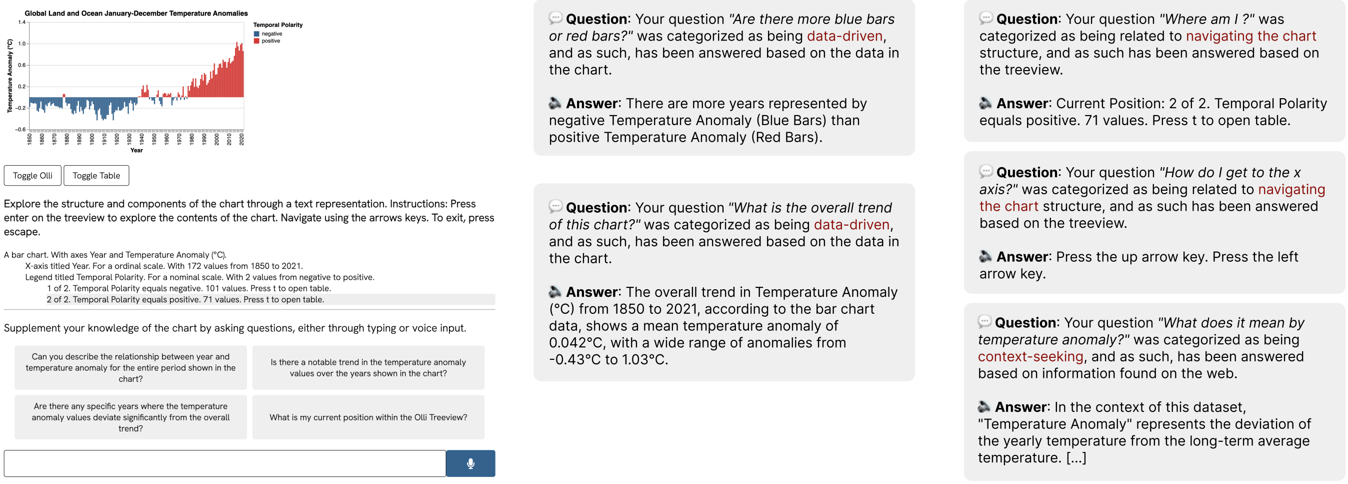
VizAbility: Enhancing Chart Accessibility with LLM-based Conversational Interaction

Joshua Gorniak
joshua.gorniak@bc.edu
Boston College
Chestnut Hill, Massachusetts, USA

Yoon Kim
yoonkim@mit.edu
MIT
Cambridge, Massachusetts, USA

Donglai Wei
donglai.wei@bc.edu
Boston College
Chestnut Hill, Massachusetts, USA

Nam Wook Kim
nam.wook.kim@bcu
Boston College
Chestnut Hill, Massachusetts, USA



Explore the structure and components of the chart through a text representation. Instructions: Press enter on the treeview to explore the contents of the chart. Navigate using the arrows keys. To exit, press escape.

A bar chart. With axes Year and Temperature Anomaly (°C).
    X-axis titled Year. For a ordinal scale. With 172 values from 1850 to 2021.
    Legend titled Temporal Polarity. For a nominal scale. With 2 values from negative to positive.
        1 of 2. Temporal Polarity equals negative. 101 values. Press t to open table.
        2 of 2. Temporal Polarity equals positive. 71 values. Press t to open table.

Supplement your knowledge of the chart by asking questions, either through typing or voice input.

Can you describe the relationship between year and temperature anomaly for the entire period shown in the chart?

Is there a notable trend in the temperature anomaly values over the years shown in the chart?

Are there any specific years where the temperature anomaly values deviate significantly from the overall trend?

What is my current position within the Olli Treeview?

💬 **Question**: Your question *"Are there more blue bars or red bars?"* was categorized as being data-driven, and as such, has been answered based on the data in the chart.

🔈 **Answer**: There are more years represented by negative Temperature Anomaly (Blue Bars) than positive Temperature Anomaly (Red Bars).

💬 **Question**: Your question *"What is the overall trend of this chart?"* was categorized as being data-driven, and as such, has been answered based on the data in the chart.

🔈 **Answer**: The overall trend in Temperature Anomaly (°C) from 1850 to 2021, according to the bar chart data, shows a mean temperature anomaly of 0.042°C, with a wide range of anomalies from -0.43°C to 1.03°C.

💬 **Question**: Your question *"Where am I ?"* was categorized as being related to navigating the chart structure, and as such has been answered based on the treeview.

🔈 **Answer**: Current Position: 2 of 2. Temporal Polarity equals positive. 71 values. Press t to open table.

💬 **Question**: Your question *"How do I get to the x axis?"* was categorized as being related to navigating the chart structure, and as such has been answered based on the treeview.

🔈 **Answer**: Press the up arrow key. Press the left arrow key.

💬 **Question**: Your question *"What does it mean by temperature anomaly?"* was categorized as being context-seeking, and as such, has been answered based on information found on the web.

🔈 **Answer**: In the context of this dataset, "Temperature Anomaly" represents the deviation of the yearly temperature from the long-term average temperature. [...]

Figure 1: VizAbility's overall user interface presents the keyboard navigation of chart content or data tables augmented with the ability to ask natural language questions. Example queries are shown on the right side, including visual & data queries, navigation, and context-seeking questions.

ABSTRACT

Traditional accessibility methods like alternative text and data tables typically underrepresent data visualization's full potential. Keyboard-based chart navigation has emerged as a potential solution, yet efficient data exploration remains challenging. We present VizAbility, a novel system that enriches chart content navigation with conversational interaction, enabling users to use natural language for querying visual data trends. VizAbility adapts to the user's navigation context for improved response accuracy and facilitates verbal command-based chart navigation. Furthermore, it can address queries for contextual information, designed to address the needs of visually impaired users. We designed a large language model (LLM)-based pipeline to address these user queries, leveraging chart data & encoding, user context, and external web knowledge. We conducted both qualitative and quantitative studies to evaluate VizAbility's multimodal approach. We discuss further opportunities based on the results, including improved benchmark testing, incorporation of vision models, and integration with visualization workflows.

CCS CONCEPTS

• **Human-centered computing** → **Interactive systems and tools**; **Visualization systems and tools**.

KEYWORDS

data visualization, accessibility, blind and low vision people

Conference acronym 'XX, June 03–05, 2018, Woodstock, NY

ACM Reference Format:
Joshua Gorniak, Yoon Kim, Donglai Wei, and Nam Wook Kim. 2018. VizAbility: Enhancing Chart Accessibility with LLM-based Conversational Interaction. In *Woodstock '18: ACM Symposium on Neural Gaze Detection, June 03–05, 2018, Woodstock, NY*. ACM, New York, NY, USA, 19 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Data visualization has become an indispensable tool in our broader society, aiding in the comprehension of important information and facilitating informed decision-making [48]. Its strength stems from leveraging the vast information bandwidth of our visual perception, which surpasses other sensory modalities [25]. However, an over-reliance on visual representation can inadvertently marginalize those with blindness or low vision (BLV), restricting their ability to engage with and understand data visualizations [52]. Individuals with BLV often come across data visualizations while using screen readers such as JAWS, NVDA, and VoiceOver to navigate the web [45, 61]. Unfortunately, a significant portion of data visualizations on the web remains largely inaccessible to this group [36, 61], resulting in a pronounced information gap.

Numerous assistive technologies have been developed to allow BLV users to access visualizations using sensory modalities other than vision [45]. Tactile visualizations can provide a tangible representation of data while necessitating specialized hardware such as haptic displays [55] and embossing machines [22]. On the other hand, sonification can enable users to discern trends and anomalies through sound [67], but it is typically limited to single-series data. Traditional methods for adapting web visualizations for screen readers include data tables and alternative text [45], while these methods often diminish the inherent advantages of data visualizations. Keyboard-based chart navigation [21, 64, 70, 72] have emerged as an alternative solution. However, orienting oneself and navigating within complex chart encoding structures pose challenges for efficient data exploration [44].

This work introduces VizAbility, a novel approach to augment keyboard navigation of chart content with conversational interaction (Figure 1). First, we use Olli's tree view (referenced in [14]) to create a keyboard-navigable text representation of a chart. Next, we enhance this tree view with an LLM-based question-and-answer module for addressing on-demand queries. These natural language queries allow users to understand the chart without needing to navigate and mentally synthesize different parts of the chart to derive insights. VizAbility uses the user's position within the tree view to efficiently respond to both VISUAL and ANALYTICAL queries, facilitating the exploration of data trends and visual patterns. Moreover, VizAbility can handle CONTEXTUAL queries, providing background information about the chart, tailored specifically for the needs of BLV individuals [43]. We based these query types on actual questions asked by BLV people in previous studies [42, 43]. Additionally, it can manage NAVIGATION queries, allowing users to control their position through verbal commands.

Our LLM-based pipeline first uses few-shot prompting to classify user queries into VISUAL, ANALYTICAL, CONTEXTUAL, and NAVIGATION queries. Once classified, VizAbility employs a query-specific prompting strategy. For analytical and visual queries, we aggregate both the chart's transformed data and color encoding into one CSV file, which is subsequently fed along with the keyboard-navigable text representation with the user's location [14] to the LLM via a CSV Agent [3]. CONTEXTUAL queries utilize a Web Browser Agent [4], whereas NAVIGATION queries employ the LLM to discern the starting/ending nodes from a user query and employ a breadth-search algorithm to calculate the shortest path between the nodes. We designed the prompts to minimize hallucinations and address unanswerable queries via structured output formatting. We collaborated with a blind participant in the development of VizAbility, holding a series of feedback sessions.

We carried out quantitative assessments to evaluate the question & answering pipeline. We evaluated response quality using a combined dataset of 979 real BLV user questions derived from previous research [43] and 48 synthetically generated navigation queries. Splitting the dataset, 80% was used for testing and 20% for validation. Our query classification achieved an accuracy of 87.39%. The final response evaluation involved a manual assessment by two researchers, followed by a more scalable LLM-based evaluation using GPT4. Both the human and GPT4 assessments followed a 5-point Likert Scale that assigned a value ranging from "Very Poor" to "Very Good" depending on the response's coherence to the ground truth. For the human evaluation, 77.36% and 5.14% of the responses were rated as "Very Good' and "Good" respectively, contributing to an overall rate of 82.50%. We computed Kendall's τ score (= 0.5526, $p < 0.001$) to assess the consistency between the human and GPT4 assessment methods, observing significant alignment. As a baseline comparison, we used the GPT4-based evaluation to assess responses generated by GPT-4 with vision (GPT-4V) on the same test data. We observed the performance of GPT-4V was poorer, with only 27.96% and 10.10% of responses being designated as "Very Good" and "Good" respectively.

We conducted a preliminary usability study with six BLV participants recruited through the National Institute for the Blind. Initially, participants explored VizAbility without guidance and were subsequently introduced to various query types. They also completed the System Usability Scale survey. The results suggest that while participants could learn to use the system, discerning query types without guidance proved challenging. Nonetheless, they acknowledged the merits of the integrated approach and offered suggestions for further improvements and potential applications. For instance, we introduced data tables as an alternative to the tree view and added cold-start query recommendations to assist users in getting started. Combining insights from both quantitative and qualitative evaluations, we identify potential avenues for future work. These include enhancing user-driven customization, developing a more robust benchmarking system, incorporating vision models, and integrating our solution into existing visualization tools.

Our main contributions lie in the following:

- Design and development of VizAbility, which incorporates an LLM-based question-and-answer module to enhance keyboard navigation of chart content for screen reader users
- Development of a benchmark dataset comprising ground truths and chart specifications that augment existing questions posed by blind individuals [43], facilitating further research in this area.
- Preliminary evaluation that demonstrates the effectiveness of VizAbility in comparison to baseline question-and-answer systems, as well as its usability as assessed by blind participants.

The VizAbility source code and dataset are available at [redacted for anonymous review].

2 RELATED WORK

2.1 Accessibility Systems for Data Visualization

The recent survey offers an overview of previous efforts exploring the use of non-visual modalities, such as speech, sound, and touch [45]. For example, **sonification** employs non-speech auditory channels, such as pitch and volume, to represent data [57, 67]. While this can offer users a swift overview of a graph, it struggles to communicate exact values and might not be effective beyond single-series charts [26]. An empirical study indicates that blind individuals favor speech over sonification, as the cognitive load for a sonified graph feels subjectively more intense [57].

Tactile systems employ methods like embossed prints, haptic feedback through vibrations, and braille for text representation. These systems enable both simultaneous and on-demand exploration of data trends, offering an advantage over linear audio [24]. However, they also necessitate enhanced perceptual motor skills. Similar to sonification, accurately discerning complex structures can be challenging, often demanding a more refined spatial resolution [22]. Producing tactile graphs typically involves specialized hardware, such as embossers, which might not be economically feasible for the average user [45]; thus, they are typically used and created in the field of education by teachers [23].

Screen readers, utilizing text/speech modalities, stand as the predominant assistive technology, particularly for navigating web content. The go-to accessibility techniques for screen readers encompass alternative text and data tables. Yet, these strategies often reduce data visualizations to brief descriptions or mere numbers, undermining their inherent advantages. An alternative approach involves crafting keyboard-navigable text descriptions derived from the chart's structure. A select group of data visualization tools and toolkits, such as HighCharts [34] and amCharts [1], offer some degree of this navigation and customization [44]. In recent times, several systems have advanced their navigation capabilities, representing charts as traversable graph structures [21, 30, 64, 70, 72].

Voice-based virtual assistants are emerging as valuable accessibility tools in human-computer interaction [65]. However, only a handful of studies have delved into using natural language interactions for accessing data visualization content. For instance, Murillo-Morales & Miesenberger [54] showcased a prototype system where users can ask predefined questions related to data metrics such as mean, extremes, and range. In a similar vein, VoxLens [60] facilitates voice-activated interactions capable of addressing basic queries with terms like "maximum" and "median". Additionally, Kim et al. [43] used a Wizard-of-Oz approach to study the types of questions blind individuals pose about charts.

To address the limitations of relying on a single sensory modality, **multi-sensory perception** is frequently utilized. A prevalent strategy involves merging verbal (speech) cues with non-verbal ones, such as sonification, tactile graphics, and haptic feedback. Examples include offering on-demand audio descriptions of touched elements [29, 31, 47] or pairing sonification with speech or screen readers [63, 64]. However, these solutions often necessitate specialized software and hardware, especially for interactive tactile support, making them expensive to implement.

In this study, we adopt an integrated approach that merges structured chart and table navigation using the keyboard with conversational interaction via verbal commands. Our work builds on the prior work [44] that suggests the respective advantages of data tables—familiarity, structured chart navigation—deeper engagement, and conversational interaction via natural language commands—faster data exploration. Our primary technical advancement centers on employing LLMs to enhance the current chart question-and-answer mechanism for the visually impaired.

2.2 Question & Answering Systems for Data Visualization

Within the realm of image understanding research, **visual question answering** has been rigorously explored in both natural language processing and computer vision, specifically regarding answering text-based queries about images [12, 38, 71]. Yet, the majority of these endeavors have centered on natural scene images rather than human-generated visuals such as data visualizations.

Recent studies have begun to focus on **data visualization images** [35]. For example, FigureQA [40] offers a corpus tailored for yes/no questions, such as "Is Light Gold less than Periwinkle?". Conversely, DVQA [39] expands its purview to encompass questions about chart structure ("are the bars horizontal?"), data retrieval ("what percent of people prefer A?"), and reasoning ("Is A preferred more than B?"). While both FigureQA and DVQA rely on synthetically generated charts, PlotQA introduces a large-scale dataset of real-world scientific plots. Unlike the templated questions of the aforementioned datasets, ChartQA delivers human-composed questions, enhanced using LLMs [53]. These models predominantly process pixel images as input. For instance, they extract data tables and other image features [41, 53], feeding them into vision and language task models [18]. Consequently, their accuracy largely hinges on their image processing capabilities, often leading to suboptimal results (e.g., failing to recover data values due to the absence of data labels). In a different approach, Kim et al.[42] proposed a system that not only answers questions but also provides explanations, operating on Vega-lite[59] instead of images. All the current question-answering systems are limited to basic visualization types like bar, line, and pie charts.

While chart QA systems hint at the potential for enhancing visualization accessibility, they often overlook the **specific needs of BLV users**. Recent studies have shown that BLV users frame questions differently compared to those with sight [20, 33]. A limited number of systems directly address the challenge of crafting question-and-answer systems tailored for the blind [54, 60]. However, these systems do not always offer specialized features for the blind and are constrained in their question-answering capabilities. For instance, VoxLens [60] is limited to charts with single series data, while the system by Murillo-Morales & Miesenberger [54] is restricted to bar charts. Kim et al. [43] have recently curated a set of questions posed by blind individuals through a wizard-of-oz study, laying the groundwork for more refined and targeted question-and-answer systems.

In this paper, we present an enhanced chart question-and-answer system tailored for blind users, leveraging the power of LLMs. Our approach focuses on reasoning, predicated on the availability of chart encoding and underlying data. We utilize Vega-lite as input, thereby accommodating a variety of chart types. The system handles a wide array of queries, including data and visual queries found in existing question-answer systems, as well as contextual and navigation queries specific to chart accessibility.

3 VIZABILITY DESIGN DECISIONS

In this section, we outline the key design decisions made *during the development and evaluation* of VizAbility. These decisions were guided by a combination of prior empirical research findings and practical considerations based on user feedback throughout the design and development process. D: Decisions relevant to our primary contributions D: User interface and usability decisions.

D1*: Enable understanding the chart encoding structure.* Bridging the perceptual gap between BLV and sighted individuals requires a deep understanding of what the chart looks like. While some blind individuals may not prioritize visual encoding information [51, 64], previous research indicates that navigating charts based on their visual encoding helps BLV users gain a clearer visual understanding [44]. In this work, we use Olli [14] to generate a keyboard-navigable text representation of charts.

D2*: Support efficient data exploration via natural language interaction.* Furthermore, extracting aggregate measures and discerning perceptual patterns beyond basic value retrievals becomes challenging when navigating data points individually via keyboard input [44]. This issue exacerbates as the hierarchical text representation becomes deeper and wider with complex chart encodings, resulting in hard mental operations [32]. In this work, we adopt a conversational interaction approach that transcends traditional methods such as sonification and tactile perception, which are limited in scalability for modern data visualizations. Leveraging LLMs and user context within keyboard navigation, we address visual and analytic queries that facilitate rapid exploration of nuanced trends and patterns in charts.

D3*: Provide contextual knowledge on demand for better chart comprehension.* Current chart question and answering systems often neglect the distinct types of questions posed by blind versus sighted individuals. Recent research involving blind participants indicates that they frequently ask contextual questions alongside data-related and visual inquiries [43]. These questions often seek external information not present in the chart, such as meanings about axes or specific data labels. Providing answers to these inquiries can enhance the self-efficacy and autonomy of blind individuals. In our approach, we use an LLM with web search capabilities to address these contextual queries.

D4*: Alleviate the difficulty of keyboard-based chart navigation.* Navigating complex chart structures can become less intuitive and more cumbersome [44, 72], when restricted to keyboard inputs alone. In our work, we aim to mitigate this challenge by facilitating nonlinear investigation across the chart structure via speech commands. In our work, we address this challenge by enabling nonlinear exploration of chart structures through speech commands. This multimodal approach enhances flexibility and efficiency. Furthermore, we aim to assist users in orienting themselves within the chart structure. Understanding one's position within a digital chart holds equal importance to spatial awareness in physical mobility [19].

D5*: Provide a fallback strategy using familiar data presentation format.* The hierarchical text representation of charts, while effective, can sometimes be seen as overly complex for certain users. This observation was noted in previous research [44], as well as in our user study (see Section 6). In response, we offer conventional data tables as an alternative to navigating the chart structure. This option is advantageous due to its compatibility with screen readers and widespread user familiarity [44, 72]. Additionally, we incorporate the user's context within the data table to enhance our system's ability to accurately respond to data-oriented queries.

D6*: Implement error prevention strategies for enhanced LLM interaction.* User queries can often be ambiguous or not directly related to the available data. Likewise, LLMs can face technical limitations, such as time-outs or processing errors, which can disrupt the interaction flow. Strategies to anticipate and mitigate these issues can help manage user expectations and offer a fluid user experience. We address these challenges by implementing proactive query refinement for ambiguous queries and reactive suggestions of alternative queries in case of failures.

D7*: Follow POUR accessibility principles.* The W3C Web Accessibility Initiative specifies four essential principles for web accessibility: **P**erceivable, **O**perable, **U**nderstandable, and **R**obust [10]. Ensuring these basic principles while developing a new assistive technology can be easily overlooked but is important to enhance its overall utility. We strive to adhere to these principles in the design of VizAbility. Examples include reaffirming user queries (perceivable); suggesting cold-start queries (operable), indicating delays in responses (understandable), and allowing both speech and text inputs to formulate queries (robust).

4 VIZABILITY SYSTEM INTERFACE & ARCHITECTURE

Below, we outline the input chart format for VizAbility, explain how VizAbility facilitates keyboard navigation and conversational interaction with the chart, and delve into further accessibility considerations integral to the design decisions outlined earlier.

4.1 Input Chart Format

VizAbility operates on the premise that both the visual encoding information and the underlying dataset are accessible. In our work, we employ Vega-Lite specifications [59] as the primary input for our system. Other chart specifications like Observable Plot [6] and HighCharts [34] can be adapted, provided they expose the underlying data and visual encoding variables. New adapters will need to be written to work with Olli, which currently supports Vega & Vega-Lite, and Observable Plot [6]. Alternatively, charts can be translated to the Vega-Lite specifications to be directly used in VizAbility. The symbolic representations underlying the chart are parsed to create a keyboard-navigable tree view and to provide useful information about the chart's appearance for the question-and-answer pipeline, which is detailed below.

4.2 Exploring Chart Encoding using Keyboard

We leverage Olli [14] to make the chart encoding explorable—D1, as it provides an off-the-shelf, open-source solution based on Vega-lite. Olli renders a visual chart for sighted users and a keyboard-navigable tree view featuring chart descriptions at various levels of detail (see Figure 2). A more detailed explanation of Olli, including its design considerations, supported chart types, and empirical evaluation, is available in prior work [14, 72].

Olli's tree view displays the chart content in a hierarchical structure, starting with the chart type description at the root, followed by visual encoding channels such as axes and legends. Within each encoding channel node, Olli lists data categories or numerical ranges depending on the data type being encoded; e.g., for a color legend, it lists all categories in the legend. Individual data points reside in these group nodes. All four chart types we used in this work, including line chart, bar chart, scatter plot, and choropleth map, had four levels of information granularity.

Based on its hierarchical structure, users can navigate the different levels of the tree view using up and down arrow keys (*barchart* → *legend* → *negative polarity*) while using left and right arrow keys to navigate sibling nodes within each level (*negative polarity* → *positive polarity*). In order to access individual data points, Olli requires users to press *t* to open up a screen-reader-compatible data table. This table shows a subset of the whole data, only displaying data points within the selected category or numerical range.

The current version of Olli **does not support navigating a choropleth map by geographic regions**. We extended it to support the level of *detail* channel in Vega-lite[1]. As a result, we can encode country names or state names into the *detail* channel, which is in turn converted into an additional encoding channel node (see Figure 2).

4.3 Exploring Underlying Data Table via Keyboard

VizAbility offers users the flexibility to switch between the tree view and a conventional raw data table view (see options displayed in the buttons in Figure 1)—D5. While the tree view facilitates structured exploration based on visual encoding, the data table provides additional advantages like sorting features, enabling users to quickly access specific data values and patterns. We disable navigation queries in the data table module as screen readers provide a slew of keyboard navigation shortcuts such as moving between headers and cells. The data table module supports the alphabetical/numeric sorting by each column.

4.4 Rapid Chart Probing via Conversational Interaction

Keyboard navigation helps blind users grasp the chart's visual structure and data, but can be cumbersome for higher-level comprehension, such as computing aggregates or understanding overall visuals. We integrate LLMs for intuitive chart question-answering and enhance interaction by combining keyboard navigation with speech inputs.

4.4.1 Data Set.

We utilized a prior study's data set [43], comprising 979 BLV user questions spanning four visual stimuli (bar, line, scatter, and map) for the development and quantitative evaluation of VizAbility (Table 1). We then partition the pool of questions once more into an 80/20 split between the testing and validation sets via stratified random sampling so that there is a proportionate representation of each query type amongst both sets.

The questions were gathered through a wizard-of-oz study, where a human facilitator acted as a question-answering system. We reconstructed the visualization images into Vega-Lite specifications and partitioned the questions into ANALYTICAL, VISUAL, and CONTEXTUAL queries, which we derived from the question taxonomy in prior work [42, 43]. In addition, we employed GPT4 to generate 12 example NAVIGATION queries for each of the four visual stimuli, appending them to the prior data set of 979 BLV user questions. We define each query type in Section 4.4.2.

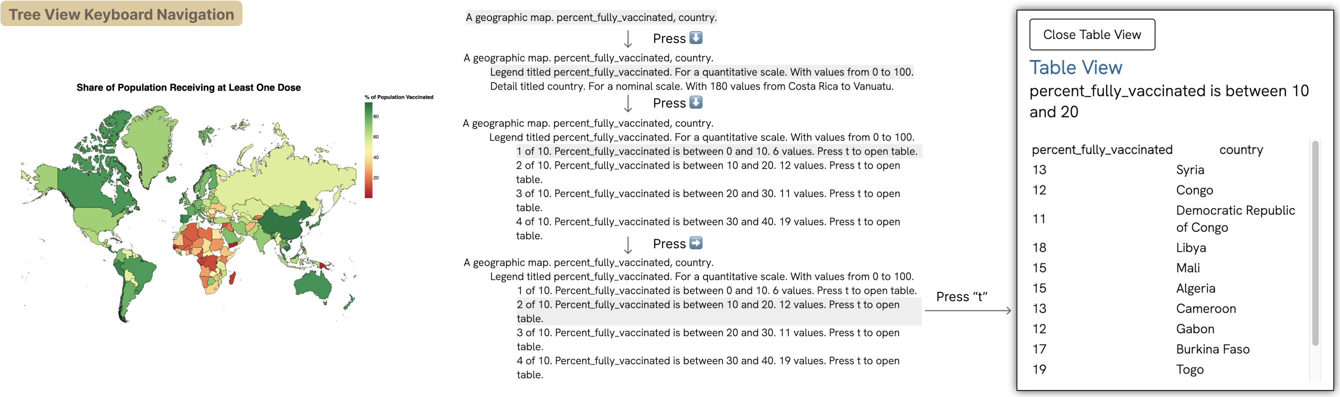
How did we derive the question taxonomy? The original taxonomy by Kim et al.[43] offers more fine-grained categories, such as granular analytical tasks like retrieving values and finding extremum[11]. They also categorized queries as visual vs. non-visual and look-up vs. compositional, similar to previous chart QA systems [42], while separating non-data queries. In our work, we reclassify these queries, as the detailed categorization is not needed for users and may not enhance the system's performance, considering the advanced language comprehension capabilities of LLMs. First, we consolidated data-driven and visual tasks into ANALYTICAL (or non-visual) and VISUAL queries, respectively. We do not separately consider look-up vs. compositional, as we expect LLMs to handle them without differentiation. Finally, we consolidated non-data queries into CONTEXTUAL queries.

What additional metrics did we introduce? We also elaborate on the previous taxonomy by introducing two new binary metrics: OPEN-ENDEDNESS and ANSWERABILITY. Open-ended queries allow for multiple valid interpretations and responses, and often invoke questions like "Why?". Open-ended questions may also involve computations, especially when the operation parameters are ambiguous: "And what was that temperature?". Answerable queries are relevant to the data set. By contrast, unanswerable queries are irrelevant and cannot be feasibly addressed using the corresponding chart.

How did we generate navigation queries? We used GPT-4 with few-shot prompting to generate NAVIGATION queries with an equal distribution of orientation and wayfinding queries. To simulate real user chart interaction, we include the corresponding active element in the treeview for each query.

How did we generate ground-truths? The ground truths for the testing and validation sets were manually generated by two researchers independently and then merged by resolving conflicts. The process involved reading charts, calculating numbers, and searching for information online. Throughout the generation process, we emphasized verboseness. For instance, the ground truth response to the question "What is the vaccination rate of South Africa" is "The vaccination rate for South Africa is 36%", as opposed to the more concise "36%". To prioritize the model's ability to translate ambiguous user wording into precise nodes within the treeview,

[1] https://vega.github.io/vega-lite/docs/encoding.html#detail


Tree View Keyboard Navigation
Share of Population Receiving at Least One Dose
% of Population Vaccinated
80
60
40
20
Press ⬇
Press ⬇
Press ➡
Press "t"

A geographic map. percent_fully_vaccinated, country.

A geographic map. percent_fully_vaccinated, country.
Legend titled percent_fully_vaccinated. For a quantitative scale. With values from 0 to 100.
Detail titled country. For a nominal scale. With 180 values from Costa Rica to Vanuatu.

A geographic map. percent_fully_vaccinated, country.
Legend titled percent_fully_vaccinated. For a quantitative scale. With values from 0 to 100.
1 of 10. Percent_fully_vaccinated is between 0 and 10. 6 values. Press t to open table.
2 of 10. Percent_fully_vaccinated is between 10 and 20. 12 values. Press t to open table.
3 of 10. Percent_fully_vaccinated is between 20 and 30. 11 values. Press t to open table.
4 of 10. Percent_fully_vaccinated is between 30 and 40. 19 values. Press t to open table.

A geographic map. percent_fully_vaccinated, country.
Legend titled percent_fully_vaccinated. For a quantitative scale. With values from 0 to 100.
1 of 10. Percent_fully_vaccinated is between 0 and 10. 6 values. Press t to open table.
2 of 10. Percent_fully_vaccinated is between 10 and 20. 12 values. Press t to open table.
3 of 10. Percent_fully_vaccinated is between 20 and 30. 11 values. Press t to open table.
4 of 10. Percent_fully_vaccinated is between 30 and 40. 19 values. Press t to open table.

Close Table View

Table View

percent_fully_vaccinated is between 10 and 20

percent_fully_vaccinated	country
13	Syria
12	Congo
11	Democratic Republic of Congo
18	Libya
15	Mali
15	Algeria
13	Cameroon
12	Gabon
17	Burkina Faso
19	Togo

Figure 2: An example of a user's keyboard traversal of the Olli Tree. Users can widen/narrow the scope of the text via the up/down arrow keys (respectively), and otherwise navigate between sibling nodes using left/right arrow keys. To access individual data, users can press the 't' key to view a snapshot data table

Query Type	Line Chart	Bar Chart	Scatterplot	Choropleth Map	Total
Analytical Query	147	168	242	193	750
Visual Query	42	28	54	41	165
Contextual Query	16	14	28	12	70
Navigation Query	12	12	12	12	48
Total	217	222	336	258	1033

Table 1: Distribution of data for four types of queries across different chart formats, including line chart, bar chart, scatterplot, and choropleth map, indicating the prevalence of analytical queries in the dataset.

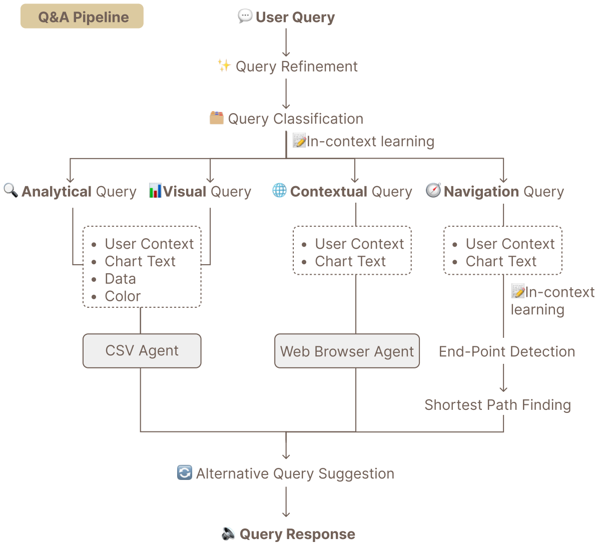


Figure 3: VizAbility pipeline takes a user query and refines it to improve clarity. The query is classified into one of four query types, each of which is fed to a different agent. If VizAbility fails to respond, it attempts to suggest alternative queries.

the ground truth responses for NAVIGATION queries consist solely of starting and ending nodes.

4.4.2 *Supported Query Types.*

ANALYTICAL queries—D2 primarily focus on understanding the underlying data, such as "Is Africa the country that needs the vaccine the most?" or "What is the highest positive anomaly?" **VISUAL queries**—D2 relate to visual encoding information or demand visual interpretation, exemplified by questions like "What color is North America?" or "Is the line fluctuating?" ANALYTICAL and VISUAL queries are not entirely distinct; VISUAL queries often necessitate data interpretation, as in "Which country exhibits the darkest shades for both the lowest and highest values?". However, we continue to differentiate these query types to communicate to LLMs that a query involves information encoded visually, which is specific to the problem of chart and question answering.

CONTEXTUAL questions—D3 seek information not directly present on the chart but require ancillary knowledge related to it. For instance, some questions aim to understand the chart's encoding, like "What is a scatterplot?" or "What does 'positive temperature anomaly' mean?" Others ask about context related to the data, such as "Where is Palestine?" or "Why does the data start in 1880? What occurred then?" Additionally, there are inquiries about the data's origin, exemplified by "What is the source of this information?" or "From where was this data obtained?"

Navigation queries—D4 are a category we introduced to enhance user experience. These queries are tailored to the synergy between keyboard navigation and conversational interaction. For instance, to reduce cumbersome keyboard navigation and assist users in orientation, questions such as "How can I get to the X-axis" (wayfinding) or "Where am I?" (orientation) can be beneficial. Our motivation for this stems from a previous empirical study [44], where blind individuals highlighted such challenges with Olli's tree view.

4.4.3 Query Classification. First, we aim to classify user queries based on this categorization rather than diving straight into responses. Once classified, we proceed to address each type of query in the subsequent phase (see the next section). This task division provides the LLM with a well-defined task and has been proven to increase its performance [69], while also enabling more efficient allocation of computational resources for each type of query. Figure 4 shows our few-shot prompting approach. In the prompt, we provide a clear definition for each query type. To bolster the definition, we accompany each with four exemplar questions.

These examples are sourced from our validation set, chosen based on their close alignment with the user query at query time. Specifically, for each query type and the given user query, we sift through the validation set to pinpoint the four most analogous queries. These are then incorporated as representative examples for each query definition within the prompt. For this endeavor, we used sentence transformers [56] to generate text embeddings and then applied cosine similarity to these embeddings to identify the most closely aligned examples. This method offers greater precision compared to arbitrarily selecting samples for each query type.

We constrain the range of LLM responses by explicitly instructing it to output either: "Analytical Query", "Visual Query", "Contextual Query", or "Navigation Query". To thwart any potential hallucinations from the LLM, we provide an accessible escape route by instructing the model to return "I am sorry. I am unable to answer this question" when confronted with a question that does not immediately conform to any of the specified query types. Without such a safeguard, GPT frequently generates technical jargon and error messages that can deter users.

4.4.4 Query-Specific Prompting. The answering pipeline diverges into three unique paths, depending on the query type (Figure 1).

Analytical & Visual Queries. D2—Figure 6 illustrates our approach to handling analytical and visual queries. To circumvent the predefined token limit of the LLM, we consolidate the transformed data extracted from the Vega View [7] into an external CSV file. This file is then processed by LangChain's CSV Agent [3], which operates in the background. Under the hood, this agent leverages the Pandas DataFrame agent, subsequently executing Python code generated by the LLM. We purposefully avoid including the entire raw dataset, recognizing that it might differ from the final view data. Often, the agent can get stuck in an infinite loop of thinking. To prevent this issue, we have implemented a time constraint. If the time limit is exceeded, VizAbility will display the message, "Answer: I'm sorry, but the process has been terminated because it took too long to arrive at an answer," and will also suggest alternative questions to the user (see Section 4.5).

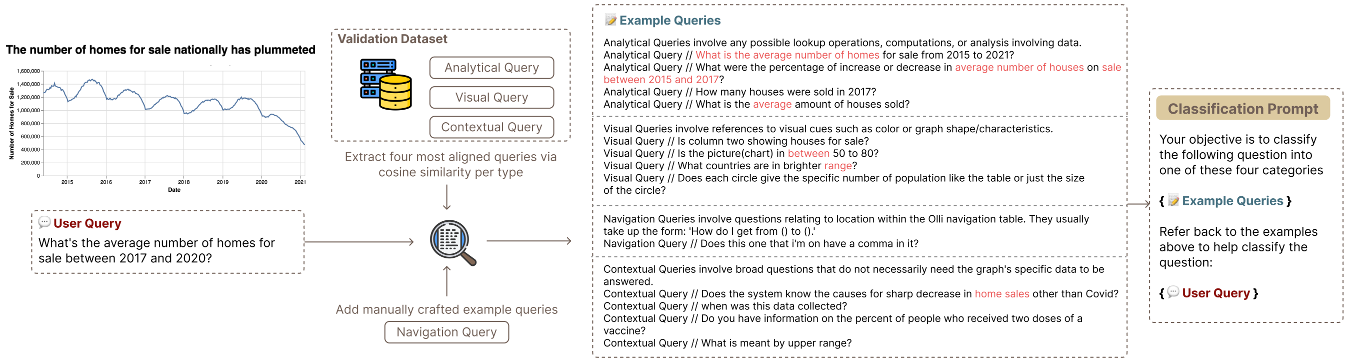
While the CSV agent can handle most data-related queries, it is not aware of any visual encoding information of the chart. To address visual queries, we extract color information directly from the Vega View [7] and incorporate it as an additional column within the CSV file. This modification ensures that each data point is paired with its corresponding color. Initially, the extracted color data is in hex codes. To enhance user-friendliness, we employ a color-matching algorithm to convert the hex codes into more common English names. This algorithm works by cross-referencing the source hex code with a predefined list of color hex codes and English names [2], ultimately determining the closest matching name based on their relatives distances within the CIELAB color space.

The color augmentation process enables answering visual questions like "What color is Algeria? What other countries are the color of Algeria?", as VizAbility responds: "Algeria is orange-red and other countries with the same color are Syria, Iraq, Congo, [...]." Furthermore, LLM is lenient with user queries and accepts a certain margin of error for color input. e.g., if the user asks about what *blue* represents, the system can infer *blue* refers to *steelblue* in the map.

To provide further visual context for the chart, we have integrated a textual representation of the chart generated by Olli directly into the LLM prompt (see Figure 6). This addition has the potential to significantly enhance the performance of visual question-answering. For example, when presented with the question "What does the graph show?", the system, without the text representation, provided a response like "The graph shows the data from the dataframe, which includes the year, value, temporal polarity, ...". However, when furnished with the text representation, the LLM responded with a more comprehensive and human-friendly answer: "The graph shows the temporal polarity of the temperature anomaly (in degrees Celsius) from 1850 to 2021 and the y-axis representing the temporal anomaly in degree Celsius. [...]"

Moreover, we supplement it with the user's current position within the tree view, tracked via the user's keyboard movements. This feature can help address potentially ambiguous questions. For instance, a user might ask, "What's an average?" with the intention of inquiring about the average within a category where their cursor is located. We also ensure that the responses are properly formatted with commas and special characters so that they are optimized for screen reader interpretation. For example, we present the number 468297 as 468,297 to improve clarity, especially since NVDA, the most popular screen reader [68], would otherwise read it as 'four six eight two nine seven' in its default settings, which could be less intuitive for users.

Contextual Queries. D3—To address contextual queries that require background or external information on what is available in the chart or its data, we have incorporated a Web Browser agent [4] to retrieve more general information relevant to chart comprehension. For example, when presented with the contextual query, "What do you mean by temperature anomalies," the LLM responds with, "Temperature anomalies are any measure of temperatures that are unusual for a particular region, season, or time period. [...]" Categorizing questions beforehand enabled us to streamline the process and eliminate unnecessary, resource-intensive prompts needed for analytical and visual queries.



User Query

What's the average number of homes for sale between 2017 and 2020?

Example Queries

Analytical Queries involve any possible lookup operations, computations, or analysis involving data.
Analytical Query // What is the average number of homes for sale from 2015 to 2021?
Analytical Query // What were the percentage of increase or decrease in average number of houses on sale between 2015 and 2017?
Analytical Query // How many houses were sold in 2017?
Analytical Query // What is the average amount of houses sold?

Visual Queries involve references to visual cues such as color or graph shape/characteristics.
Visual Query // Is column two showing houses for sale?
Visual Query // Is the picture(chart) in between 50 to 80?
Visual Query // What countries are in brighter range?
Visual Query // Does each circle give the specific number of population like the table or just the size of the circle?

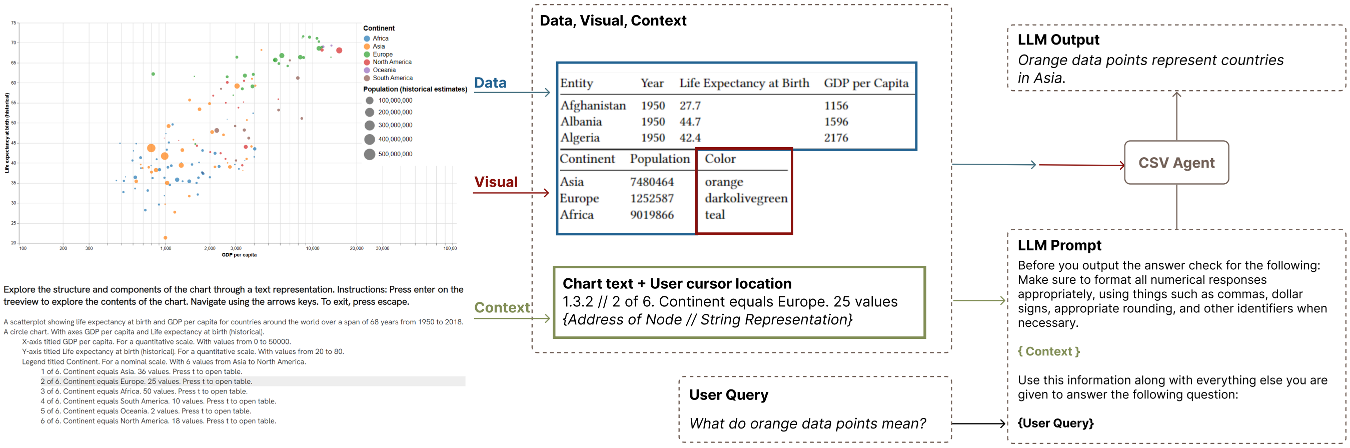
Navigation Queries involve questions relating to location within the Olli navigation table. They usually take up the form: 'How do I get from () to ().'
Navigation Query // Does this one that i'm on have a comma in it?

Contextual Queries involve broad questions that do not necessarily need the graph's specific data to be answered.
Contextual Query // Does the system know the causes for sharp decrease in home sales other than Covid?
Contextual Query // when was this data collected?
Contextual Query // Do you have information on the percent of people who received two doses of a vaccine?
Contextual Query // What is meant by upper range?

Your objective is to classify the following question into one of these four categories

Refer back to the examples above to help classify the question:

Figure 4: User questions are initially categorized based on query type via an LLM trained with few-shot prompting. We populate the prompt with sample questions and their corresponding ground truth classifications, which we extract from the validation set. Only those validation questions that share the highest cosine similarity score with the user query are selected within each query type.



Explore the structure and components of the chart through a text representation. Instructions: Press enter on the treeview to explore the contents of the chart. Navigate using the arrows keys. To exit, press escape.

A scatterplot showing life expectancy at birth and GDP per capita for countries around the world over a span of 68 years from 1950 to 2018.
A circle chart. With axes GDP per capita and Life expectancy at birth (historical).
X-axis titled GDP per capita. For a quantitative scale. With values from 0 to 50000.
Y-axis titled Life expectancy at birth (historical). For a quantitative scale. With values from 20 to 80.
Legend titled Continent. For a nominal scale. With 6 values from Asia to North America.
1 of 6. Continent equals Asia. 36 values. Press t to open table.
2 of 6. Continent equals Europe. 25 values. Press t to open table.
3 of 6. Continent equals Africa. 50 values. Press t to open table.
4 of 6. Continent equals South America. 10 values. Press t to open table.
5 of 6. Continent equals Oceania. 2 values. Press t to open table.
6 of 6. Continent equals North America. 18 values. Press t to open table.

Entity	Year	Life Expectancy at Birth	GDP per Capita
Afghanistan	1950	27.7	1156
Albania	1950	44.7	1596
Algeria	1950	42.4	2176

Continent	Population	Color
Asia	7480464	orange
Europe	1252587	darkolivegreen
Africa	9019866	teal

Chart text + User cursor location
1.3.2 // 2 of 6. Continent equals Europe. 25 values
{Address of Node // String Representation}

What do orange data points mean?

Orange data points represent countries in Asia.

Before you output the answer check for the following: Make sure to format all numerical responses appropriately, using things such as commas, dollar signs, appropriate rounding, and other identifiers when necessary.

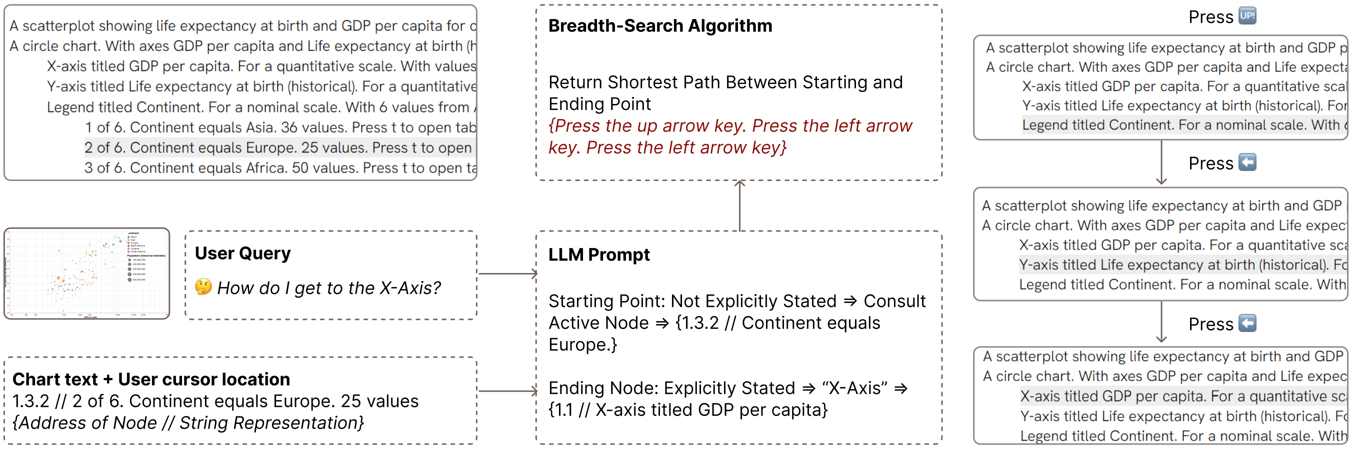
Use this information along with everything else you are given to answer the following question:

Figure 5: Query-specific evaluation for Analytical and Visual queries. We parse the chart's transformed data set and aggregate color encoding within a CSV file, which we then supply to an LLM via a CSV agent. For further context, we also populate the prompt with the user's active position within the Olli Tree, in addition to a text representation of the Tree itself.

Often, contextual queries require information about the chart or data to disambiguate user queries. For example, queries such as "Does the system know the causes for the sharp decrease in home sales other than Covid?" or "When was this data collected?" need to understand what the chart is about, without requiring detailed underlying data. Therefore, we accommodate these instances by incorporating the readily available high-level text representation of the chart into the Web Browser Agent.

Navigation Queries. D4—We seek to integrate users' keyboard navigation with the conversational module via navigation queries. VizAbility currently supports two types of navigation queries: (a) **wayfinding** questions, in which, upon being provided a starting and ending point within the tree view, the model returns a series of directions dictating the shortest traversal and (b) **orientation** questions, in which the VizAbility returns the user's current location within the tree view.

To handle navigation queries, we attribute a unique address to each node of the tree view and convey this address, along with the user's current position, to the LLM. Through the utilization of few-shot prompting, we instruct the LLM to discern the starting point and ending point from the user query. It is important that the model has significant leniency in user queries, as it is highly unlikely that the user will specify the exact starting/ending points verbatim. Thus, the few-shot prompting primes the LLM to properly interpret the user query. For example, in response to the query "Take me to Haiti" (related to the choropleth map), the LLM comprehends the user query's context and correctly deduces that the absence of an explicit starting node means the user intends to initiate navigation from their current location. On the other hand, VizAbility can easily infer the ending point, which is the node titled: "3 of 180. Country equals Haiti. 1 value. Press t to open table." If the model cannot discern any starting or ending point, it yields: "The question was interpreted as involving navigation, but either no starting/ending


A scatterplot showing life expectancy at birth and GDP per capita for c
A circle chart. With axes GDP per capita and Life expectancy at birth (h
X-axis titled GDP per capita. For a quantitative scale. With values
Y-axis titled Life expectancy at birth (historical). For a quantitative
Legend titled Continent. For a nominal scale. With 6 values from /
1 of 6. Continent equals Asia. 36 values. Press t to open tab
2 of 6. Continent equals Europe. 25 values. Press t to open
3 of 6. Continent equals Africa. 50 values. Press t to open ta

Breadth-Search Algorithm

Return Shortest Path Between Starting and Ending Point
{Press the up arrow key. Press the left arrow key. Press the left arrow key}

User Query

🤔 *How do I get to the X-Axis?*

Chart text + User cursor location

1.3.2 // 2 of 6. Continent equals Europe. 25 values
{Address of Node // String Representation}

LLM Prompt

Starting Point: Not Explicitly Stated ⇒ Consult Active Node ⇒ {1.3.2 // Continent equals Europe.}

Ending Node: Explicitly Stated ⇒ "X-Axis" ⇒ {1.1 // X-axis titled GDP per capita}

Press ⬆

A scatterplot showing life expectancy at birth and GDP p
A circle chart. With axes GDP per capita and Life expecta
X-axis titled GDP per capita. For a quantitative scal
Y-axis titled Life expectancy at birth (historical). For
Legend titled Continent. For a nominal scale. With 6

Press ⬅

A scatterplot showing life expectancy at birth and GDP
A circle chart. With axes GDP per capita and Life expec
X-axis titled GDP per capita. For a quantitative sca
Y-axis titled Life expectancy at birth (historical). Fo
Legend titled Continent. For a nominal scale. With

Press ⬅

A scatterplot showing life expectancy at birth and GDP
A circle chart. With axes GDP per capita and Life expec
X-axis titled GDP per capita. For a quantitative sc
Y-axis titled Life expectancy at birth (historical). Fo
Legend titled Continent. For a nominal scale. With

Figure 6: Query-specific evaluation for Navigation queries. We pass a text representation of the Olli Tree and the addresses of corresponding nodes within the Tree to an LLM alongside the user question. With the aid of few-shot prompting, the LLM then identifies the starting and ending nodes within the Olli Tree. Should the starting node not be explicitly mentioned within the question, the model instead utilizes the user's current location within the Tree. We then execute a breadth-search algorithm and relay the shortest path between starting and ending nodes back to the user.

point was provided, or the tree view was not activated. Please try again."

Once the starting and ending points have been identified, we employ a breadth-search algorithm that returns string instructions of the shortest path, which users can then manually follow at their own discretion. To avoid repetition we coalesce instructions whenever possible. For instance, for instructions pertaining to the navigation between two sibling nodes, we convert "Press the right arrow key. Press the right arrow key. Press the right arrow key." to "Press the right arrow key 3 times." We initially opted for this approach as opposed to automatically moving the user to their desired ending point with the rationale that autonomy and transparency are crucial for our intended audience.

4.5 Mitigating LLM Failures through Responsive Feedback

Several factors can create a disconnect between the user's query and the output from the LLM. These factors might be technical, such as when the CSV Agent executor is prematurely terminated for exceeding its time limit, or they might arise from the ambiguity of the user's query and the system's consequent struggle to categorize and address it. To tackle these challenges, we implement strategies that are both proactive (occurring before classification and answering) and reactive (occurring after these processes), aiming to create a more user-friendly environment.

4.5.1 Proactive Error Mitigation Strategies. D6—The versatility of VizAbility poses novel challenges in user query interpretation and answering. As reflected in the validation data set, some user queries may be too broad, ambiguous, or can even refer to variables that are not explicitly present in the data. For instance, the real user query "What kind of vaccine they used?" is irrelevant to the choropleth map displaying the share of the population that received at least one dose of the Covid-19 vaccine. User queries can also be poorly worded, such as "What parts of North America are not in the 80 to 100 percent range", deeming it more difficult for an LLM to accurately interpret and compute the answer. These irrelevant and ambiguous queries can significantly decrease the overall accuracy of the system if not accounted for.

To mitigate the effects that ambiguous queries may have on the system's overall accuracy, we introduce an additional **query refining process** that occurs before user query classification. We furnish a GPT-3.5-Turbo prompt template with the user query and—serving as the primary source of context—a text representation of the actively engaged chart. The LLM is instructed to add to—but not alter—the question so that it is as specific and relevant to the data as possible, whilst still retaining its original meaning. Referring back to the earlier example of an ambiguous query, the LLM refines it to: "Which countries in North America have a percentage of fully vaccinated individuals below 80%". The pipeline continues as detailed in prior sections, but now utilizes the refined query instead of the raw user query. Whereas before VizAbility could not accurately answer the above question and instead outputted a list of all of the countries represented in the data, the system is now able to consistently identify all countries that occupy the desired range.

4.5.2 Reactive Error Mitigation Strategies. D6—LangChain's CSV Agent leverages chain of thought reasoning administered over a series of sequential prompts (in conjunction with pandas dataframe) to answer questions pertaining to a CSV file. Each individual prompt must therefore follow a specified format of Observation, Action, and Action Input for a subsequent prompt to properly parse the result [3]. We have observed that sometimes an individual prompt in this sequential chain may be outputted in an incorrect format, thus triggering a cascading effect that ultimately results in an OutputParserException. To mitigate this error, we utilize prompt engineering and direct the CSV Agent Executor "You must follow the structure of Observation, Thought, Action, Action Input, etc." to ensure that the output format is consistent between sequential queries. If the CSV Agent Executor is active for an extended period of time, we manually terminate it to preserve time-efficiency.

We handle these two potential errors by being explicit in our language; e.g., "I am sorry but I cannot answer the question". To foster an iterative and exploratory environment, we also employ a pipeline that recommends two questions that retain the essence of the original user query but eliminate error-inducing language. We achieve this recommendation by monitoring the response of the answering pipeline. If the output exhibits any sign of error (refer back to the above cases) or is otherwise incomplete, i.e. "No answer can be found", we populate a prompt template with the original user query, the error output (which identifies the error), and a text representation of the chart for additional context. This prompt is then passed through the LLM, which generates two new questions that are subsequently relayed back to the user in the form of clickable buttons. For instance, the error-inducing question: "Where are these houses sold?" yields the following LLM output, "The data does not contain any information about the location of the houses." and is accompanied by two rephrased queries: "What information regarding the sale of these homes is provided in the dataset beyond the date and inventory quantities?" and "Could you elaborate on any additional details related to the properties or their characteristics available within the dataset?", each displayed as a button.

4.6 Fostering a More Accessible User Experience

Providing different query methods and audible cues. D7—Users can submit conversational queries via voice recordings that are processed via the Whisper speech recognition [8]. However, oftentimes, enabling microphones can be problematic. Thus, we provide an alternative text box so that they can type the queries using the keyboard. Upon inputting their question (regardless of the modality), users are provided with an audible cue of "Loading. Please Wait". Every subsequent 3 seconds, the user is exposed to yet another audible cue, this time "Still Loading". This loading cue significantly improves transparency and mitigates any possible confusion that can arise from an unresponsive webpage.

Enhancing user trust and transparency in responses. D7—VizAbility does not solely display the answer, and instead provides the user query and brief justification behind its response in conjunction with the actual answer. For instance, the following is articulated by VizAbility when a user asks, "What is a choropleth map?": "Your question 'What is a choropleth map?' was categorized as being context-seeking, and as such, has been answered based on information found on the web." By letting users know the scope of the answer (i.e., whether it was sourced from the internet, data, or the tree view), we allow users to verify and evaluate the effectiveness of the LLM response independently, thus bolstering user trust and system transparency.

Offering cold-start query suggestions for onboarding. D7—To help users figure out what queries are possible, VizAbility generates four initial queries, each of which belongs to one of the four query types. We achieve this suggestion by querying the LLM with a prompt that uses in-context impersonation [58] ("Pretend you are a blind/low vision user who is presented with a chart."), coupled with a text representation of the current chart. For instance, the LLM generates the following questions for the bar chart: "What is the temperature anomaly for the year 2020?"; "Can you provide a description of the color scheme used in the bar chart to represent the temperature anomalies?"; "Are there any patterns or relationships between the year and the temporal polarity of the temperature anomaly?"; "How do I get from my current position in the text representation to the x-axis?" These questions appear as interactive buttons, allowing users to either choose a suggested question or input their own.

Query Type	Precision	Recall	F1	Count
Analytical	90.96%	93.10%	92.02%	551
Contextual	64.65%	67.37%	65.98%	95
Navigation	100%	97.50%	98.73%	40
Visual	89.09%	74.81%	81.33%	131

Table 2: Quantitative results contextualize VizAbility's classification accuracy of 87.39% through the lenses of Precision, Recall, and F1 scores. The system is most proficient at classifying analytical and navigation queries, as indicated by the significantly higher F1 Scores attributed to these query types.

4.7 Notes on Interactive Charts

VizAbility extends its capabilities to interactive charts, allowing for dynamic updates in both the tree view and question-answer components when users modify the chart. An example of this update is seen in the scatter plot (referenced in Figure 6), which includes a slider for filtering data by year. As a user selects a specific year, VizAbility generates a new view of the data for the LLM and simultaneously updates the tree view to reflect the chosen year.

5 EVALUATION: Q&A PERFORMANCE BENCHMARK

For our quantitative evaluation, we concentrated on validating the question-answering pipeline using the testing dataset. This evaluation comprised two components: assessing the accuracy of query classification and evaluating the correctness of question responses.

5.1 Classification Evaluation

Our evaluation yielded an overall classification accuracy of 87.39%. Table 2 presents detailed results, including precision, recall, and F1-scores for each class. Table 3 provides examples of misclassified instances.

The model was effective in identifying analytical queries, with a 93.10% ($\frac{513}{550}$) recall and 90.96% ($\frac{513}{564}$) precision. Analytical queries often contain distinct computationally-heavy language, such as "correlation" and "average decrease," as seen in examples like "What's the correlation between GDP per capita and population?" and "For the time from September 2019 to September 2020, what was the average decrease in homes for sale?" This specific language may

True Class	Predicted Class	Example Text
Analytical	Contextual	Does the data include booster shot?
Analytical	Visual	Is Asia in the medium upper range in 60 to 80 area?
Contextual	Analytical	What is the percentage at which vaccinated population reach herd immunity?
Contextual	Analytical	What is the source of this data?
Contextual	Visual	Describe a scatterplot.
Contextual	Visual	Can I get a map of the US?
Visual	Analytical	How many years in total are represented on the X axis?
Visual	Analytical	Which part of Asia has a darker shade of green...
Visual	Analytical	What is the trend for Brown?
Visual	Contextual	Is there anything specific you want me to look for in this chart?
Navigation	Visual	...am I closer to the start or the end of the Y-axis?

Table 3: Additional Examples of Misclassification Cases

aid in differentiating these queries from broader contextual ones, potentially leading to a higher overall accuracy.

Of the 95 queries designated as contextual in the ground truth, 67.37% (64) were classified as such by VizAbility. In addition to the 64 true positives, the model incorrectly classified 30 analytical queries and 5 visual queries as contextual in nature - contributing to a precision of 64.65% ($\frac{64}{99}$). Poorly classified contextual queries like "What is the source of this data?" and "Can I get a map of the US?" may indicate that the model often emphasized individual words over the overall meaning of queries. For instance, the word "data" likely led the model to classify the first query as analytical. Similarly, the word "map" likely prompted the model to mistakenly identify the second query as visual.

A recall of 74.81% or $\frac{98}{131}$ for visual queries indicates that VizAbility correctly classified around $\frac{3}{4}$ of the queries designated as visual by the ground truth. On the contrary, visual queries such as "Which part of Asia has a darker shade of green, which has the most vaccinated amount of people?" and "What is the trend for Brown?" were falsely identified as being analytical, despite the presence of visual language {"Brown", "darker shade", "green"}. This lower performance may be due to the significant overlap between visual and analytical queries. For example, we consider any question that references visual components of the data ("Brown") as visual, even if it also involves computations ("trend"). On the other hand, the model rarely misclassified analytical or contextual queries as visual, as demonstrated by an 89.91% ($\frac{98}{109}$) precision score.

VizAbility's proficiency in distinguishing navigation queries may be attributed to the distinct and uniform structure to which most GPT4-generated example queries conform; i.e., queries either assume the role of wayfinding or orientation questions. The results, showing a 100% ($\frac{39}{39}$) precision and 97.50% ($\frac{39}{40}$) recall, suggest that VizAbility is effective in distinguishing these tasks from the typical data retrieval or lookup associated with analytical and visual queries. Nonetheless, the misclassified query "While exploring inventory values between 800000 and 1000000, am I closer to the start or the end of the Y-axis?" hints that the model might sometimes give undue weight to individual words. For example, the reference to 'the Y-axis', a visual element of the chart, could have led the model to categorize the navigation query as visual.

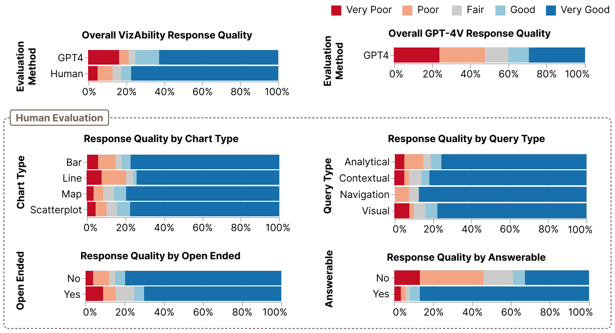


Figure 7: Quantitative results display the distributions of quality ratings (via a 5-point likert scale) for VizAbility responses. For more granularity, the results are also partitioned by query type, chart type, and question characteristics. VizAbility's performance is considerably higher than the GPT-4V baseline.

5.2 Question Response Evaluation

Our evaluation of the response quality is twofold. First, two researchers manually inspected the quality of the responses. Based on this ground-truth evaluation, we established an LLM-based evaluation using GPT-4, which is intended to facilitate scalable, automatic evaluation for efficient benchmarking tests, following recent literature in natural language evaluation [27, 49, 50, 66]. We describe each of these procedures below.

5.2.1 Human Evaluation. We evaluated each pair of system response and corresponding ground truth individually based on a five-point Likert scale: [Very Poor, Poor, Fair, Good, Very Good]. Our focus was on a response's 'correctness' in terms of its coherence and consistency with the ground truth. Our assessment scheme favored explanatory responses over overly brief ones, in line with our recognition of trustworthiness and transparency as essential factors. Table 4 shows how we defined each increment on the Likert Scale, which is further elaborated in the GPT4-based assessment in Section 5.2.2.

Of the 817 user queries, 632 or 77.36% were deemed "Very Good" in the human assessment. Responses rated as "Very Good" often restated the user's question, formatted quantitative data correctly, and

Very Good	Response B is not only similar but also faithful to Response A.
Good	Response B is mostly similar to Response A but may lack some of the more specific key details such as labels.
Fair	It is unclear whether there are any similarities between Response A and Response B due to the ambiguity of one or two of the Responses.
Poor	The content of Response B is somewhat irrelevant to that of Response A; the core information in Response B does not match that of Response A.
Very Poor	The content of Response B is irrelevant to that of Response A and bears no similarities.

Table 4: A delineation of the gradations used in the Likert scale to assess the degree of coherence between Response A and Response B, ranging from ’Very Good’ to ’Very Poor’.

included contextual information. For example, in response to the query “What continent has the highest vaccination rate?” related to a choropleth map, VizAbility answered, “The continent with the highest vaccination rate based on the percentage of fully vaccinated people in each country is South America, with an average percentage of 69.7%.” This response, more detailed than the ground truth “The continent with the highest vaccination rate is: South America”, demonstrates VizAbility’s ability to provide comprehensive answers. The distribution for Good, Fair, and Poor responses was 5.14% or $\frac{42}{817}$, 4.65% or $\frac{38}{817}$, and 7.71% or $\frac{63}{817}$ respectively. The human assessment yielded 42 or 5.14% of questions as being ”Very Poor” in coherence to the ground truth. We elaborate on these findings and investigate relationships between the types of user queries and VizAbility’ responses below (Figure 7).

Partitioning based on query classification. VizAbility exhibited the greatest accuracy in answering navigation questions. 87.5% ($\frac{35}{40}$) of navigation queries received a “Very Good” assessment, which equates to the correct translation of the user query into concrete starting and end nodes within the treeview. For instance, VizAbility correctly parsed the question “What’s the quickest path to get from the top of the tree to inventory values above 1400000?” and identified “Starting Point: ‘A line chart. With axes Date and Number of Homes for Sale’; Starting Address: 1; Ending Point: ‘Inventory is between 1400000 and 1600000’; Ending Address: 1.2.6”, which corresponds with the ground truth. User references to the treeview may not always be explicit (“top of the tree” → “a line chart...”). 82.11% ($\frac{78}{95}$) of responses to contextual queries were identified as being “Very Good”. This metric was slightly lower for both visual and analytical queries, from which 77.86% ($\frac{102}{131}$) and 75.68% ($\frac{417}{551}$) of responses had warranted a “Very Good” assessment respectively. Nevertheless, the distribution of assessment scores is relatively consistent across query types, with a tendency to skew towards “Very Good” (Figure 7).

Partitioning based on chart types. The distribution of assessment scores is similar across chart stimuli, suggesting that chart type may not have as significant an influence on overall system performance compared to other factors. Questions pertaining to the choropleth map had the highest frequency of yielding “Very Good” responses (79.61% or $\frac{164}{206}$). This is followed by responses for the scatter plot (77.65% or $\frac{205}{264}$), bar graph (77.27% or $\frac{136}{176}$), and line chart (74.27% or $\frac{127}{171}$). The low variability amongst chart stimuli highlights VizAbility’s versatility in addressing a wide range of data visualizations.

Partitioning on the answerability of queries. We found that of all the questions for which VizAbility’s response was prescribed a “Very Poor” rating, almost half (45.24% or $\frac{19}{42}$) had also been identified as being unanswerable (Figure 7). More notably, 76% ($\frac{48}{63}$) of “Poor” responses also happened to be unanswerable. For instance, referring to the choropleth map which depicts the percentage of the global population receiving at least one dose of the vaccine, the query “Which country has the fewest vaccinations?” is unanswerable because of the absence of precise counts of vaccines from the data set. Nonetheless, VizAbility fails to acknowledge that relative percentages do not equate to precise counts and generates the “Very Poor” output “Among the listed countries, Burundi has the fewest vaccinations.”. Instances like these skew the overall accuracy of the system towards “Very Poor”. By isolating just the answerable queries and recomputing accuracies, we find that 87.03% ($\frac{584}{671}$) of responses are now “Very Good” and consistent with the ground truth, increasing the mean assessment score from 4.42 to 4.70.

Partitioning based on the open-endedness of queries. 70.10% ($\frac{143}{204}$) of system responses to open-ended queries were prescribed a score of “Very Good”. For the assessment of open-ended contextual queries, we emphasized VizAbility’s ability to not only draw on external information, but also support claims with relevant data derived from the chart. The contextual query, “Is there any other places that I can get some new information like a graph which shows by each region of the US?” with ground truth: “Yes, charts depicting inventory of houses for sale by region can be found on the internet.”, received the following “Very Good” response:

> “Yes, you can find a similar graph that displays home sales data by different regions of the US. The existing dataset provides a line chart showing the number of homes for sale over time, and this type of data is often collected and analyzed at both the national and regional levels. *Organizations such as the National Association of Realtors, Zillow, and Redfin* regularly publish reports [...]”

In this response, VizAbility not only explicitly identifies credible sources that the user can consult, but also clarifies those sources’ relevance to the line chart. Nonetheless, VizAbility proved to be more proficient in answering questions that were not open-ended. Of the 613 user questions that are narrower in scope (not open-ended), 79.77% yielded "Very Good" system responses - a value almost 10% higher than the reported accuracy for open-ended queries (Figure 7). This performance increase may be due to the narrow scope of non-open-ended questions, which could reduce the chances of computational or logical errors; e.g., scatter plot questions such as "What color is North America?" are straightforward. Consequently, VizAbility's response, "The color that represents North America in the dataset is red." is almost verbatim to the ground truth, "The color of North America on the scatter plot is: Red". Similarly, responses to non-open-ended queries were also less frequently assessed as "Very Poor"; 3.92% vs. the rate of 8.82% for open-ended queries.

5.2.2 Automatic Evaluation using GPT4. We leverage LLMs for holistic evaluations that align better with human judgments compared to traditional metrics like BLEU, ROUGE, and BERTScore, which focus on simple text-level differences [49].

Our evaluation prompt, inspired by Liu et al. [50], presented two responses to GPT4 (gpt-4-0125-preview): Response A and Response B, with Response A acting as the ground truth. GPT4 was directed to assess the coherence of Response B in relation to Response A. We refrained from revealing which response was the ground truth or our own creation, as we hypothesized this indication to be extraneous information for the LLM, given its task to solely assess the coherence of two responses. The lower Kendall τ score (= 0.2900) with human ground-truths for a similar prompt during our iterative testing, which explicitly defined Response A as the ground truth, supported this decision.

Given that the coherence metric serves as an umbrella term for response evaluation, encompassing related aspects such as correctness, phrasing, and verbosity, we further prime the LLM through few-shot prompting with references. We populate the prompt with a manually selected sample of questions, VizAbility responses, and their corresponding ground truth that we derive from the validation set, along with example scorings. To further contextualize the LLM's evaluation responses, we direct it to append a one-sentence rationale to its Likert score. During prompting, if the score deviated from the 1-5 range, GPT4 reassessed its evaluation of Response B. The results were formatted as `Score: {coherence score} Rationale: {rationale behind coherence score}`.

Our automatic evaluation prompt yielded a Kendall's τ score of 0.5526 ($p < 0.001$), signifying a strong correlation ($|\tau| \in [0.3, 1.0]$) between the automatic assessments and human evaluation [17]. The exact distribution of score ratings for GPT4 assessments is as follows: "Very Poor": 16.52% or $\frac{132}{817}$, "Poor": 5.02% or $\frac{41}{817}$, "Fair": 3.18% or $\frac{26}{817}$, "Good": 12.85% or $\frac{105}{817}$, "Very Good": 62.42% or $\frac{510}{817}$. GPT4 demonstrated more variability in its assessments, with fewer outputs clustered around the "Very Good" rating. More precisely, we observed a reduction of 122 "Very Good" assessments and an increase of 63 "Good" assessments between the human and LLM evaluations, which might signify that GPT4 had adopted stricter and more rigourous criteria. For the choropleth map query, "What is the highest vaccination rate in Africa?", which yielded the system response, "The country with the highest vaccination rate in Africa based on the geographic map dataset is Rwanda.", the human evaluation designated this response as "Very Good" in overall quality due to its clarity, explanatory language, and close correspondence with the ground truth: "The highest vaccination rate in Africa is 78.00% and belongs to Rwanda.". Nonetheless, GPT4 attributed a score of "Good", citing "Both Response A and Response B identify Rwanda as the country with the highest vaccination rate in Africa, although Response B does not provide the specific percentage rate."

5.3 Baseline Comparisons

5.3.1 Comparison to a question-answering system with symbolic inputs. We compared our system to a similar system that focuses on chart reasoning. [42]. Directly running the system proved challenging due to compatibility issues with outdated dependencies. A similar evaluation, using our dataset from blind participants, was conducted in prior work, revealing an overall factual accuracy rate of 16% [43]. This evaluation, however, was limited to just 245 queries from the total dataset. It excluded CONTEXTUAL queries and others deemed unanswerable due to ambiguous wording. Furthermore, the evaluation focused solely on questions related to bar and line charts, aligning with the system's supported question types.

Overall, the system demonstrated some level of proficiency in answering straightforward value retrieval and extrema questions. The relatively low performance was mainly attributed to query comprehension and handling of diverse task types (e.g., yes/no and range questions). Seeking to maintain consistency with the prior system, we extracted data solely from the bar and line charts for a more fitting comparison. When narrowing the scope to these two types of visual stimuli, VizAbility reports $\approx$ 76.01% accuracy for the line chart and $\approx$ 81.82% for the bar chart, only considering 'Very Good' and 'Good' responses, signifying a significant improvement in user query handling.

5.3.2 Comparison to GPT-4V(ision) with image inputs. We conducted a comparative analysis between VizAbility and GPT-4 with Vision. Supplying GPT-4V with input images of the line chart, bar graph, scatter plot, and choropleth map, we generated responses to all 777 analytical, contextual, and visual queries in the test set. We assessed the quality of GPT-4V responses using our GPT4-based automatic evaluation pipeline, which has been verified to correlate strongly with human evaluations. We omitted navigation queries because it is infeasible to provide just one reference image for the keyboard-navigable treeview. On the other hand, supplying the treeview text is essentially equivalent to using VizAbility.

The GPT4 automatic evaluation yielded the following distribution of scores: 29.86% or $\frac{232}{777}$ of responses received a "Very Good" assessment. 10.55% or $\frac{82}{777}$ of responses were "Good", whereas 12.23% or $\frac{95}{777}$ were "Fair". "Poor" and "Very Poor" responses occurred at equal frequencies: 23.68% or $\frac{184}{777}$. This discrepancy in performance between GPT-4V and VizAbility, which outputs "Very Good" responses at a frequency of 77.36%, can be attributed to GPT-4V lacking access to the raw data required for queries that require symbolic computations, resulting in inconclusive responses for certain prompts, such as "What is the average vaccination rate for countries in the world?":

> "The image you've provided is a world map that shows the share of the population receiving at least one dose of a COVID-19 vaccine... To determine the average vaccination rate for countries around the world, one would need to access the raw data from a reliable

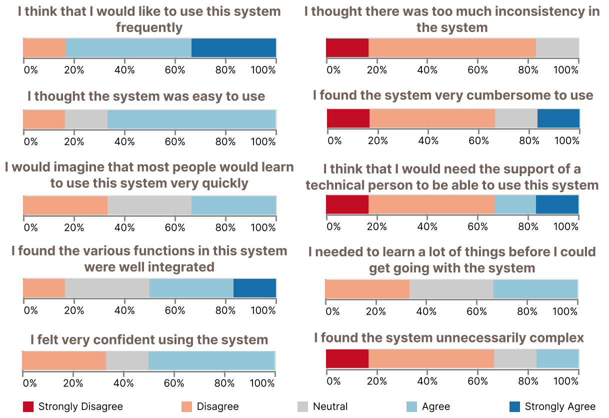


Figure 8: System Usability Scale Survey Results, showing participant responses on a Likert scale from 'Strongly Disagree' to 'Strongly Agree' for various statements about the system's frequency of use, ease of use, learning curve, integration of functions, consistency, need for technical support, and complexity, as part of the qualitative study.

> source such as the World Health Organization, a government health department, or a global health tracking website...".

Furthermore, the responses outputted by GPT-4V tend to be excessively verbose. For instance, in the above response, GPT-4V initially strays away from the user question by providing an unnecessary and potentially redundant description of the choropleth map. On average, GPT-4V responses were longer by 97.69 words, which can detract from user readability and concision. Yet, GPT-4V often performed reasonably well on certain visual questions such as asking overall shapes of trends (e.g., "How is the graph moving?" "Would it be [...] going up like hills and valleys type of trend?").

6 EVALUATION: USER STUDY WITH BLIND PEOPLE

During the development process, we engaged with a blind participant who had prior experience using a screen reader on a daily basis. This participant provided feedback at two intermediate stages of development in what was an iterative design process. In addition to this intermediate prototype evaluation, we conducted a preliminary usability study with six additional blind/low-vision individuals. IRB approval was obtained before the commencement of the study. The stimuli used in the study was an early version of VizAbility. While the major features remain unchanged, the improvements made after the study are described in Section 6.6.

6.1 Participants

We recruited six blind/low-vision individuals from the National Institute of the Blind [5]. Their demographics are shown in Table 5. We tried to recruit diverse participants based on their gender and screen reader expertise. Our participants comprise three females and three males. Their ages were distributed as follows: two participants aged 25-34, two aged 45-54, one aged 55-64, and one aged 65 or older. Two participants have been blind since birth, while the other four experienced blindness onset later in life. Regarding their proficiency, all participants possessed at least intermediate experience, with three classified as advanced and two as experts. In terms of assistive technology, four participants primarily used JAWS as their screen reader, while the remaining two utilized VoiceOver and NVDA, respectively.

6.2 Procedure

The hour-long experiment was conducted over Zoom, and moderated by the first author. Upon entering the session, participants opened up our system in a web browser and chose a chart of their choice among the four options: line chart, bar chart, scatterplot, or choropleth map (see Table 5 for participants' choice of charts). To mimic a real-world encounter with VizAbility, we refrained from giving any contextual information about each chart, in order not to introduce any bias into the participant's selection.

The study was divided into three parts. The first two - spanning roughly 20 minutes each - focused on the individual components of our multimodal approach—the keyboard-navigable tree view and the conversational module. The data table was not included in the study. In the assessment of each component, participants had 5-10 minutes to explore it freely, noting strengths and weaknesses. Subsequently, the component's function was explained, followed by a 10-minute guided exploration led by the moderator.

To conclude, each participant was asked a series of component-specific questions. For the keyboard-navigable tree view, these included: "Describe the chart to the best of your capabilities", "How easy was it to navigate this interface?", "How useful is this tool?" For the conversational module, questions were: "Describe the chart to the best of your capabilities", "Assign a ranking for each question type based on their usefulness.", "How easy was it to use this interface?", "How useful is this tool?"

The final part centered on the components' combined functionality to assess the potential advantages of their collaborative operation. After having been exposed to the entire system, participants were asked to reassess VizAbility based on ease of use and functionality. We maintained a consistent order for the parts without randomization. We conducted a brief post-task survey, which we distributed to each participant immediately after their completion of the study, to learn about the participants' overall experience with VizAbility. Participants completed their surveys within 24 hours of participating in the study, allowing sufficient time for reflection whilst ensuring that VizAbility remained fresh in their minds.

6.3 Analysis Process

Our analysis process involved a qualitative method, considering the study's small scale. During each session, the session moderator took detailed notes, capturing key observations and participant responses in real-time. Post-session, they reviewed the recordings of the interviews and interactions. This step involved careful, repeated listening to the audio to extract in-depth insights and to cross-reference them with the notes taken during the sessions as well as usability survey results. The analysis focused on identifying recurring themes, patterns of user behavior, and specific instances of user-system interaction challenges or successes. The analytical

PID	Gender	Age	Vision Level	Screen Reader Expertise	Screen Reader	Chart Selected
P1	Male	45-54	Blind with later onset	Expert	JAWS	Bar Chart
P2	Female	65 or older	Blind since birth	Advanced	VoiceOver	Line Chart
P3	Female	25-34	Blind with later onset	Intermediate	JAWS	Choropleth Map
P4	Female	25-34	Blind since birth	Advanced	JAWS	Scatterplot
P5	Male	45-54	Blind with later onset	Expert	JAWS	Bar Chart
P6	Male	55-64	Blind with later onset	Advanced	NVDA	Choropleth Map

Table 5: Distribution of Participant Information, detailing gender, age range, level of vision impairment, screen reader expertise, preferred screen reader technology, and the type of chart selected for the study.

process was iterative, with findings from the initial sessions informing subsequent reviews. Analysis results were discussed with a senior author to validate interpretations and ensure a diverse perspective in the analysis.

6.4 Behavioral Observations

Here, we detail participants' actions and feedback while using VizAbility during the study sessions.

6.4.1 Navigating the tree view. Participants were able to utilize the tree view using arrow keys and tab shortcuts as reported in prior studies [44, 72], although the learning curve proved to be slightly steeper for P2 and P5. P5 remarked on the "cumbersome" structure of the tree for the bar chart, noting that it was due to the presence of over 170 unique data values. Rather than tediously navigating through the data using the down arrow key, P5 wished for a more efficient method to move between specific nodes within the tree view. P2 echoed this sentiment, highlighting the risk of disorientation, particularly with larger and more intricate data sets.

Several participants (P1, P3, P4, P5, P6) independently recognized the distinctive structure of the tree view, which presents a data set through visual encoding variables. For example, P5, after navigating a choropleth map and expressing frustration over manually sifting through 172 countries without an apparent order, was pleasantly surprised when using the right arrow key led him to the same data set, this time organized by vaccination rates in 10 percent increments. This participant then confirmed that the tree view was more effective in conveying a visualization's structure compared to a traditional data table.

After having used their keyboard to navigate through the tree view, participants were asked to describe the visual stimuli to the best of their capabilities. Responses were mixed, with two participants (P3 and P4) only being able to identify the variables represented by each axis (country v. percent of population vaccinated, and average life expectancy v. GDP per capita, respectively). This result suggests that despite being a good overall indicator of chart structure, the tree view alone is not sufficient for complete data visualization. The result was reaffirmed by the usefulness rating most individuals attributed to the system, with the average hovering around a 3 out of 5.

6.4.2 Exploring the conversational module. Although 4 Participants (P1, P2, P3, P5) gravitated towards the text input modality, all affirmed the importance of retaining an option for voice input as well. All but one participant (P1, P2, P3, P4, P5) immediately asked data-driven questions (either simple fetches for data, like "What is the vaccination percentage for Haiti" or more complex queries involving multiple steps), with P6 instead asking a contextual question: "Is there a way to rank the various countries into continents?" (regarding the choropleth map). This coincided with subsequent participant ratings for the usefulness of the four query types, with all users asserting ANALYTICAL queries as the most useful for chart comprehension. Most users (P1, P2, P3, P5) could not fathom the possibility that more broad questions were supported.

Following this independent exploration of the conversational model, participants were made aware of the four distinct types of queries and were once again directed to input their own questions; however, this time around, they had to broadly adhere to one of the four query classifications. Users demonstrated a greater proficiency with the conversational module during this guided exploration, with P1 even chaining multiple individual queries to arrive at a broader understanding of the chart. By consecutively asking "What is the temperature for 2020?" and "What color is 2020?", the participant was able to deduce that the color 'red' indicates positive temperature values.

We also observed an affinity for contextual queries among the participant pool. One user (P4) who had little to no experience with map visualizations prior to the study asked: "What is a choropleth map?", to which the LLM outputted a correct response. However, when the same participant asked, "What is a temporal polarity" (pertaining to the bar chart), the LLM responded with a definition tied to linguistics. Although initially taken aback, the user acknowledged the possible ambiguities with the word "temporal polarity" (which has multiple meanings), and upon rephrasing her query to incorporate more precision, received a more accurate response. The participant attributed her realization to the VizAbility's justification (outputted alongside the response), which explicitly told her that it sourced its answer from the internet.

6.4.3 Integrating the two components. Participants were then introduced to navigation queries. We explained the purpose of these queries, emphasizing their role in wayfinding and orientation, and then allowed them to formulate their own navigation queries. All users concurred that these queries were essential for understanding the tree view (P6: "Maybe not as much as for smaller datasets, but I definitely see their use for complex data like this"), a sentiment echoed in the usefulness ratings they assigned to the integrated system. While previous Likert scale ratings for the individual components averaged around 3, after this introduction, participants consistently rated the complete system between 4 and 5, with 5 being extremely useful.

Most participants tended to input short and concise navigation queries. Rather than inputting "How do I get from my current location to the percentage vaccinated value for Guam", one user (P5) opted for the much simpler "Take me to Guam". Showcasing its conversational strengths, our model was able to precisely identify the starting as well as ending nodes from this colloquial text, yielding the instructions: "Press the right arrow key. Press the down arrow key two times."

6.5 User Feedback and Reflection

Participants completed a post-study questionnaire based on the System Usability Scale (see Figure 8). Notably, most participants (4 Agree; 1 Strongly Agree; 1 Disagree) concurred with the statement: "I found the various functions in this system were well integrated." Results can be found in Figure 8. Participants also valued VizAbility's commitment to accessibility and transparency, especially within the conversational module. They envisioned real-world applications for VizAbility, relating it to their personal experiences. For instance, P1 saw its potential in providing testing accommodations for GRE exams, noting its superiority over human proctors in translating visual graphs. P6, who teaches the NVDA screen reader to the BLV community, expressed interest in incorporating the system into his lessons. However, there was also constructive feedback.

Although most participants deemed the structure of navigation query responses (a sequence of directions) to be satisfactory, P2 advised that the system should automatically transport the user's cursor to the desired location, as opposed to currently requiring the user to manually traverse the tree view themselves. One participant (P5) sought more control over the nature of LLM responses outputted by the conversational model. He brought up the necessity of having some implementation of a dial to regulate the verboseness of the outputted answers. The same user who commented on the cumbersome structure of the tree view (P5) further elaborated that he would prefer a more concise raw data table in its place, especially for less extensive datasets. Apropos implementing a raw data table, P5 remarked: "I would prefer a simple table. I'm used to it. I know how to do it. Not all blind people do, but I do".

6.6 Changes After User Feedback

Here, we briefly describe how we incorporated the lessons learned from the user study.

Injecting chart information into contextual queries. D3 Participant 4's experience revealed a limitation in our initial contextual query handling: VizAbility often misinterpreted user intent due to missing chart information. For instance, it treated "temporal polarity" as a linguistic term, not a data dimension depicted in the chart. We addressed this by enriching prompts with chart tree view text (Figure 3), which led to correct interpretation of the data (e.g., "temporal polarity" in global temperature data). This observation also led to exploring query ambiguity in general, resulting in pre-processing question refinement (Section 4.5).

Automating navigation to end-points in the tree view. D4 Participant 2's preference for automatic traversal suggested that manually pressing a series of keys can be cumbersome, especially for lengthy navigation paths. Originally, we aimed to ensure transparency and grant users control over the process. However, recognizing this issue, we decided to introduce an option for automatic traversal (see Section 4.4.4). While manual navigation might become tedious with familiarity, we kept it as a default option to accommodate varying technical proficiencies.

Providing a raw data table. D5 To address Participant 5's concerns regarding the complexity of navigating the tree view, we introduced a conventional raw data table as an alternative (see 4.3). Though this addition may not significantly contribute to the novelty of our work, it underscores our dedication to creating an inclusive system.

Providing query suggestions. D6 The initial self-guided exploration of charts showed that most participants (P1, P2, P3, P5) struggled to ask questions beyond data-retrieval queries. Despite recognizing the value of visual, contextual, and navigation queries, participants were unaware of these query types until they were explicitly explained by the moderator. This observation, along with P2's suggestion for help documentation and preference for interactive guidance, led to the addition of query suggestions to the system (see Section 4.5 and Section 4.6).

7 DISCUSSION & FUTURE WORK

Our evaluation studies underscore the potential of VizAbility and also pinpoint areas for enhancement. We reflect on the limitations and challenges, paving the way for future opportunities.

7.1 Limitations and Opportunities

Customizing verbosity levels. Despite our initial aim to offer concise and informative answers, P5's recommendation for user-adjustable response verbosity underscored the importance of user agency over designer-imposed settings. Given that speech is processed serially, the text length read by screen readers becomes a pivotal design consideration. This concern has been reiterated in prior research [9, 13, 37, 72]. Similarly, offering users the capability to customize node descriptions in the tree view could prove advantageous.

Enhancing understanding of user context and question answerability. Our quantitative study results show that there is still an opportunity to improve the conversation module. These enhancements encompass recognizing unanswerable questions, effectively managing broad queries, and further refining the accuracy of analytical and visual query responses. Although the conversational module is not perfect in interpreting the ambiguous nature of natural languages, our efforts to make responses safe and explanatory still allowed participants to easily recover from mistakes.

7.2 Need for Rigorous and Inclusive Benchmark Testing

The cornerstone of our work is the conversational module, designed to address the challenges with keyboard navigation. While the existing dataset enabled a meaningful evaluation of response quality based on real-world queries, our study revealed the need for a more extensive and inclusive benchmarking dataset that incorporates viewpoints of blind and low-vision individuals [28].

Addressing advanced query types for a variety of chart types. Our evaluation was constrained not only by the four chart types but also by the limited range of questions (e.g., three query types), preventing a full assessment of VizAbility's capabilities. For example, although we noted some visual queries necessitating reasoning over data and contextual queries seeking chart information, our limited dataset was not sufficient to capture multi-category questions requiring advanced multi-hop reasoning [16]. Furthermore, our study did not evaluate situational questions related to a user's current interest point within the tree view, which our dataset lacks. Questions dependent on understanding previous conversational context were also not explored. Considering the generative abilities of LLMs, synthetically generating these types of questions using human-created examples and advanced prompt engineering might be a viable method [46].

Incorporating vision and testing with varied metrics. Although GPT4V falls short in performance compared to VizAbility, it shows promising visual description capabilities. This advancement contrasts with a few years ago when interpreting synthetic images like graphic designs and data visualizations was inferior to natural scene images [15]. Similarly, recent image-based ChartQAs still depend on OCR to extract text from images and create data tables [35, 41]. Analyzing the capabilities of emerging vision-LLMs—from low-level analytic tasks like value look-ups and comparisons to higher-level cognitive operations like explaining chart patterns—will help find ways to integrate vision capability with the symbolic processing of VizAbility to achieve greater performance. Moreover, expanding beyond mere correctness to include other pertinent measures such as fluency, informativeness, and relevance to the query [49] will be helpful for further improving the user experience of VizAbility.

7.3 Integrating into Existing Visualization Tools

Accommodating practitioners' visualization workflows. Since VizAbility operates under the assumption that a chart specification is available, it may not be directly applicable to charts currently found on the web. Instead, our vision is to integrate VizAbility within existing data visualization platforms. Prior research underscores that many data visualization practitioners base their choices on the accessibility features of these platforms [36]. Another study highlights the lack of accessible design support these tools offer [44]. Exploring the design space to determine how VizAbility can seamlessly fit into current data visualization workflows would be compelling. Additionally, considering the degree of customization for data visualization designers, such as setting default verbosity levels and offering query guidance & suggestions, warrants further investigation.

Supporting multiple-coordinated and dynamic data visualizations. Furthermore, many existing visualization tools are evolving beyond creating singular charts to producing multi-coordinated charts or dashboards. Exploring the expansion of VizAbility to accommodate these advanced forms of visualization presents an intriguing challenge. A recent study by Srinivasan et al. [62] investigates this problem by developing dashboards tailored for screen reader navigation, integrated with descriptions that aid in dashboard comprehension and interaction. Key questions arise in this context: How might we integrate a conversational agent into these dashboards? How should the agent resolve ambiguities in user queries about relevant charts? What constitutes an ideal design for mixed-initiative interaction in such environments? Furthermore, an exciting frontier is enabling users to interactively generate charts for data exploration. These questions open up new and unexplored avenues in the field of visualization accessibility.

8 CONCLUSION

In this study, we introduced VizAbility, a tool that enhances structured chart navigation through conversational interactions. Our quantitative assessments demonstrate a notable advancement beyond existing systems, and our qualitative analyses underscore the importance of the system's integrated approach and its dedication to transparency. As a direction for future research, we aim to develop a more comprehensive and inclusive benchmark dataset to foster continuous enhancements of VizAbility, along with integrating vision capabilities to bridge the current performance gaps.

ACKNOWLEDGMENTS

[currently redacted for anonymous review]

REFERENCES

[1] [n. d.]. amCharts. https://www.amcharts.com/accessibility/. Accessed: 2023-03-05.
[2] [n. d.]. CSS color codes. https://www.w3.org/wiki/CSS/Properties/color/keywords. Accessed: May 2, 2024.
[3] [n. d.]. LangChain CSV Agent Documentation. https://python.langchain.com/docs/integrations/toolkits/csv. Accessed: May 2, 2024.
[4] [n. d.]. LangChain: Serp API. https://python.langchain.com/docs/integrations/tools/serpapi. Accessed on Sep 7, 2023.
[5] [n. d.]. National Federation of the Blind. https://nfb.org/. (Accessed on 11/29/2023).
[6] [n. d.]. Observable Plot. https://observablehq.com/plot/. Accessed on Sep 7, 2023.
[7] [n. d.]. Vega View API. https://vega.github.io/vega/docs/api/view/. Accessed: May 2, 2024.
[8] [n. d.]. Whisper. https://openai.com/research/whisper. Accessed on Sep 7, 2023.
[9] 2023. W3C Complex Images. https://www.w3.org/WAI/tutorials/images/complex/.
[10] 2023. WAI Accessibility Principles. https://www.w3.org/WAI/fundamentals/accessibility-principles/.
[11] Robert Amar, James Eagan, and John Stasko. 2005. Low-level components of analytic activity in information visualization. In *IEEE Symposium on Information Visualization, 2005. INFOVIS 2005.* 111–117. https://doi.org/10.1109/INFVIS.2005.1532136
[12] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. 2015. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision* (Santiago, Chile). IEEE, 2425–2433.
[13] HK Ault, JW Deloge, RW Lapp, MJ Morgan, and JR Barnett. 2002. Evaluation of long descriptions of statistical graphics for blind and low vision web users. In *Computers Helping People with Special Needs: 8th International Conference, ICCHP 2002 Linz, Austria, July 15–20, 2002 Proceedings 8.* Springer, 517–526.
[14] Matt Blanco, Jonathan Zong, and Arvind Satyanarayan. 2022. Olli: An Extensible Visualization Library for Screen Reader Accessibility. In *IEEE VIS Posters.* http://vis.csail.mit.edu/pubs/olli
[15] Zoya Bylinskii, Nam Wook Kim, Peter O'Donovan, Sami Alsheikh, Spandan Madan, Hanspeter Pfister, Fredo Durand, Bryan Russell, and Aaron Hertzmann. 2017. Learning Visual Importance for Graphic Designs and Data Visualizations. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology* (Québec City, QC, Canada) *(UIST '17).* Association for Computing Machinery, New York, NY, USA, 57–69. https://doi.org/10.1145/3126594.3126653

[16] Wenhu Chen, Hanwen Zha, Zhiyu Chen, Wenhan Xiong, Hong Wang, and William Yang Wang. 2020. HybridQA: A Dataset of Multi-Hop Question Answering over Tabular and Textual Data. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, Trevor Cohn, Yulan He, and Yang Liu (Eds.). Association for Computational Linguistics, Online, 1026–1036. https://doi.org/10.18653/v1/2020.findings-emnlp.91

[17] Cheng-Han Chiang and Hung-yi Lee. 2023. Can large language models be an alternative to human evaluations? *arXiv preprint arXiv:2305.01937* (2023).

[18] Jaemin Cho, Jie Lei, Hao Tan, and Mohit Bansal. 2021. Unifying Vision-and-Language Tasks via Text Generation. In *Proceedings of the 38th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 139)*, Marina Meila and Tong Zhang (Eds.). PMLR, 1931–1942. https://proceedings.mlr.press/v139/cho21a.html

[19] Pramod Chundury, Biswaksen Patnaik, Yasmin Reyazuddin, Christine Tang, Jonathan Lazar, and Niklas Elmqvist. 2022. Towards Understanding Sensory Substitution for Accessible Visualization: An Interview Study. *IEEE Transactions on Visualization and Computer Graphics* 28, 1 (2022), 1084–1094. https://doi.org/10.1109/TVCG.2021.3114829

[20] Pierre Dognin, Igor Melnyk, Youssef Mroueh, Inkit Padhi, Mattia Rigotti, Jarret Ross, Yair Schiff, Richard A Young, and Brian Belgodere. 2022. Image captioning as an assistive technology: Lessons learned from vizwiz 2020 challenge. *Journal of Artificial Intelligence Research* 73 (2022), 437–459.

[21] Frank Elavsky, Lucas Nadolskis, and Dominik Moritz. 2024. Data Navigator: An Accessibility-Centered Data Navigation Toolkit. *IEEE Transactions on Visualization and Computer Graphics* 30, 1 (2024), 803–813. https://doi.org/10.1109/TVCG.2023.3327393

[22] Christin Engel and Gerhard Weber. 2017. Analysis of Tactile Chart Design *(PETRA '17)*. Association for Computing Machinery, New York, NY, USA, 197–200. https://doi.org/10.1145/3056540.3064955

[23] Christin Engel and Gerhard Weber. 2017. Improve the Accessibility of Tactile Charts. In *Human-Computer Interaction - INTERACT 2017*, Regina Bernhaupt, Girish Dalvi, Anirudha Joshi, Devanuj K. Balkrishan, Jacki O'Neill, and Marco Winckler (Eds.). Springer International Publishing, Cham, 187–195.

[24] Christin Engel and Gerhard Weber. 2018. A User Study to Evaluate Tactile Charts with Blind and Visually Impaired People. In *Computers Helping People with Special Needs*, Klaus Miesenberger and Georgios Kouroupetroglou (Eds.). Springer International Publishing, Cham, 177–184.

[25] Jean-Daniel Fekete, Jarke J. van Wijk, John T. Stasko, and Chris North. 2008. *The Value of Information Visualization.* Springer Berlin Heidelberg, Berlin, Heidelberg, 1–18. https://doi.org/10.1007/978-3-540-70956-5_1

[26] Leo Ferres, Gitte Lindgaard, Livia Sumegi, and Bruce Tsuji. 2013. Evaluating a Tool for Improving Accessibility to Charts and Graphs. *ACM Trans. Comput.-Hum. Interact.* 20, 5, Article 28 (nov 2013), 32 pages. https://doi.org/10.1145/2533682.2533683

[27] Jinlan Fu, See-Kiong Ng, Zhengbao Jiang, and Pengfei Liu. 2023. GPTScore: Evaluate as You Desire. arXiv:2302.04166 [cs.CL]

[28] Vinitha Gadiraju, Shaun Kane, Sunipa Dev, Alex Taylor, Ding Wang, Emily Denton, and Robin Brewer. 2023. "I wouldn't say offensive but...": Disability-Centered Perspectives on Large Language Models. In *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency (FAccT '23)*. Association for Computing Machinery, New York, NY, USA, 205–216. https://doi.org/10.1145/3593013.3593989

[29] John A Gardner and Vladimir Bulatov. [n. d.]. Making Scientific Graphics Accessible With Viewplus Iveo®.

[30] A. Jonathan R. Godfrey, Paul Murrell, and Volker Sorge. 2018. An Accessible Interaction Model for Data Visualisation in Statistics. In *Computers Helping People with Special Needs*, Klaus Miesenberger and Georgios Kouroupetroglou (Eds.). Springer International Publishing, Cham, 590–597.

[31] Cagatay Goncu and Kim Marriott. 2011. GraVVITAS: generic multi-touch presentation of accessible graphics. In *IFIP Conference on Human-Computer Interaction*. Springer, 30–48. https://doi.org//10.1007/978-3-642-23774-4_5

[32] Thomas R. G. Green and Marian Petre. 1996. Usability Analysis of Visual Programming Environments: A 'Cognitive Dimensions' Framework. *Journal of Visual Languages & Computing* 7, 2 (1996), 131–174. https://doi.org/10.1006/jvlc.1996.0009

[33] Danna Gurari, Qing Li, Abigale J Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P Bigham. 2018. Vizwiz grand challenge: Answering visual questions from blind people. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Salt Lake City, UT, USA). IEEE, 3608–3617.

[34] Highcharts 2022. Highcharts accessibility module. https://www.highcharts.com/docs/accessibility/accessibility-module

[35] Enamul Hoque, Parsa Kavehzadeh, and Ahmed Masry. 2022. Chart Question Answering: State of the Art and Future Directions. *Computer Graphics Forum* 41, 3 (2022), 555–572. https://doi.org/10.1111/cgf.14573 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.14573

[36] Shakila Cherise S Joyner, Amalia Riegelhuth, Kathleen Garrity, Yea-Seul Kim, and Nam Wook Kim. 2022. Visualization Accessibility in the Wild: Challenges Faced by Visualization Designers. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 83, 19 pages. https://doi.org/10.1145/3491102.3517630

[37] Crescentia Jung, Shubham Mehta, Atharva Kulkarni, Yuhang Zhao, and Yea-Seul Kim. 2022. Communicating Visualizations without Visuals: Investigation of Visualization Alternative Text for People with Visual Impairments. *IEEE Transactions on Visualization and Computer Graphics* 28, 1 (2022), 1095–1105. https://doi.org/10.1109/TVCG.2021.3114846

[38] Kushal Kafle and Christopher Kanan. 2017. Visual question answering: Datasets, algorithms, and future challenges. *Computer Vision and Image Understanding* 163 (2017), 3–20. https://doi.org/10.1016/j.cviu.2017.06.005 Language in Vision.

[39] Kushal Kafle, Brian Price, Scott Cohen, and Christopher Kanan. 2018. DVQA: Understanding Data Visualizations via Question Answering. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Computer Society, Los Alamitos, CA, USA, 5648–5656. https://doi.org/10.1109/CVPR.2018.00592

[40] Samira Ebrahimi Kahou, Adam Atkinson, Vincent Michalski, Ákos Kádár, Adam Trischler, and Yoshua Bengio. 2017. FigureQA: An Annotated Figure Dataset for Visual Reasoning. *CoRR* abs/1710.07300 (2017). arXiv:1710.07300 http://arxiv.org/abs/1710.07300

[41] Shankar Kantharaj, Xuan Long Do, Rixie Tiffany Ko Leong, Jia Qing Tan, Enamul Hoque, and Shafiq Joty. 2022. OpenCQA: Open-ended Question Answering with Charts. arXiv:2210.06628 [cs.LG]

[42] Dae Hyun Kim, Enamul Hoque, and Maneesh Agrawala. 2020. Answering Questions about Charts and Generating Visual Explanations. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3313831.3376467

[43] Jiho Kim, Arjun Srinivasan, Nam Wook Kim, and Yea-Seul Kim. 2023. Exploring Chart Question Answering for Blind and Low Vision Users. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 828, 15 pages. https://doi.org/10.1145/3544548.3581532

[44] N. W. Kim, G. Ataguba, S. C. Joyner, Chuangdian Zhao, and Hyejin Im. 2023. Beyond Alternative Text and tables: Comparative Analysis of Visualization Tools and Accessibility Methods. *Computer Graphics Forum* 42, 3 (2023), 323–335. https://doi.org/10.1111/cgf.14833 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.14833

[45] N. W. Kim, S. C. Joyner, A. Riegelhuth, and Y. Kim. 2021. Accessible Visualization: Design Space, Opportunities, and Challenges. *Computer Graphics Forum* 40, 3 (2021), 173–188. https://doi.org/10.1111/cgf.14298 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.14298

[46] Hyung-Kwon Ko, Hyeon Jeon, Gwanmo Park, Dae Hyun Kim, Nam Wook Kim, Juho Kim, and Jinwook Seo. 2023. Natural language dataset generation framework for visualizations powered by large language models. *arXiv preprint arXiv:2309.10245* (2023).

[47] Steven Landau and Karen Gourgey. 2001. Development of a talking tactile tablet. *Information Technology and Disabilities* 7, 2 (2001).

[48] Bongshin Lee, Eun Kyoung Choe, Petra Isenberg, Kim Marriott, and John Stasko. 2020. Reaching Broader Audiences With Data Visualization. *IEEE Computer Graphics and Applications* 40, 2 (2020), 82–90. https://doi.org/10.1109/MCG.2020.2968244

[49] Zhen Li, Xiaohan Xu, Tao Shen, Can Xu, Jia-Chen Gu, and Chongyang Tao. 2024. Leveraging Large Language Models for NLG Evaluation: A Survey. arXiv:2401.07103 [cs.CL]

[50] Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-Eval: NLG Evaluation using GPT-4 with Better Human Alignment. arXiv:2303.16634 [cs.CL]

[51] Alan Lundgard and Arvind Satyanarayan. 2022. Accessible Visualization via Natural Language Descriptions: A Four-Level Model of Semantic Content. *IEEE Transactions on Visualization and Computer Graphics* 28, 1 (2022), 1073–1083. https://doi.org/10.1109/TVCG.2021.3114770

[52] Kim Marriott, Bongshin Lee, Matthew Butler, Ed Cutrell, Kirsten Ellis, Cagatay Goncu, Marti Hearst, Kathleen McCoy, and Danielle Albers Szafir. 2021. Inclusive data visualization for people with disabilities: a call to action. *Interactions* 28, 3 (apr 2021), 47–51. https://doi.org/10.1145/3457875

[53] Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. 2022. ChartQA: A Benchmark for Question Answering about Charts with Visual and Logical Reasoning. arXiv:2203.10244 [cs.CL]

[54] Tomas Murillo-Morales and Klaus Miesenberger. 2017. Non-visually performing analytical tasks on statistical charts. In *Harnessing the Power of Technology to Improve Lives*. IOS Press, 339–346.

[55] Sabrina Paneels and Jonathan C Roberts. 2009. Review of designs for haptic data visualization. *IEEE Transactions on Haptics* 3, 2 (2009), 119–137.

[56] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. arXiv:1908.10084 [cs.CL]

[57] Prabodh Sakhardande, Anirudha Joshi, Charudatta Jadhav, and Manjiri Joshi. 2019. Comparing User Performance on Parallel-Tone, Parallel-Speech, Serial-Tone

and Serial-Speech Auditory Graphs. In *Human-Computer Interaction – INTERACT 2019*, David Lamas, Fernando Loizides, Lennart Nacke, Helen Petrie, Marco Winckler, and Panayiotis Zaphiris (Eds.). Springer International Publishing, Cham, 247–266.

[58] Leonard Salewski, Stephan Alaniz, Isabel Rio-Torto, Eric Schulz, and Zeynep Akata. 2023. In-Context Impersonation Reveals Large Language Models'Strengths and Biases. In *Advances in Neural Information Processing Systems*, A. Oh, T. Neumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (Eds.), Vol. 36. Curran Associates, Inc., 72044–72057. https://proceedings.neurips.cc/paper_files/paper/2023/file/e3fe7b34ba4f378df39cb12a97193f41-Paper-Conference.pdf

[59] Arvind Satyanarayan, Dominik Moritz, Kanit Wongsuphasawat, and Jeffrey Heer. 2017. Vega-Lite: A Grammar of Interactive Graphics. *IEEE Transactions on Visualization and Computer Graphics* 23, 1 (2017), 341–350. https://doi.org/10.1109/TVCG.2016.2599030

[60] Ather Sharif, Olivia H. Wang, Alida T. Muongchan, Katharina Reinecke, and Jacob O. Wobbrock. 2022. VoxLens: Making Online Data Visualizations Accessible with an Interactive JavaScript Plug-In. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 478, 19 pages. https://doi.org/10.1145/3491102.3517431

[61] Alexa F. Siu, Danyang Fan, Gene S-H Kim, Hrishikesh V. Rao, Xavier Vazquez, Sile O'Modhrain, and Sean Follmer. 2021. COVID-19 Highlights the Issues Facing Blind and Visually Impaired People in Accessing Data on the Web. In *Proceedings of the 18th International Web for All Conference* (Ljubljana, Slovenia) *(W4A '21)*. Association for Computing Machinery, New York, NY, USA, Article 11, 15 pages. https://doi.org/10.1145/3430263.3452432

[62] Arjun Srinivasan, Tim Harshbarger, Darrell Hilliker, and Jennifer Mankoff. 2023. Azimuth: Designing Accessible Dashboards for Screen Reader Users. In *Proceedings of the 25th International ACM SIGACCESS Conference on Computers and Accessibility* (<conf-loc>, <city>New York</city>, <state>NY</state>, <country>USA</country>, </conf-loc>) *(ASSETS '23)*. Association for Computing Machinery, New York, NY, USA, Article 49, 16 pages. https://doi.org/10.1145/3597638.3608405

[63] Marzia Taibbi, Cristian Bernareggi, Andrea Gerino, Dragan Ahmetovic, and Sergio Mascetti. 2014. Audiofunctions: Eyes-free exploration of mathematical functions on tablets. In *International Conference on Computers for Handicapped Persons*. Springer, 537–544. https://doi.org//10.1007/978-3-319-08596-8_84

[64] John R Thompson, Jesse J Martinez, Alper Sarikaya, Edward Cutrell, and Bongshin Lee. 2023. Chart Reader: Accessible Visualization Experiences Designed with Screen Reader Users. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (<conf-loc>, <city>Hamburg</city>, <country>Germany</country>, </conf-loc>) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 802, 18 pages. https://doi.org/10.1145/3544548.3581186

[65] Alexandra Vtyurina, Adam Fourney, Meredith Ringel Morris, Leah Findlater, and Ryen W. White. 2019. VERSE: Bridging Screen Readers and Voice Assistants for Enhanced Eyes-Free Web Search. In *Proceedings of the 21st International ACM SIGACCESS Conference on Computers and Accessibility* (Pittsburgh, PA, USA) *(ASSETS '19)*. Association for Computing Machinery, New York, NY, USA, 414–426. https://doi.org/10.1145/3308561.3353773

[66] Jiaan Wang, Yunlong Liang, Fandong Meng, Zengkui Sun, Haoxiang Shi, Zhixu Li, Jinan Xu, Jianfeng Qu, and Jie Zhou. 2023. Is ChatGPT a Good NLG Evaluator? A Preliminary Study. arXiv:2303.04048 [cs.CL]

[67] R. Wang, C. Jung, and Y. Kim. 2022. Seeing Through Sounds: Mapping Auditory Dimensions to Data and Charts for People with Visual Impairments. *Computer Graphics Forum* 41, 3 (2022), 71–83. https://doi.org/10.1111/cgf.14523 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.14523

[68] WebAIM. 2023. Screen Reader User Survey. https://webaim.org/projects/screenreadersurvey8/. Accessed: Dec 6, 2023.

[69] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In *Advances in Neural Information Processing Systems*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (Eds.), Vol. 35. Curran Associates, Inc., 24824–24837. https://proceedings.neurips.cc/paper_files/paper/2022/file/9d5609613524ecf4f15af0f7b31abca4-Paper-Conference.pdf

[70] Markus Weninger, Gerald Ortner, Tobias Hahn, Olaf Drümmer, and Klaus Miesenberger. 2015. ASVG- Accessible Scalable Vector Graphics: intention trees to make charts more accessible and usable. *Journal of assistive technologies* 9, 4 (2015), 239–246.

[71] Qi Wu, Damien Teney, Peng Wang, Chunhua Shen, Anthony Dick, and Anton van den Hengel. 2017. Visual question answering: A survey of methods and datasets. *Computer Vision and Image Understanding* 163 (2017), 21–40. https://doi.org/10.1016/j.cviu.2017.05.001 Language in Vision.

[72] Jonathan Zong, Crystal Lee, Alan Lundgard, JiWoong Jang, Daniel Hajas, and Arvind Satyanarayan. 2022. Rich Screen Reader Experiences for Accessible Data Visualization. *Computer Graphics Forum* 41, 3 (2022), 15–27. https://doi.org/10.1111/cgf.14519 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.14519

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009