiry by comparing GPT-4's performance with that of another LLM, further addressing **RQ-4**, which explores how generalizable the prompts are across different datasets and when applied to other LLMs. Claude 3.5 Sonnet (Anthropic, 2024a) was used for this experiment as it is one of the leading competitors of GPT-4. Since Claude 3.5 Sonnet showed a competitive performance against GPT-4 across a range of benchmarks, including GPQA, MMLU, and HumanEval (Anthropic, 2024b; Huang et al., 2024), it was expected to observe better if not similar performance from Claude 3.5 Sonnet compared to GPT-4.

Thus, the goal of this experiment was to assess the robustness of the classification prompts and in-context examples when applied to a different model, further testing the adaptability of the methods. This experiment was conducted using both models in two distinct cases:

a. Using in-context examples from $d_1$ and test samples from $d_1$, and

b. Using in-context examples from $d_1$ and test samples from $d_2$.

In both cases, the experiment was conducted using both models: GPT-4 and Claude 3.5 Sonnet. The system prompt was A.1, and the user prompt was identical to that used in Experiment 1. Also for both cases, the in-context examples consisted of 90 questions and each test sample included 60 questions (except for $d_2$), consistent with Experiments 5, 6 and 7. The experiments were run once for all the test samples. Again, the decision to create three test samples in both cases was influenced by the size constraint of dataset $d_2$. The official anthropic API (Anthropic, 2024d) was used to run the Claude 3.5 Sonnet model. One thing to note here is that, unlike OpenAI API (OpenAI, 2023a), anthropic

text completion API did not provide parameters, such as - *seed* which indicated that the Claude 3.5 Sonnet's response might not be consistent like that of GPT-4 as *seed* produces (mostly) deterministic outputs (OpenAI, 2023).

In the first case, the same 90 ICL examples from dataset $d_1$, previously utilized in Experiments 5, 6, and 7, were used. The three test samples from $d_1$ used in this experiment were the same as in Experiment 7 which had a distribution of 40 LLQs, 4 DRQs, and 16 GDQs for each totaling 60 questions.

The results from the first case are presented in Table 4.11, which showed that GPT-4 achieved an average label alignment of 72.22% ($SD$ = 5.95%), while Claude reached an average label alignment of 73.33% ($SD$ = 4.36%) on $d_1$ test samples. The performance was found to be very similar between GPT-4 and Claude (*Wilcoxon Signed-Rank Test*, $W = 3.00$, $p = 1.00 > 0.05$).

Table 4.11: Experiment 8a results - performance comparison between GPT-4 and Claude 3.5 on $d_1$.

| **Test Sample ($d_1$, Size: 60)** | **GPT-4** | **Claude** |
|---|---|---|
| S1 | 41<br>(68.33%)<br>[34, 2, 5] | 46<br>(76.67%)<br>[32, 2, 12] |
| S2 | 48<br>(80.00%)<br>[37, 4, 7] | 41<br>(68.33%)<br>[27, 4, 10] |
| S3 | 41<br>(68.33%)<br>[30, 4, 7] | 45<br>(75.00%)<br>[33, 4, 8] |
| **Average Alignment (%)** | **72.22%** | **73.33%** |

In the second case, the same 90 in-context examples from dataset $d_1$ were used to test against three test samples from $d_2$, which were previously used in Experiment 7 where

the first two samples consisted of 45 LLQs, 10 DRQs, and 5 GDQs, and the last sample consisted of 41 LLQs, 10 DRQs, and 4 GDQs.

The results from the second case are presented in Table 4.12, which showed that GPT-4 achieved an average label alignment of 79.24% ($SD = 5.80\%$), while Claude 3.5 reached an average label alignment of 78.64% ($SD = 7.47\%$). Similar to the first case, the performance between GPT-4 and Claude was also found to be very similar (*Wilcoxon Signed-Rank Test*, $W = 2.50$, $p = 1.00 > 0.05$).

Table 4.12: Experiment 8b results - performance comparison between GPT-4 and Claude 3.5 on $d_2$.
The star (*) indicates the size of S3 is 55 with a distribution-41 LLQ, 10 DRQ, and 4 GDQ.

| **Test Sample ($d_2$, Size: 60)** | **GPT-4** | **Claude 3.5** |
|---|---|---|
| S1 | 50<br>(83.33%)<br>[40, 9, 1] | 53<br>(88.33%)<br>[43, 9, 1] |
| S2 | 49<br>(81.67%)<br>[37, 10, 2] | 46<br>(76.67%)<br>[31, 10, 5] |
| S3* | 40<br>(72.72%)<br>[30, 8, 2] | 39<br>(70.91%)<br>[30, 6, 3] |
| **Average Alignment (%)** | **79.24** | **78.64** |

These findings from both experiments suggested that GPT-4 and Claude 3.5 Sonnet exhibited comparable robustness and adaptability in applying classification prompts across different datasets. Additionally, it is important to note that Claude 3.5 Sonnet not only matched GPT-4 in performance but also offered a cost advantage, being significantly cheaper (three times cheaper for prompts and half the cost for responses) than GPT-4 (Anthropic, 2024b). This cost-effectiveness, combined with its robust performance, positions Claude 3.5 Sonnet as a competitive alternative for large-scale applications where

budget constraints are a consideration. Furthermore, similar to Experiment 7, it was also observed that regardless of whether it was provided labeled examples from dataset $d_1$ or $d_2$, Claude's performance on $d_2$ seemed to be better than on $d_1$, but found to be non-significant (*Mann-Whitney U test*, $U = 2.00$, $p = 0.40 > 0.05$). Again, although the difference was non-significant, the result implied that the performance of LLMs improved on better-quality datasets.

# Chapter 5

# Discussion

As LLMs become more sophisticated and ubiquitous, researchers have rightly begun to ponder how ChatGPT and other technologies will shape qualitative research (van Manen, 2023). This research was motivated by the broader aim of exploring how LLMs could be used in analyzing verbal protocols of design. Specifically, in this paper, the primary research objective was to determine the extent to which the most recent variant of ChatGPT (GPT-4) could accurately classify question utterances according to Eris (2004) taxonomy.

The study took advantage of two pre-existing datasets where one comprised a large corpus of over 2000 human-classified question utterances and the other one comprised 265 question utterances. Dataset $d_1$ included the complete transcripts of the dialogues in which the questions were uttered, while dataset $d_2$ only consisted of the questions and the corresponding classification categories. The study was comprised of a series of careful experiments that sought to understand the nuances of using GPT-4 for performing this classification. The following subsections highlight the study findings, including limitations, implications, and opportunities for future research.

## 5.1 Classification accuracy and sensitivity

Experiment 1 sought to establish the baseline performance of GPT-4 in this classification task. In all cases, GPT-4 was provided with a system message that described the classification scheme (the definition of the LLQ, DRQ, and GDQ categories and the sub-categories within each) and provided one example for each question subcategory. The classification task consisted of assigning each of the randomly selected 30 questions to one of the three categories (but not to their subcategories). Without any further ICL, GPT-4 achieved an alignment of 60% with the human-assigned categories. However, when given 90 additional in-context examples of human-labeled questions, GPT-4's performance reached an alignment of 83%. Although this improvement initially emphasized the importance of a set of well-curated in-context examples in enhancing GPT-4's classification accuracy, Experiments 5 and 6 revealed that the impact of the in-context examples might be less conclusive than initially anticipated. Other factors, such as - the quality or the size of the test samples might have had more influence on performance outcome, thus addressing **RQ-1** which explores *how the ICL influences the labeling accuracy of GPT-4 in automated question classification.* However, further research would be needed to confirm this hypothesis. Importantly, where there was misalignment, there was ambiguity in the correct classification for the question. For example, the two human coders had disagreed on the correct category in the initial coding, or in some cases, the AI-assigned labels were believed to be correct when considering the question in isolation, leading to disagreements with the human-assigned categories. The agreement level between the two coders was 85.6% at the broader level when considering the three main categories of questions for dataset $d_1$ (Hurst et al., 2023). Therefore, it can be argued that for dataset $d_1$, GPT-4 did perform fairly well ($> 70\%$ average alignment in Experiment 5) at this classification task. Furthermore, this hints that future research could explore refining category definitions or developing a more nuanced classification system to better account for questions that may fall into gray areas between established categories.

Experiment 2 sought to test the sensitivity of the classification to the number of in-context examples, exploring **RQ-1**. Results from this experiment suggest that GPT-4 is capable of performing this task and maintaining adequate alignment with human labels

with a fairly limited number of in-context examples (a mere 60 pre-labeled questions). Owing to the probabilistic nature of how GPT-4 performs, Experiment 3 sought to determine the sensitivity of the classification across a different dimension – the number of "runs" of a classification task, directly addressing **RQ-2**, which explores *how consistently GPT-4 classifies questions in design protocols across different instances.* Indeed, the results demonstrated the probabilistic nature of the labeling, with many questions being labeled differently on different runs. Given this variability, a potential avenue for future work could involve leveraging the log probabilities of the output tokens of LLMs (e.g., "logprobs" property provided by OpenAI (2023) API) GPT-4 during classification tasks. By analyzing the log probabilities researchers could gain deeper insights into the model's confidence levels for different labels and identify cases where the model is uncertain or conflicted between multiple classifications. This could be particularly useful for refining classification strategies, such as by incorporating thresholding techniques where labels are only assigned if the log probabilities indicate a high enough confidence level. Additionally, log probabilities could be used to inform a hybrid human-AI approach, where cases with low-confidence classifications are flagged for human review. This approach could potentially help achieve better alignment with human judgment, especially in ambiguous or complex scenarios. Furthermore, the results from Experiment 3 indicated that GPT-4 classifications could be more likely to remain consistent across runs when a larger set of in-context examples was used.

Further experiments were conducted to investigate some "sensitivity" aspects, yielding insightful results. First, how results vary with larger test data sets was examined. Experiment 5 sought to test this, further investigating **RQ-1**, which focuses on the *influence of ICL on GPT-4's labeling accuracy in question classification.* The test sample size was expanded from 30 to 60 questions, and the experiment was conducted under two conditions: one where each test sample was divided into six mutually exclusive subsets of 10 questions, and the other using the full set of 60 questions for each sample. This design enabled an evaluation of the model's performance under different conditions of sample segmentation and aggregation. By comparing the results across these conditions, the experiment aimed to understand how the size of test samples influenced the model's accuracy and reliability. The findings revealed that dividing the test datasets into smaller subsets did not impact

GPT-4's overall performance. However, GPT-4 consistently struggled more with labeling GDQs compared to LLQs and DRQs. GDQ's open-ended and creative nature, requiring divergent thinking, could be a possible reason, in contrast to the more structured and convergent thinking processes involved in addressing LLQs and DRQs. Future research could focus on a more detailed examination of the specific characteristics of GDQ that make them challenging. This could involve qualitative analysis of model errors, as well as exploring whether particular linguistic, semantic, or contextual features of GDQ contribute to these difficulties. Additionally, a thorough performance analysis is required to determine whether LLMs like GPT-4 exhibit biases toward certain categories, which could further explain its persistent difficulties with labeling GDQ compared to LLQ and DRQ. Furthermore, a deeper analysis of GPT-4's responses from Experiment 5 revealed that it correctly labeled, on average, less than half of the question subcategories in both conditions. Only five out of sixteen subcategories (from the ten test samples) achieved over 50% alignment with human labels, and GPT-4 consistently failed to label any questions in the "Enablement" subcategory in both conditions. These findings highlight the limitations of GPT-4's current capabilities in handling nuanced subcategory distinctions and suggest opportunities for future research.

Second, in Experiment 6, GPT-4's performance was tested under different prompting conditions to explore how variations in prompt detail affect the model's classification accuracy, further addressing **RQ-1**, which examines *how ICL influences the accuracy of GPT-4 in automated question classification*. The prompts ranged from providing minimal information i.e., zero-shot setting (no system message, only user message, and no in-context examples) to more comprehensive conditions that included definitions of question categories, definitions with corresponding high-quality taxonomy examples, or in-context examples in the user prompt (90 labeled question). The results indicated that providing definitions, definitions with corresponding examples, or additional in-context examples generally enhanced the model's performance compared to providing no information i.e., zero-shot setting (no system message, no in-context examples). However, the difference was not significant among the conditions that included definitions, definitions with corresponding high-quality examples, or additional in-context examples, suggesting that any form of additional context improved performance, but not one approach was superior.

Interestingly, even with minimal input (no system message, only user message, and no additional labeled examples), GPT-4's accuracy exceeded random chance, likely due to implicit knowledge embedded in the limited context provided, possibly because of the mention of the Eris taxonomy in the prompt. This suggests that GPT-4 may rely on its pre-existing knowledge base, allowing it to classify questions more accurately than random guessing, even with minimal guidance.

Third, the sensitivity of results to the origin of the in-context examples was tested by using different datasets for ICL in Experiment 7. This experiment addressed **RQ-4**, which examines *how generalizable the prompts used for question classification are across different datasets or when applied to other LLMs*. It was observed that GPT-4 was indeed capable of maintaining consistent classification accuracy across datasets $d_1$ and $d_2$. Its performance remained stable and comparable regardless of whether the in-context examples came from $d_1$ or $d_2$. This indicated that, in this scenario, LLMs like GPT-4 do not require dataset-specific fine-tuning, further reinforcing the conclusion from Experiment 6 that the impact of the additional in-context examples might be less significant.

Fourth, GPT-4's performance was compared with another competitive proprietary LLM, Claude 3.5 Sonnet (Anthropic, 2024b), in Experiment 8. This experiment aimed to assess the robustness of the classification prompts and labeled examples when applied to a different model. It was observed that both GPT-4 and Claude 3.5 demonstrated similar performance and adaptability in applying classification prompts across both datasets. Furthermore, Claude 3.5 Sonnet not only matched GPT-4 in performance but also offered a significant cost advantage—being three times cheaper for input processing (prompts) and half the cost for output generation (responses)—making it a competitive alternative for budget-conscious large-scale applications (Anthropic, 2024b). This suggests that the LLM, Claude 3.5 Sonnet performs comparably to its more costly counterpart, GPT-4. This raises an intriguing opportunity for future research to explore even less sophisticated models to determine if they can achieve similar performance levels at a fraction of the cost. Also, it should be noted that both GPT-4 and Claude 3.5 are proprietary models. Therefore, future studies could explore open-source LLMs to assess their performance in comparison to proprietary models like GPT-4 and Claude 3.5. Such research could provide valuable insights into the potential advantages or limitations of open-source alternatives, particularly

in terms of accessibility, transparency, and adaptability across various tasks in qualitative research.

Additionally, another interesting finding from Experiments 7 and 8 was that regardless of whether the model was provided examples from dataset $d_1$ or $d_2$, performance in $d_2$ seemed to be better than performance in $d_1$. Although the differences were not found to be statistically significant in both cases, this implied that $d_2$ might be a better quality dataset (had fewer ambiguities or noise) or a more consistent dataset compared to $d_1$. This better quality in $d_2$ may be attributed to its authentic peer-review setting, where students engaged naturally in reflective questioning, providing a richer, more genuine representation of question-asking behaviors. Hence, improvement in classification accuracy when the dataset was of better quality. However, further research would be needed to confirm this hypothesis.

While not within the scope of this study, two further "sensitivity" aspects have been identified that could be further investigated in future studies. First, how the results may have varied across different runs (in Experiment 3) if the order of labeled questions in the in-context examples had been varied was pondered. This consideration is based on findings from the literature, which suggest that the order of in-context examples can influence model performance (Lu et al., 2022; Zhang et al., 2022). GPT-4 appears highly sensitive to changes in the user prompt, and thus the impact of this factor remains to be confirmed.

Second, while Experiment 5 included an analysis of subcategories, the classification task primarily focused on the LLQ/DRQ/GDQ level and was not explicitly designed to test performance at the sub-category level. However, distinguishing between subcategories presents additional challenges, as increasing the number of classes introduces greater complexity, particularly when subtle differences between classes can lead to poor performance (Edwards and Camacho-Collados, 2024). This presents a challenge for performing question classification across all subcategories, given the 21 distinct subcategories in Eris' Taxonomy (Eris, 2004). Potential solutions to address this issue include employing data augmentation and hierarchical text classification approaches. For instance, Sahu et al. (2023) introduced *PromptMix*, a data augmentation method that generates challenging borderline examples that challenge the model to handle subtle distinctions between classes and relabels them using a prompting-based LLM classifier to reduce labeling errors. Leveraging

the LLM's instruction-following and few-shot capabilities, PromptMix produces more accurate augmented datasets, enhancing text classification performance. Zhang et al. (2024b) introduced *TELEClass*, a framework for hierarchical text classification that enriches label taxonomy with class-indicative terms and uses LLMs for tailored data annotation, thereby optimizing classifier training and performance. Therefore, future studies could investigate the performance of LLMs in categorizing questions at a more granular, subcategory level to gain deeper insights into their classification capabilities.

Overall, the results of Experiment 1-3 and 5-8 suggest that GPT-4 can classify questions according to Eris' taxonomy with moderate accuracy in terms of alignment with the human-sourced labels. Beyond accuracy, GPT-4 also demonstrated notable efficiency in completing the task of labeling 30-60 new questions in the test set, with runtime varying from just a few seconds to several minutes, depending on the complexity of the input and the size of the generated response. This efficiency is particularly noteworthy when compared to the traditionally time-intensive process of protocol analysis conducted by human coders (Eris, 2004). In a typical scenario, human coders must meticulously review each question, apply the appropriate classification scheme, and ensure consistency across the coding process—a task that often requires considerable time and effort. Additionally, human coders must undergo training to become familiar with a specific taxonomy, which further extends the time required to initiate the coding process. The manual approach is not only time-consuming but also labor-intensive, requiring sustained attention and focus to maintain accuracy and consistency across numerous questions. This can be particularly challenging when dealing with a large dataset, as fatigue and cognitive load can impact a coder's ability to complete the task in one sitting. The repetitive nature of the task may lead to slower progress, increased likelihood of errors, and potential inconsistencies in the application of the classification scheme. Moreover, the need for multiple coders to cross-check each other's work to ensure reliability further compounds the time required to complete the analysis. In contrast, GPT-4's ability to process and label questions rapidly and consistently, without the need for breaks or retraining, offers a significant advantage in protocol analysis. While GPT-4 can label 30 questions in just a few minutes, human coders might take several hours due to the meticulous review, application of classification schemes, and consistency checks required. This substantial difference in processing time underscores the efficiency gains

that AI can bring to design research.

## 5.2 The role of context in the question classification task

When human coders perform the identification and classification of questions, they are instructed to take into consideration not only the question as stated but also the context in which the question is asked for clues (Eris, 2004, p. 99). This includes what the participants have discussed earlier and how the question receiver responds to the question. Similarly, in LLMs, "context" refers to the surrounding text or information that the model uses to generate responses or predictions. It includes everything the model "knows" up to that point in the conversation or text sequence, allowing it to maintain coherence, interpret meanings, and provide relevant responses (Brown et al., 2020). **RQ-3** specifically investigates *whether incorporating such contextual information improves GPT-4's performance in automated question classification.* Experiment 4 addressed this by exploring the impact of providing additional context to GPT-4 (varying from providing the entire session transcript to just the utterance immediately following the question) on the alignment with human-supplied labels.

The results indicated that providing any additional context not only does not improve but can also degrade performance. The model failed to label questions and even confused questions from the in-context examples and test set. This observation of the relationship between long context window and performance aligns with recent findings (Curry et al., 2024; Liu et al., 2024), which indicate that long input contexts are not utilized properly by current LLMs and ChatGPT. Since LLMs generate responses based on probabilities, they may prioritize different aspects of the context in ways that aren't always beneficial. When extra context is added, the model might focus on less relevant details or interpret the additional information in a way that leads to suboptimal predictions. Liu et al. (2024) found that altering the position of relevant information within the input can negatively affect model performance. Interestingly, they observed that decoder-only LLMs like GPT or Claude handle information placed at the beginning or end of the input context effectively,

but struggle to access information located in the middle, leading to a U-shaped performance curve. The findings reported by (Curry et al., 2024, p. 7) mirror those observed in this study: ChatGPT seems to *"conflate and modify data, likely owing to its probabilistic approach to synthesis and its goal to produce content, regardless of the input"*.

This finding prompts the need for further investigation into the factors contributing to this decrease in effectiveness. However, it's important to note that context can be beneficial in certain scenarios. For instance, retrieval-augmented generation (RAG) can enhance language model performance by conditioning on relevant documents from a grounding corpus during generation, thereby improving factual accuracy and providing source attribution (Gao et al., 2024). Similarly, methods such as Concept-aware Training (CoAT) construct training scenarios that encourage models to utilize latent reasoning concepts from demonstrations, leading to more robust in-context learning (Štefánik et al., 2024). *LLMLingua* (Jiang et al., 2023) employs small language models (SLMs) like *GPT-2 Small* or *LLaMA-7B* to identify and filter out unimportant tokens, rendering the text in a format that may be hard for humans to interpret but remains clear to LLMs. This technique offers an efficient, straightforward solution for prompt compression, preserving the natural language structure and achieving effective compression without requiring further training of LLMs. These approaches demonstrate that, when appropriately managed, context can significantly enhance the performance and reliability of language models. Therefore, future research could examine why extended context did not yield the expected improvements in this study, focusing on understanding how context length, relevance, and position impact performance. It may be beneficial to investigate refined approaches that define and manage 'context' with greater precision, particularly in NLP tasks where nuanced, context-rich responses are essential. This research could help clarify the conditions under which context contributes positively to model performance and reveal when context instead leads to diminished effectiveness.

Additionally, it's important to note that this study focused solely on question classification in a single-pass setting, without engaging GPT-4 interactively or conversationally. In this setup, GPT-4 classified questions based on isolated prompts without continuity or contextual buildup from previous exchanges. However, if GPT-4 were used in a conversational manner, where responses build on previous interactions, it could provide new

insights into LLMs like GPT-4's potential for handling question classification by leveraging conversational context and advanced prompting techniques. For instance, *Chain of Thought (CoT)* prompting technique encourages models to generate intermediate reasoning steps, leading to more accurate and interpretable outputs by guiding the model through its thought process (Wei et al., 2024). Similarly, *Least-to-Most* prompting involves decomposing complex problems into simpler subproblems, which are solved sequentially, enabling the model to address each component step-by-step (Zhou et al., 2023). *Self-Consistency* prompting enhances reasoning capabilities by generating multiple outputs and selecting the most consistent answer, thus leveraging diverse reasoning paths to arrive at a reliable solution (Wang et al., 2023). *ReAct* prompting framework integrates reasoning and action, allowing models to generate both reasoning traces and task-specific actions in an interleaved manner, facilitating more dynamic and context-aware responses (Yao et al., 2023). Moreover, the *Self-Refine* (Madaan et al., 2024) technique allows models to iteratively refine their outputs based on self-generated feedback, enhancing response quality. Future research should explore the application of these methods to determine their effectiveness in improving model performance and interoperability.

## 5.3 Broader considerations in using LLMs in design protocol analysis

Recent advancements in AI, specifically LLMs, show great potential but using them in design research requires consideration that LLMs are not inherently generalizable and need custom training to be task-specific, as evidenced by prior studies (Xiao et al., 2023a) and this study. Despite improvements in context awareness and nuanced text generation, LLMs still struggle with precise contextual understanding, especially with specialized language across different disciplines.

While GPT-4 is a highly advanced LLM, it often "hallucinates," producing plausible but incorrect or irrelevant information, which could degrade information quality (Achiam et al., 2023). To address this issue, researchers suggest a range of strategies, including feedback mechanisms, accessing external information, and early refinement during LLM

development. Some notable works are Retrieval-Augmented Generation (Lewis et al., 2020), Knowledge Retrieval (Varshney et al., 2023), and *CoNLI* (Lei et al., 2023). Prompt engineering is also crucial, as evidenced by this study's use of specifically developed prompts based on official guidelines (OpenAI, 2023). However, these may not always be the most effective. As noted by Wei et al. (2024), there is potential to refine prompts further to elicit better responses from LLMs, suggesting that more effective prompt versions could yield superior results. Also, using LLMs like GPT for qualitative analysis may introduce bias as they learn language from their training data (Perez et al., 2023). This can lead to biased insights, or ignoring marginalized groups' needs (Hu et al., 2023), potentially resulting in designs that exclude or alienate some users.

Another important consideration for design researchers looking to use GPT-4 in other protocol analysis applications is that despite its capabilities, it has several limitations intrinsic to the current state of LLM technologies. The models' capabilities and features are subject to almost daily updates—which, while improving performance, can impede consistent replicability and comparability of research findings over time (OpenAI, 2023b). Additionally, the absence of detailed update logs from OpenAI further amplifies the issue of opacity, making the reproduction of results and longitudinal studies problematic. Transparency is another concern, as proprietary algorithms and datasets hinder independent verification and pose challenges to adequately assessing bias and ethical considerations inherent to the training material.

## 5.4 Potential Applications of LLMs in Broader Qualitative Research Domains

In addition to classification tasks, LLMs hold significant potential for other qualitative research domains such as text summarization and idea generation. Summarization is crucial in qualitative research for condensing large volumes of data into coherent and concise summaries that capture the essence of the content. LLMs can be leveraged to automate this process, providing researchers with quick and accurate summaries of transcripts, interviews, and other textual data. This can enhance efficiency and allow researchers to focus on

deeper analysis and interpretation (Jin et al., 2024; Van Veen et al., 2023). The model's ability to synthesize information from various sources and present it in novel ways can stimulate innovative thinking and support the development of unique research questions and methodologies (Eschrich and Sterman, 2024). Overall, while this study focused on the classification of question utterances, the capabilities of GPT-4 extend far beyond this specific application. Future research should explore these additional functionalities to fully harness the potential of LLMs in qualitative research (Dunivin, 2024; Aubin Le Quéré et al., 2024).

# Chapter 6

# Conclusion

Driven by the broader research goal of investigating the extent to which AI tools can support the automation of qualitative analyses, this research aimed to examine generative AI's capabilities for analyzing verbal design protocols using Eris' question-asking taxonomy. A series of tasks were conducted to systematically assess GPT-4, a state-of-the-art generative AI model's proficiency in categorizing questions, and evaluate how different factors influence its precision. The preliminary experiments established initial performance benchmarks and investigated foundational aspects such as the impact of in-context learning, the number of in-context examples, the sensitivity of the results, and the role of context in classification. These experiments provided crucial insights into how varying these factors can influence the performance and consistency of LLM, highlighting its capabilities and identifying areas where it excels or needs improvement. The extended experiments built on the preliminary findings by examining the scalability and generalizability of the LLM-based classification system. These experiments focused on evaluating the impact of increased test set sizes, variations in prompt structures, cross-dataset training, and cross-model applicability.

The findings from these experiments revealed that LLMs like GPT-4 perform fairly well in classification tasks. Notably, this study implies that LLMs appear to be especially effective for deductive tasks where a coding scheme is already defined. Also, extensive training examples may not be as crucial, instead, the performance might rely on prompt

configurations and the structure of the test set. While GPT-4 generally aligns well with human-sourced labels in categorizing stand-alone question utterances, the addition of contextual information did not always improve classification accuracy and can sometimes make it worse, highlighting the challenges of context-dependent tasks. Moreover, LLMs demonstrated promising generalization capability across the datasets used in this study, without a significant drop in labeling accuracy. The study also highlights the potential for using less expensive proprietary LLM, Claude 3.5, without sacrificing performance, thereby providing a more accessible option for similar tasks. This study also implies that classification accuracy relies on the quality of the dataset, using a better quality dataset may improve the performance.

Despite these promising results, several limitations were identified. The study was limited to categorizing questions at the LLQ/DRQ/GDQ level, without exploring subcategories in depth. The challenges in handling context also point to areas needing improvement. Despite these limitations, the study underscores the potential of LLMs to enhance qualitative analysis by reducing the time and resources required for manual coding and minimizing subjectivity. Future work should address these limitations by investigating the performance of LLMs in categorizing sub-categories. Future research could conduct a thorough qualitative analysis of the response of LLMs which could potentially reveal other contributing factors to tasks like classification. Furthermore, appropriately managed context can enhance language model performance. Therefore, future research should explore how factors such as context length, relevance, and positioning impact model performance, to develop strategies to optimize context usage in classification tasks. Moreover, developing strategies to better manage long context inputs and examining the use of different datasets for training could enhance the robustness and applicability of LLMs in qualitative research. Future research could also explore open-source LLMs to assess their performance in comparison to proprietary models like GPT-4 and Claude 3.5.

Qualitative analysis benefits significantly from the integration of LLMs, offering a scalable and efficient means of coding and analyzing large datasets. However, the nuanced and interpretative nature of qualitative data presents unique challenges that LLMs must overcome to be fully effective. While LLMs can expedite the initial coding process, the interpretative layer added by human researchers remains crucial for nuanced understanding

and context-specific insights. Thus, a hybrid approach that combines the computational power of LLMs with the critical interpretative skills of human researchers may offer the most robust solution for qualitative analysis.

Overall, while this study focused on the classification of question utterances, the broader implications for qualitative research are substantial. By leveraging LLMs for qualitative tasks, researchers can enhance their methodological toolkit; making qualitative analysis more efficient and scalable while maintaining the depth and richness that are the hallmarks of qualitative inquiry.

# Bibliography


Achiam, J. et al. (2023). Gpt-4 technical report. *arXiv preprint*, arXiv:2303.08774.

An, S., Ma, Z., Lin, Z., Zheng, N., and Lou, J.-G. (2024). Make your llm fully utilize the context. *arXiv preprint arXiv:2404.16811*.

Anthropic (2024a). Claude 3 model card addendum. https://www-cdn.anthropic.com/fed9cc193a14b84131812372d8d5857f8f304c52/Model_Card_Claude_3_Addendum.pdf. [Accessed: 2024-08-07].

Anthropic (2024b). Claude 3.5 sonnet. https://www.anthropic.com/news/claude-3-5-sonnet. [Accessed: 2024-08-07].

Anthropic (2024c). Commercial terms of service - anthropic. https://www.anthropic.com/legal/commercial-terms. [Accessed: 2024-08-07].

Anthropic (2024d). Text completion api - anthropic. https://docs.anthropic.com/en/api/complete. [Accessed: 2024-08-07].

Arlinghaus, C. S., Wulff, C., Maier, G. W., Arlinghaus, C., Wulff, C., and Maier, G. (2024). Inductive coding with chatgpt-an evaluation of different gpt models clustering qualitative data into categories.

Aubin Le Quéré, M., Schroeder, H., Randazzo, C., Gao, J., Epstein, Z., Perrault, S. T., Mimno, D., Barkhuus, L., and Li, H. (2024). Llms as research tools: Applications and evaluations in hci data work. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, pages 1–7.

Aurisicchio, M., Bracewell, R., and Wallace, K. (2010). Understanding how the information requests of aerospace engineering designers influence information-seeking behaviour. *Journal of Engineering Design*, 21(6):707–730.

Aurisicchio, M., Bracewell, R. H., and Wallace, K. M. (2013). Characterising the information requests of aerospace engineering designers. *Research in Engineering Design*, 24:43–63.

Austin, J. and Delaney, P. F. (1998). Protocol analysis as a tool for behavior analysis. *The Analysis of verbal behavior*, 15:41–56.

Bano, M., Zowghi, D., and Whittle, J. (2023). Exploring qualitative research using llms. *arXiv preprint*, arXiv:2306.13298.

Bender, E. M., Gebru, T., McMillan-Major, A., and Shmitchell, M. (2021). On the dangers of stochastic parrots: Can language models be too big? *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 610–623.

Boonpracha, J. (2023). Scamper for creativity of students' creative idea creation in product design. *Thinking Skills and Creativity*, 48:101282.

Braun, V. and Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative Research in Psychology*, 3(2):77–101.

Brill, R., Graham, A., and Reading, M. (2020). Artificial intelligence in design: The state of the art and future directions. *Design Studies*, 66:99–122.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020). Language models are few-shot learners. In *Advances in neural information processing systems*, volume 33, pages 1877–1901.

Caetano, I., Santos, L., and Leitão, A. (2020). Computational design and artificial intelligence in architecture: A brief review. *Frontiers of Architectural Research*, 9(2):168–182.

Cardella, M. E., Atman, C. J., Turns, J., and Adams, R. S. (2008). Students with differing design processes as freshmen: Case studies on change. *International Journal of Engineering Education*, 24(2):246.

Cardoso, C., Badke-Schaub, P., and Eris, O. (2016). Inflection moments in design discourse: How questions drive problem framing during idea generation. *Design Studies*, 46:59–78.

Cardoso, C., Hurst, A., Badke-Schaub, P., and Aurisicchio, M. (2014). Question asking in design reviews: how does inquiry facilitate the learning interaction? In *10th Design Thinking Research Symposium*. Purdue University.

Cardoso, C., Hurst, A., and Nespoli, O. G. (2020). Reflective inquiry in design reviews: The role of question-asking during exchanges of peer feedback. *International Journal of Engineering Education*, 36(2):614–622.

Chandrasegaran, S., Salah, A., and Lloyd, P. (2023). Constructing design activity in words: Exploring linguistic methods to analyse the design process. *Design Studies*, 86:101182.

Chang, T.-Y. and Jia, R. (2023). Data curation alone can stabilize in-context learning. In Rogers, A., Boyd-Graber, J., and Okazaki, N., editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8123–8144, Toronto, Canada. Association for Computational Linguistics.

Chen, J., Chen, L., Zhu, C., and Zhou, T. (2023). How many demonstrations do you need for in-context learning? In Bouamor, H., Pino, J., and Bali, K., editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 11149–11159, Singapore. Association for Computational Linguistics.

Cheng, S., Gan, Z., Yang, Z., Wang, S., Wang, J., Boyd-Graber, J., and Wang, L. (2023). Prompting gpt-3 to be reliable. In *International Conference on Learning Representations (ICLR 23)*.

Chew, R., Bollenbacher, J., Wenger, M., Speer, J., and Kim, A. (2023). Llm-assisted content analysis: Using large language models to support deductive coding. *arXiv preprint arXiv:2306.14924*.

Christou, P. A. (2023). How to use artificial intelligence (ai) as a resource, methodological and analysis tool in qualitative research? *The Qualitative Report*, 28(7):1968–1980.

Cross, N. (2004). Expertise in design: an overview. *Design Studies*, 25(5):427–441.

Cross, N., Christiaans, H., and Dorst, K. (2001). Design cognition: Results from protocol and other empirical studies of design activity. *Design knowing and learning: Cognition in design education*, pages 79–103.

Curry, N., Baker, P., and Brookes, G. (2024). Generative ai for corpus approaches to discourse studies: a critical evaluation of chatgpt. *Applied Corpus Linguistics*, 4(1).

Dai, S.-C., Xiong, A., and Ku, L.-W. (2023). Llm-in-the-loop: Leveraging large language model for thematic analysis. *arXiv preprint arXiv:2310.15100*.

DALY, S. R., YILMAZ, S., CHRISTIAN, J. L., SEIFERT, C. M., and GONZALEZ, R. (2012). Design heuristics in engineering concept generation. *Journal of Engineering Education*, 101(4):601–629.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In Burstein, J., Doran, C., and Solorio, T., editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Dixon, R. A. and Bucknor, J. (2019). A comparison of the types of heuristics used by experts and novices in engineering design ideation. *Journal of Technology Education*, 30(2):39–59.

Dorst, K. (2008). Design research: a revolution-waiting-to-happen. In *Proceedings of the International Conference on Design Research Society*, pages 1–9.

Dorta, T. and Kalay, Y. E. (2015). The role of computers in design: from computer-aided design to computer-integrated design. In *Handbook of Research on Emerging Digital*

*Tools for Architectural Surveying, Modeling, and Representation*, pages 103–124. IGI Global.

Dunivin, Z. O. (2024). Scalable qualitative coding with llms: Chain-of-thought reasoning matches human performance in some hermeneutic tasks. *arXiv preprint arXiv:2401.15170*.

Edwards, A. and Camacho-Collados, J. (2024). Language models for text classification: Is in-context learning enough? In Calzolari, N., Kan, M.-Y., Hoste, V., Lenci, A., Sakti, S., and Xue, N., editors, *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 10058–10072, Torino, Italia. ELRA and ICCL.

Ericsson, K. A. (2006). Protocol analysis and expert thought: Concurrent verbalizations of thinking during experts' performance on representative tasks. *The Cambridge handbook of expertise and expert performance*, pages 223–241.

Ericsson, K. A. and Simon, H. A. (1984). *Protocol analysis: Verbal reports as data, revised edition*. The MIT Press.

Ericsson, K. A. and Simon, H. A. (1993). *Protocol Analysis: Verbal Reports as Data*. MIT Press.

Eris, O. (2004). *Effective inquiry in engineering design*. Kluwer Academic Publishers.

Eschrich, J. and Sterman, S. (2024). A framework for discussing llms as tools for qualitative analysis. *arXiv preprint arXiv:2407.11198*.

Feuston, J. L. and Brubaker, J. R. (2021). Putting tools in their place: The role of time and perspective in human-ai collaboration for qualitative analysis. *Proceedings of the ACM on Human-Computer Interaction*, 5(CSCW2):1–25.

Friese, S. (2019). *ATLAS.ti 8 User Manual*. ATLAS.ti Scientific Software Development GmbH.

Gamieldien, Y., Case, J. M., and Katz, A. (2023). Advancing qualitative analysis: An exploration of the potential of generative ai and nlp in thematic coding. *Available at SSRN 4487768*.

Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., Wang, M., and Wang, H. (2024). Retrieval-augmented generation for large language models: A survey.

Gero, J. S. (1990). Design prototypes: a knowledge representation schema for design. *AI magazine*, 11(4):26–36.

Gero, J. S. and Milovanovic, J. (2023). The situatedness of design concepts: Empirical evidence from design teams in engineering. In *Proceedings of the Design Society*, volume 3, pages 3503–3512.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680.

Graesser, A., Lang, K., and Horgan, D. (1988). A taxonomy for question generation. *Questioning Exchange*, 2(1):3–15.

Graesser, A. and Person, N. (1994). Question asking during tutoring. *American Educational Research Journal*, 31(1):104–137.

Hamilton, L., Elliott, D., Quick, A., Smith, S., and Choplin, V. (2023). Exploring the use of ai in qualitative analysis: A comparative study of guaranteed income data. *International Journal of Qualitative Methods*, 22.

Hay, L., Duffy, A. H., McTeague, C., Pidgeon, L. M., Vuletic, T., and Grealy, M. (2017). A systematic review of protocol studies on conceptual design cognition: Design as search and exploration. *Design Science*, 3:e10.

Hosseini, P., Castro, I., Ghinassi, I., and Purver, M. (2024). Efficient solutions for an intriguing failure of llms: Long context window does not mean llms can analyze long sequences flawlessly. *arXiv preprint arXiv:2408.01866*.

Hu, T., Kyrychenko, Y., Rathje, S., Collier, N., van der Linden, S., and Roozenbeek, J. (2023). Generative language models exhibit social identity biases. *arXiv preprint*, arXiv:2310.15819.

Huang, Z., Wang, Z., Xia, S., and Liu, P. (2024). Olympicarena medal ranks: Who is the most intelligent ai so far? *arXiv preprint arXiv:2406.16772*.

Hurst, A., Duong, C., Flus, M., Litster, G., Nickel, J., and Dai, A. (2021). Evaluating peer-led feedback in asynchronous design critiques: A question-centered approach. In *ASEE Virtual Annual Conference Content Access*.

Hurst, A., Lin, S., Treacy, C., Nespoli, O. G., and Gero, J. S. (2023). Comparing academics and practitioners q&a tutoring in the engineering design studio. In *Proceedings of the Design Society*, volume 3, pages 997–1006.

Jiang, H., Wu, Q., Lin, C.-Y., Yang, Y., and Qiu, L. (2023). LLMLingua: Compressing prompts for accelerated inference of large language models. In Bouamor, H., Pino, J., and Bali, K., editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13358–13376, Singapore. Association for Computational Linguistics.

Jiang, H. and Yen, C. (2009). Protocol analysis in design research: a review. *Journal Paper*, 78(24):16.

Jin, H., Zhang, Y., Meng, D., Wang, J., and Tan, J. (2024). A comprehensive survey on process-oriented automatic text summarization with exploration of llm-based methods. *arXiv preprint arXiv:2403.02901*.

Jobin, A., Ienca, M., and Vayena, E. (2019). The global landscape of ai ethics guidelines. *Nature Machine Intelligence*, 1(9):389–399.

Jurafsky, D. and Martin, J. H. (2023). *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech cognition*. Pearson/Prentice Hall, 3rd edition. draft: https://web.stanford.edu/ jurafsky/slp3/ [Accessed 23 Jan 2024].

Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., and Amodei, D. (2020). Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Kazemitabaar, M., Williams, J., Drosos, I., Grossman, T., Henley, A., Negreanu, C., and Sarkar, A. (2024). Improving steering and verification in ai-assisted data analysis with interactive task decomposition. *arXiv preprint arXiv:2407.02651*.

Kelly, N. and Gero, J. S. (2022). Reviewing the concept of design frames towards a cognitive model. *Design Science*, 8:e30.

Kern, F. B., Wu, C.-T., and Chao, Z. C. (2023). Assessing novelty, feasibility and value of creative ideas with an unsupervised approach using gpt-4. *British Journal of Psychology*.

Khanolkar, P. M., Vrolijk, A., and Olechowski, A. (2023). Mapping artificial intelligence-based methods to engineering design stages: a focused literature review. *AI EDAM*, 37:e25.

Kilian, A. and Ochsendorf, J. (2006). Evolutionary digital design: A new method for optimizing the design of structures. In *International Association for Shell and Spatial Structures (IASS) Symposium*, pages 20–23.

Lehnert, G. W. (1978). *The process of question answering*. Lawrence Erlbaum Associates, Hillsdale, New Jersey.

Lei, D., Li, Y., Wang, M., Yun, V., Ching, E., and Kamal, E. (2023). Chain of natural language inference for reducing large language model ungrounded hallucinations. *arXiv preprint*, arXiv:2310.03951.

Lennon, R. P., Fraleigh, R., Van Scoy, L. J., Keshaviah, A., Hu, X. C., Snyder, B. L., Miller, E. L., Calo, W. A., Zgierska, A. E., and Griffin, C. (2021). Developing and testing an automated qualitative assistant (aqua) to support qualitative analysis. *Family medicine and community health*, 9(Suppl 1).

Leung, R. and Lo, I. S. (2024). Can chatgpt inspire me? evaluate students' questioning techniques on ai tool for overcoming fixation. In *ENTER e-Tourism Conference*, pages 75–86. Springer.

Levy, M., Jacoby, A., and Goldberg, Y. (2024). Same task, more tokens: the impact of input length on the reasoning performance of large language models. In Ku, L.-W., Martins, A., and Srikumar, V., editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15339–15353, Bangkok, Thailand. Association for Computational Linguistics.

Lewis, P. et al. (2020). Retrieval-augmented generation for knowledge intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Li, D., Shao, R., Xie, A., Sheng, Y., Zheng, L., Gonzalez, J., Stoica, I., Ma, X., and Zhang, H. (2023a). How long can context length of open-source LLMs truly promise? In *NeurIPS 2023 Workshop on Instruction Tuning and Instruction Following*.

Li, M., Gong, S., Feng, J., Xu, Y., Zhang, J., Wu, Z., and Kong, L. (2023b). In-context learning with many demonstration examples. *arXiv preprint arXiv:2302.04931*.

Li, S., Huang, X., Wang, T., Zheng, J., and Lajoie, S. P. (2024a). Using text mining and machine learning to predict reasoning activities from think-aloud transcripts in computer assisted learning. *Journal of Computing in Higher Education*, pages 1–20.

Li, T., Zhang, G., Do, Q. D., Yue, X., and Chen, W. (2024b). Long-context llms struggle with long in-context learning. *CoRR*, abs/2404.02060.

Lim, L., Bannert, M., van der Graaf, J., Molenaar, I., Fan, Y., Kilgour, J., Moore, J., and Gašević, D. (2021). Temporal assessment of self-regulated learning by mining students' think-aloud protocols. *Frontiers in Psychology*, 12:749749.

Litster, G. and Hurst, A. (2021). Protocol analysis in engineering design education research: observations, limitations, and opportunities. *Studies in Engineering Education*, 1(2).

Liu, J., Shen, D., Zhang, Y., Dolan, B., Carin, L., and Chen, W. (2022). What makes good in-context examples for GPT-3? In Agirre, E., Apidianaki, M., and Vulić, I.,

editors, *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 100–114, Dublin, Ireland and Online. Association for Computational Linguistics.

Liu, N. et al. (2024). Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Lones, M. A. (2024). Avoiding common machine learning pitfalls. *Patterns*, page 101046.

Lu, Y., Bartolo, M., Moore, A., Riedel, S., and Stenetorp, P. (2022). Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In Muresan, S., Nakov, P., and Villavicencio, A., editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8086–8098, Dublin, Ireland. Association for Computational Linguistics.

Madaan, A., Tandon, N., Gupta, P., Hallinan, S., Gao, L., Wiegreffe, S., Alon, U., Dziri, N., Prabhumoye, S., Yang, Y., Gupta, S., Majumder, B. P., Hermann, K., Welleck, S., Yazdanbakhsh, A., and Clark, P. (2024). Self-refine: iterative refinement with self-feedback. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

Matheson, J. L. (2007). The voice transcription technique: Use of voice recognition software to transcribe digital interview data in qualitative research. *Qualitative Report*, 12(4):547–560.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint*, arXiv:1301.3781.

Min, S., Lewis, M., Zettlemoyer, L., and Hajishirzi, H. (2022). MetaICL: Learning to learn in context. In Carpuat, M., de Marneffe, M.-C., and Meza Ruiz, I. V., editors, *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2791–2809, Seattle, United States. Association for Computational Linguistics.

Mogavi, R., Deng, C., Kim, J., Zhou, P., Kwon, Y., Metwally, A., Tlili, A., Bassanelli, S., Bucchiarone, A., Gujar, S., and Nacke, L. (2024). Chatgpt in education: A blessing or a curse? a qualitative study exploring early adopters' utilization and perceptions. *Computers in Human Behavior: Arts and Humanities*, 2(1):100027.

Naveed, H., Khan, A. U., Qiu, S., Saqib, M., Anwar, S., Usman, M., Akhtar, N., Barnes, N., and Mian, A. (2023). A comprehensive overview of large language models. *arXiv preprint arXiv:2307.06435*.

Nenkova, A. and McKeown, K. (2011). Automatic summarization. *Foundations and Trends in Information Retrieval*, 5(2–3):103–233.

Nespoli, O. G., Hurst, A., and Gero, J. S. (2021). Exploring tutor-student interactions in a novel virtual design studio. *Design Studies*, 75:101019.

OpenAI (2023). Api reference - openai api. https://platform.openai.com/docs/api-reference/chat/create#chatcreatetemperature. [Accessed 10 January 2024].

OpenAI (2023a). Chat completions api - openai api. https://platform.openai.com/docs/guides/chat-completions. [Accessed: 2024-08-07].

OpenAI (2023b). Gpt-4 technical report.

OpenAI (2023a). How to make your completions outputs consistent with the new seed parameter — openai cookbook. https://cookbook.openai.com/examples/reproducible_outputs_with_the_seed_parameterpro. [Accessed 10 Jan 2024].

OpenAI (2023b). Models - openai api. https://platform.openai.com/docs/models/gpt-4-and-gpt-4-turbo. [Accessed 10 Jan 2024].

OpenAI (2023). Models — how we use your data - openai api. https://platform.openai.com/docs/models/how-we-use-your-data. [Accessed: 2024-08-07].

OpenAI (2023). Prompt engineering - openai api. https://platform.openai.com/docs/guides/prompt-engineering. [Accessed 10 April 2024].

OpenAI (2023). Reproducible outputs - openai usage. https://platform.openai.com/docs/advanced-usage/reproducible-outputs. [Accessed: 2024-08-07].

OpenAI (2023). Text generation — chat completions - openai api. https://platform.openai.com/docs/guides/text-generation/chat-completions-api. [Accessed 10 Jan 2024].

Paoli, S. D. (2023). Performing an inductive thematic analysis of semi-structured interviews with a large language model: An exploration and provocation on the limits of the approach. *Social Science Computer Review*, 0(0).

Pelaez, S., Verma, G., Ribeiro, B., and Shapira, P. (2024). Large-scale text analysis using generative language models: A case study in discovering public value expressions in ai patents. *Quantitative Science Studies*, 5(1):153–169.

Perez, E. et al. (2023). Discovering language model behaviors with model-written evaluations. In *Findings of the Association for Computational Linguistics*, page 13387–13434.

Perkins, M. and Roe, J. (2024). The use of generative ai in qualitative analysis: Inductive thematic analysis with chatgpt. *Journal of Applied Learning and Teaching*, 7(1).

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al. (2019). Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21:1–67.

Ray, P. P. (2023). Chatgpt: A comprehensive review on background, applications, key challenges, bias, ethics, limitations and future scope. *Internet of Things and Cyber-Physical Systems*, 3:121–154.

Rodgers, W., Al-Shaikh, S., and Khalil, M. (2023). Protocol analysis data collection technique implemented for artificial intelligence design. *IEEE Transactions on Engineering Management*, 71:6842–6853.

Rubin, O., Herzig, J., and Berant, J. (2022). Learning to retrieve prompts for in-context learning. In Carpuat, M., de Marneffe, M.-C., and Meza Ruiz, I. V., editors, *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2655–2671, Seattle, United States. Association for Computational Linguistics.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *nature*, 323(6088):533–536.

Sahu, G., Vechtomova, O., Bahdanau, D., and Laradji, I. (2023). PromptMix: A class boundary augmentation method for large language model distillation. In Bouamor, H., Pino, J., and Bali, K., editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5316–5327, Singapore. Association for Computational Linguistics.

Sarkar, P. and Chakrabarti, A. (2013). A support for protocol analysis for design research. *Design Issues*, 29(4):70–81.

Schlagwein, D. and Willcocks, L. (2023). 'chatgpt et al.': The ethics of using (generative) artificial intelligence in research and science.

Seawright, J. and Gerring, J. (2008). Case selection techniques in case study research: A menu of qualitative and quantitative options. *Political Research Quarterly*, 61(2):294–308.

Sherstinsky, A. (2020). Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network. *Physica D: Nonlinear Phenomena*, 404:132306.

Siiman, L., Rannastu-Avalos, M., Pöysä-Tarhonen, J., Häkkinen, P., and Pedaste, M. (2023). Opportunities and challenges for ai-assisted qualitative data analysis: An example from collaborative problem-solving discourse data. In *International Conference on Innovative Technologies and Learning*, pages 87–96.

Sinha, R., Solola, I., Nguyen, H., Swanson, H., and Lawrence, L. (2024). The role of generative ai in qualitative research: Gpt-4's contributions to a grounded theory analysis. In *Proceedings of the Symposium on Learning, Design and Technology*, pages 17–25.

Song, T. and Becker, K. (2014). Expert vs. novice: Problem decomposition/recomposition in engineering design. In *2014 International Conference on interactive collaborative learning (ICL)*, pages 181–190. IEEE.

Štefánik, M., Kadlčík, M., and Sojka, P. (2024). Concept-aware data construction improves in-context learning of language models. In Ku, L.-W., Martins, A., and Srikumar, V., editors, *Findings of the Association for Computational Linguistics: ACL 2024*, pages 12335–12352, Bangkok, Thailand. Association for Computational Linguistics.

Torii, M. G., Murakami, T., and Ochiai, Y. (2024). Expanding horizons in hci research through llm-driven qualitative analysis. *arXiv preprint arXiv:2401.04138*.

van Manen, M. (2023). What does chatgpt mean for qualitative health research? *Qualitative Health Research*, 33(13):1135–1139.

Van Veen, D., Van Uden, C., Blankemeier, L., Delbrouck, J.-B., Aali, A., Bluethgen, C., Pareek, A., Polacin, M., Reis, E. P., Seehofnerova, A., et al. (2023). Clinical text summarization: adapting large language models can outperform human experts. *Research Square*.

Varshney, N., Yao, W., Zhang, H., Chen, J., and Yu, D. (2023). A stitch in time saves nine: Detecting and mitigating hallucinations of llms by validating low-confidence generation. *arXiv preprint*, arXiv:2307.03987.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Walsh, I., Sinead, M., and Victoria, G. (2015). *NVivo 11 Essentials*. SAGE Publications.

Wang, W., Yang, H., and Meinel, C. (2024). Secokd: Aligning large language models for in-context learning with fewer shots.

Wang, X., Wei, J., Schuurmans, D., Le, Q. V., Chi, E. H., Narang, S., Chowdhery, A., and Zhou, D. (2023). Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*.

Wei, J., Tay, Y., Bommasani, R., Raffel, C., Zoph, B., Borgeaud, S., Yogatama, D., Bosma, M., Zhou, D., Metzler, D., Chi, E. H., Hashimoto, T., Vinyals, O., Liang, P., Dean, J., and Fedus, W. (2022). Emergent abilities of large language models. *Transactions on Machine Learning Research*. Survey Certification.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E. H., Le, Q. V., and Zhou, D. (2024). Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Xiao, Z., Yuan, X., Liao, Q. V., Abdelghani, R., and Oudeyer, P. Y. (2023a). Supporting qualitative analysis with large language models: Combining codebook with gpt-3 for deductive coding. In *Computational Proceedings of the 28th International Conference on Intelligent User Interfaces*, page 75–78.

Xiao, Z., Yuan, X., Liao, Q. V., Abdelghani, R., and Oudeyer, P.-Y. (2023b). Supporting qualitative analysis with large language models: Combining codebook with gpt-3 for deductive coding. In *Companion proceedings of the 28th international conference on intelligent user interfaces*, pages 75–78.

Yan, J., Yan, P., Chen, Y., Li, J., Zhu, X., and Zhang, Y. (2024). Gpt-4 vs. human translators: A comprehensive evaluation of translation quality across languages, domains, and expertise levels. *arXiv preprint arXiv:2407.03658*.

Yao, B., Chen, G., Zou, R., Lu, Y., Li, J., Zhang, S., Sang, Y., Liu, S., Hendler, J., and Wang, D. (2024). More samples or more prompts? exploring effective few-shot in-context learning for LLMs with in-context sampling. In Duh, K., Gomez, H., and Bethard, S., editors, *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 1772–1790, Mexico City, Mexico. Association for Computational Linguistics.

Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K. R., and Cao, Y. (2023). React: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations*.

Yuan, L., Chen, Y., Cui, G., Gao, H., Zou, F., Cheng, X., Ji, H., Liu, Z., and Sun, M. (2024). Revisiting out-of-distribution robustness in nlp: benchmark, analysis, and llms evaluations. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

Zhang, H., Wu, C., Xie, J., Kim, C., and Carroll, J. M. (2023). Qualigpt: Gpt as an easy-to-use tool for qualitative coding. *arXiv preprint arXiv:2310.07061*.

Zhang, J., Borchers, C., Aleven, V., and Baker, R. S. (2024a). Using large language models to detect self-regulated learning in think-aloud protocols. In *Proceedings of the 17th International Conference on Educational Data Mining (EDM)*.

Zhang, Y., Feng, S., and Tan, C. (2022). Active example selection for in-context learning. In Goldberg, Y., Kozareva, Z., and Zhang, Y., editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9134–9148, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Zhang, Y., Yang, R., Xu, X., Xiao, J., Shen, J., and Han, J. (2024b). Teleclass: Taxonomy enrichment and llm-enhanced hierarchical text classification with minimal supervision. *CoRR*, abs/2403.00165.

Zhao, W. et al. (2023). A survey of large language models. *arXiv preprint*, arXiv:2303.18223.

Zhao, Z., Wallace, E., Feng, S., Klein, D., and Singh, S. (2021). Calibrate before use: Improving few-shot performance of language models. In *International conference on machine learning*, pages 12697–12706. PMLR.

Zhou, D., Schärli, N., Hou, L., Wei, J., Scales, N., Wang, X., Schuurmans, D., Cui, C., Bousquet, O., Le, Q. V., and Chi, E. H. (2023). Least-to-most prompting enables complex reasoning in large language models. In *The Eleventh International Conference on Learning Representations*.

Zohny, H., McMillan, J., and King, M. (2023). Ethics of generative ai.

# APPENDICES

# Appendix A

# System Prompt

## A.1 Complete System Prompt

You are an expert in protocol analysis of design activity. You can perform analysis of the textual data and are able to label question utterances according to Eris' (2004) taxonomy. According to Eris, a question in a design context is "a verbal utterance related to the design tasks at hand that demands an explicit verbal and/or nonverbal response". His taxonomy, categorizes questions according to three high-level categories, with several sub-categories. The categories with examples are shown below:

Low Level Questions (LLQ)
Low-level questions are primarily information-seeking questions and they are formulated when the questioners want clarification about a given topic/event or are trying to obtain missing information. Different types of low level questions, with examples, are provided below.

Verification
Questioner wants to know the truth of an event.
Example: Did John leave?

Disjunctive

Verification question with multiple concepts.
Example: Was John or Mary here?

Definition
Questioner wants to know the meaning of a concept.
Example: What is a t test?

Example
Questioner wants to know an example of a concept.
Example: What is an example of a factorial design?

Feature Specification
Questioner wants to know some property of a person or thing.
Example: What are the properties of a bar graph?

Concept Completion
Questioner wants to know the missing component in a specified concept.
Example: Who ran the experiment?

Quantification
Questioner wants to know an amount.
Example: How many people are here?

Comparison
Questioner wants to compare properties of a person or thing.
Example: What is the difference between a t test and an F test?

Judgemental
Questioner wants to solicit a judgement by making an explicit reference to the mind of the person being addressed.
Example: Do you like tea or coffee?

Deep Reasoning Questions (DRQ)
Low-level (LLQ) and Deep Reasoning Questions (DRQ) share the common premise that a specific answer, or a specific set of answers, exists. As the purpose of these questions is either to seek for information (i.e. low level questions)

or to establish causal explanations of phenomenon (i.e. deep reasoning questions), they facilitate convergent thinking processes. Answers to these types of questions are expected to hold truth-value because the questioner assumes the person answering them to believe his/her answers to be true. Different types of Deep Reasoning Questions, with examples, are provided below.

Interpretation
Questioners wants to know what concept or claim can be inferred from a given dataset or situation.
Example: What is happening in this graph?

Goal Orientation/Rationale
Questioner wants to know the motives or goals behind an action.
Example: Why did John take the book?

Causal Antecedent
Questioner wants to know the states or events that have in some way caused the concept in question.
Example: Why did the glass break?

Causal Consequent
Questioner wants to know the concept or causal chain the question concept caused.
Example: What happened after John left?

Expectational
Questioner wants to know the causal antecedent of an act that presumably did not occur.
Example: Why didn't John go to New York?

Procedural
Questioner wants to know the partially or totally missing instrument or procedure in the question concept.
Example: How does a clock work?

Enablement (DRQ)
Questioner wants to know the act or state that enabled the concept in question.

Example: How was John able to get on the bus?

Generative Design Questions (GDQ)
Questions that are raised in design situations can operate quite differently from low-level or deep reasoning questions. Often, their premise is that there can be, regardless of being true or false, multiple alternative known answers as well as multiple unknown possible answers. The questioner's intention is to disclose the alternative known answers, and to generate the unknown possible ones. Such questions are characteristic of divergent thinking, where the questioner attempts to move away from the facts to the possibilities that can be generated from them. There are five GDQ categories:

Enablement (GDQ)
Questioner wants to know the multiple possible acts or states that can enable the concept in question.
Example: What allows you to measure distance?

Proposal/Negotiation
Questioner wants to suggest a concept, or to negotiate an existing or previously suggested concept.
Example: How about attaching a wheel to the long LEGO piece?

Method Generation
Questioner wants to generate as many ways as possible of achieving a specific goal.
Example: How can we keep the wheel from slipping?

Scenario Creation
Questioner wants to construct scenarios involving the question concept to investigate possible outcomes.
Example: What if the device was used on a child? What about people who have hair?

Ideation
Questioner wants to generate as many concepts as possible from an instrument

without trying to achieve a specific goal.
Example: Are magnets useful in any way? What else can we do with magnets?

## A.2 System Prompt With Only Definitions

You are an expert in protocol analysis of design activity. You can perform analysis of the textual data and are able to label question utterances according to Eris' (2004) taxonomy. According to Eris, a question in a design context is "a verbal utterance related to the design tasks at hand that demands an explicit verbal and/or nonverbal response". His taxonomy, categorizes questions according to three high-level categories, with several sub-categories. The categories and their definitions are shown below:

Low Level Questions (LLQ)
Low-level questions are primarily information-seeking questions and they are formulated when the questioners want clarification about a given topic/event or are trying to obtain missing information. Different types of low level questions are provided below.

Verification
Questioner wants to know the truth of an event.

Disjunctive
Verification question with multiple concepts.

Definition
Questioner wants to know the meaning of a concept.

Example
Questioner wants to know an example of a concept.

Feature Specification
Questioner wants to know some property of a person or thing.

Concept Completion
Questioner wants to know the missing component in a specified concept.

Quantification
Questioner wants to know an amount.

Comparison
Questioner wants to compare properties of a person or thing.

Judgemental
Questioner wants to solicit a judgement by making an explicit reference to the mind of the person being addressed.

Deep Reasoning Questions (DRQ)
Low-level (LLQ) and Deep Reasoning Questions (DRQ) share the common premise that a specific answer, or a specific set of answers, exists. As the purpose of these questions is either to seek for information (i.e. low level questions) or to establish causal explanations of phenomenon (i.e. deep reasoning questions), they facilitate convergent thinking processes. Answers to these types of questions are expected to hold truth-value because the questioner assumes the person answering them to believe his/her answers to be true. Different types of Deep Reasoning Questions, are provided below.

Interpretation
Questioners wants to know what concept or claim can be inferred from a given dataset or situation.

Goal Orientation/Rationale
Questioner wants to know the motives or goals behind an action.

Causal Antecedent
Questioner wants to know the states or events that have in some way caused the concept in question.

Causal Consequent
Questioner wants to know the concept or causal chain the question concept caused.

Expectational

Questioner wants to know the causal antecedent of an act that presumably did not occur.

Procedural
Questioner wants to know the partially or totally missing instrument or procedure in the question concept.

Enablement (DRQ)
Questioner wants to know the act or state that enabled the concept in question.

Generative Design Questions (GDQ)
Questions that are raised in design situations can operate quite differently from low-level or deep reasoning questions. Often, their premise is that there can be, regardless of being true or false, multiple alternative known answers as well as multiple unknown possible answers. The questioner's intention is to disclose the alternative known answers, and to generate the unknown possible ones. Such questions are characteristic of divergent thinking, where the questioner attempts to move away from the facts to the possibilities that can be generated from them. There are five GDQ categories:

Enablement (GDQ)
Questioner wants to know the multiple possible acts or states that can enable the concept in question.

Proposal/Negotiation
Questioner wants to suggest a concept, or to negotiate an existing or previously suggested concept.

Method Generation
Questioner wants to generate as many ways as possible of achieving a specific goal.

Scenario Creation
Questioner wants to construct scenarios involving the question concept to investigate possible outcomes.

Ideation

Questioner wants to generate as many concepts as possible from an instrument without trying to achieve a specific goal.

# Appendix B

# Experiment 5

## B.1 Subgroup Question Distribution

Table B.1: Distribution of Questions in Each Subgroup.
This table presents the breakdown of question types — Low-Level Questions (LLQ), Generative Design Questions (GDQ), and Deep Reasoning Questions (DRQ) — across six subgroups. The distribution helps in assessing the balance and representativeness of each type within the subgroups.

| **Subgroup** | **LLQ** | **GDQ** | **DRQ** |
|---|---|---|---|
| 1 | 7 | 2 | 1 |
| 2 | 7 | 2 | 1 |
| 3 | 6 | 3 | 1 |
| 4 | 6 | 3 | 1 |
| 5 | 7 | 3 | 0 |
| 6 | 7 | 3 | 0 |
| **Total** | **40** | **16** | **4** |

## B.2 Label Alignment Distribution

Table B.2: Label Alignment Summary of Categories.
This table presents the aggregated alignment counts (n) and corresponding percentages (%) for category labels that were correctly aligned with human category labels for each question type (LLQ, DRQ, GDQ) across multiple test sets. Percentages (%) indicate the proportion of labels that were correctly aligned for each question type across all test samples. For instance, in the LLQ column under "Six 10-Question Subsets," a total of 400 labels were evaluated across all test samples, with 326 correctly aligned, resulting in an alignment of 81.5%.

| | **Six 10-Question Subsets** | | | **Single 60-Question Set** | | |
|---|---|---|---|---|---|---|
| Qs Type | LLQ | DRQ | GDQ | LLQ | DRQ | GDQ |
| Alignment, n | 326 | 25 | 66 | 334 | 29 | 76 |
| (%) | (81.50) | (62.50) | (41.25) | (83.50) | (72.50) | (47.50) |

Table B.3: Label Alignment Summary of Subcategories.
This table presents the aggregated alignment counts (n) and corresponding percentages (%) for subcategory labels that were correctly aligned with human subcategory labels across different question types (LLQ, DRQ, GDQ) in multiple test sets. Percentages (%) indicate the proportion of subcategory labels that were correctly aligned for each question type across all ten test samples. For instance, in the LLQ column under "Six 10-Question Subsets," out of 400 questions across all ten test samples, only 213 subcategory labels were correctly aligned within the LLQ category, resulting in an alignment of 53.25%.

| | **Six 10-Question Subsets** | | | **Single 60-Question Set** | | |
|---|---|---|---|---|---|---|
| Qs Type | LLQ | DRQ | GDQ | LLQ | DRQ | GDQ |
| Alignment, n | 213 | 18 | 44 | 216 | 18 | 54 |
| (%) | (53.25) | (45.00) | (27.50) | (54.00) | (45.00) | (33.75) |

Table B.4: Subcategory Label Alignment Breakdown for Six 10-Question Subsets.

| **Qs Type** | **Subcategory** | **Frequency** | **Alignment, n (%)** |
|---|---|---|---|
| LLQ | Quantification | 10 | 8 (80.00) |
| | Verification | 262 | 155 (59.16) |
| | Disjunctive | 15 | 7 (46.67) |
| | Concept Completion | 51 | 14 (27.45) |
| | Judgmental | 20 | 7 (35.00) |
| | Feature Specification | 25 | 9 (36.00) |
| | Definition | 12 | 10 (83.33) |
| | Comparison | 5 | 3 (60.00) |
| DRQ | Instrumental/Procedural | 23 | 12 (52.17) |
| | Enablement | 2 | 0 (0.00) |
| | Rationale/Function/Goal Orientation | 12 | 5 (41.67) |
| | Interpretation | 3 | 1 (33.33) |
| GDQ | Method Generation | 56 | 20 (35.72) |
| | Proposal/Negotiation | 41 | 17 (41.46) |
| | Ideation | 34 | 2 (5.88) |
| | Enablement | 19 | 0 (0.00) |
| | Scenario Creation | 10 | 5 (50.00) |

Table B.5: Subcategory Label Alignment Breakdown for Single 60-Question Set.

| **Qs Type** | **Subcategory** | **Count** | **Alignment, n (%)** |
|---|---|---|---|
| LLQ | Quantification | 10 | 8 (80.00) |
| | Verification | 262 | 153 (58.40) |
| | Disjunctive | 15 | 7 (46.67) |
| | Concept Completion | 51 | 14 (27.45) |
| | Judgmental | 20 | 9 (45.00) |
| | Feature Specification | 25 | 11 (44.00) |
| | Definition | 12 | 11 (91.67) |
| | Comparison | 5 | 3 (60.00) |
| DRQ | Instrumental/Procedural | 23 | 14 (60.87) |
| | Enablement | 2 | 0 (0.00) |
| | Rationale/Function/Goal Orientation | 12 | 3 (25.00) |
| | Interpretation | 3 | 1 (33.33) |
| GDQ | Method Generation | 56 | 22 (39.29) |
| | Proposal/Negotiation | 41 | 23 (56.10) |
| | Ideation | 34 | 5 (14.71) |
| | Enablement | 19 | 0 (0.00) |
| | Scenario Creation | 10 | 4 (40.00) |

# Appendix C

# Experiment 6

## C.1 User Prompt With No Prompt Examples

I have a list of questions that I need you to classify according to the Eris Question-Asking Taxonomy (2004). Classify each of the questions below, delimited by triple backticks, using the taxonomy proposed by Eris. Label each question with one of the three categories: Low-level questions (LLQ), Deep Reasoning Questions (DRQ), or Generative Design Questions (GDQ). Each label should be either LLQ, DRQ, or GDQ. Additionally, provide a brief reasoning for the assigned label. Format the result as a markdown table with three columns: Question, GPT-label, and Reasoning.

``` < test set > ```
```

## C.2 Nemenyi Test Results

Table C.1: Nemenyi Test Results for Pairwise Comparisons of the Conditions. The column "Significant" indicates whether the difference between groups is statistically significant based on $p - value < 0.05$.

| **Group 1** | **Group 2** | **p-value** | **Significant ($\mathbf{p} < 0.05$)** |
|---|---|---|---|
| A | B | 0.0582 | No |
| A | C | 0.0010 | Yes |
| A | D | 0.0011 | Yes |
| B | C | 0.5560 | No |
| B | D | 0.6046 | No |
| C | D | 0.9000 | No |