LLM-powered Multimodal Insight Summarization for UX Testing

ANONYMOUS AUTHOR(S)

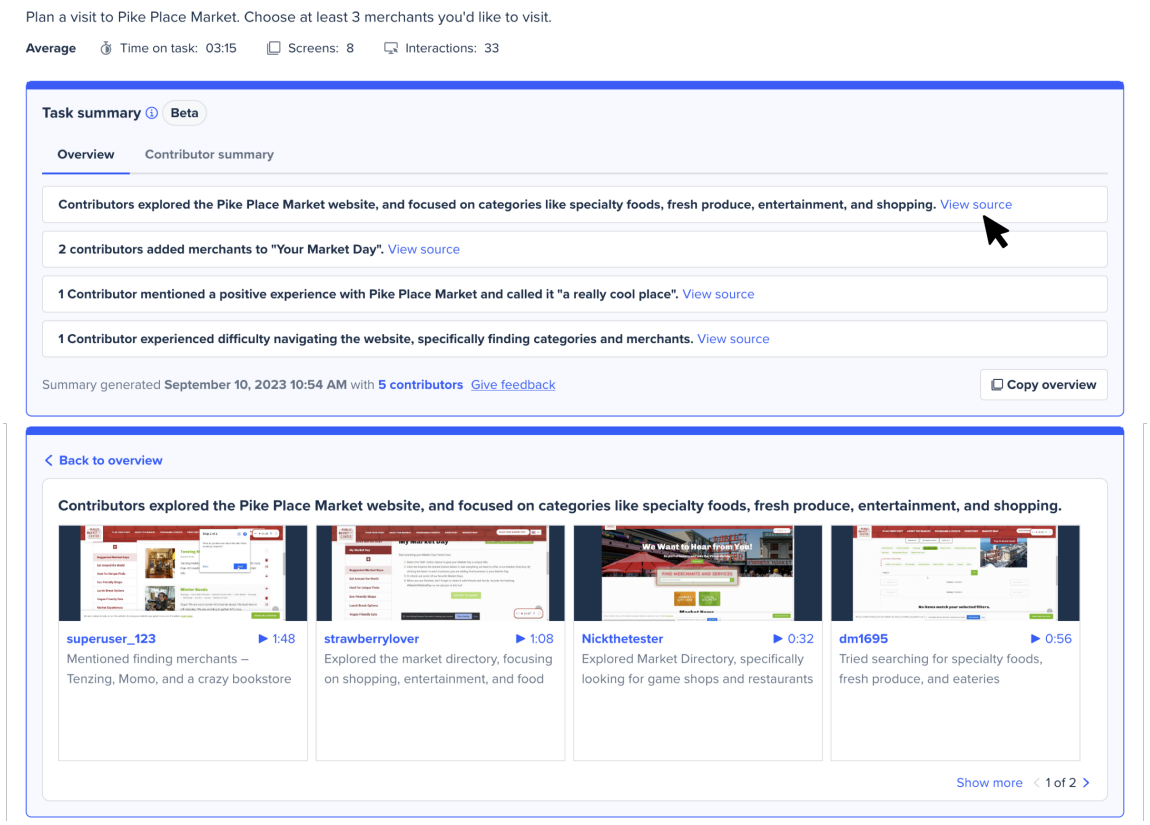

Fig. 1. This paper presents a large language model (LLM) approach for summarizing what people *said* and *did* during a UX test evidence, backed by evidence from multimodal data streams that allows users to quickly validate each insight.

User experience (UX) testing platforms capture many types of user feedback, including clickstream data, survey responses, screen recordings of participants performing tasks, and their think-aloud audio. Analyzing these multimodal data channels to extract insights, however, remains a time-consuming, manual process for UX researchers. This paper presents a large language model (LLM) approach for generating insights from multimodal UX testing data. By unifying verbal, behavioral, and design data streams into a novel natural language representation, we construct LLM prompts that generate insights combining information across all data types. To prevent hallucinations, each insight is accompanied by behavioral and verbal evidence that can be used to quickly verify its veracity. We evaluate LLM-generated insight summaries by deploying them in a popular remote UX testing platform, and present evidence that they help UX researchers more efficiently identify key findings from UX tests.

CCS Concepts: • **Human-centered computing** → **Human computer interaction (HCI)**; **Visualization**; *Interaction design.*

Additional Key Words and Phrases: UX research; usability testing; clickstream analytics; Sankey diagrams; sequence alignment

1 INTRODUCTION

User experience (UX) testing involves the collection of many types of *implicit* and *explicit* feedback, including clickstream data, survey responses, screen recordings of participants performing tasks, and think-aloud audio [23]. While UX testing platforms automate much of this collection, analyzing the feedback and correlating across different data channels to extract insights remains a manual, time-consuming process for UX practitioners [17, 24]. Although platforms have begun to leverage recent advances in Natural Language Processing — specifically, the advent of large, pre-trained transformer-based language models (LLMs) — to summarize written and verbal feedback, data streams without straightforward natural language representations have been mostly excluded from these efforts.

This paper presents an LLM-based approach for generating insights from multimodal UX testing data. More specifically, it describes a method for unifying verbal, behavioral, and design data streams into a novel natural language representation that can be used to prompt an LLM to generate insights combining information across all data types. To prevent hallucinations, the prompts require the LLM to map each insight back to verbal and behavioral evidence that can also be used to verify its accuracy.

We implemented and deployed multimodal insight summarization as an analysis feature in a remote UX testing platform. As participants interact with digital assets during a UX test, the platform captures their interactions, think-aloud audio, and the asset's underlying design data. The summarization feature first prompts an LLM to compute a sequence of natural language descriptions detailing what the participant *said* and *did* ordered by timestamp — a *multimodal transcript* — for each individual test session. Then, the feature prompts the LLM to extract, aggregate, and summarize insights across all the multimodal transcripts taken together (Figure 1). These summary insights synthesize information across verbal, behavioral, and design data from the UX tests, and link directly to timestamped actions in the transcript so that users can quickly and easily inspect the evidence used to generate them.

We report usage statistics and feedback collected from in-product surveys based on the real-world deployment starting August 30, 2023. UX researchers generally find the insight summarization feature makes them more efficient: they are able to extract and identify themes and reduce their overall "time-to-insight."

2 RELATED WORK

LLMs can potentially support UX testing and research workflows in many ways: assisting in test creation [5]; scaling qualititative analysis [28]; and even generating synthetic research data [18]. This paper examines how LLMs can scale qualitative analysis by generating insights that can summarize across different data streams.

2.1 Supporting UX testing and research

There are many remote UX testing platforms for evaluating digital experiences such as UserTesting [10], Sprig [7], Dscout [3], and Maze [4]; and UX research repositories for organizing customer feedback data such as Dovetail [2] and Notably [9]. Within the last year, many of these platforms have released LLM-powered features for summarizing natural language data streams such as verbal and written feedback [1, 6, 8]. This paper introduces an LLM approach for extracting insights from data streams without straightforward natural language representations.

Many research systems have also explored how to reduce analysis time for UX practitioners, or assist UX practitioners in gathering insights. Identifying issues that compromise task performance is critical to UX analysis, and UX practitioners frequently work under time constraints, making tools that speed analysis time particularly valuable [13, 17]. Systems are frequently designed to aid in the identification of anomalous user behavior [19]. Some automatically detect usability

issue indicators and surface them to users [25]. Others visualize timelines of think-aloud usability tests and identify usability problems by machine learning (ML) models [16], or create highly interactive visual environments to present streams of think-aloud, interaction, and eye-movement data in proximity [12].

The ZIPT system demonstrates that capturing interaction and design data during a usability test makes it easier to quantitatively analyze the results [14]. ZIPT leverages *interaction mining* [15] data to generate Sankey diagrams that summarize paths taken by users through an Android application and compute performance metrics such as task completion rate. This approach combines interaction mining data with think-aloud feedback to derive richer insights that correlate what people *did* with what they *said*.

2.2 Mapping Between Natural Language and Digital Design

Researchers have have developed ML-powered techniques for generating natural language representations of UI elements [22, 30] and screens [20, 27], promoting retrieval and accessibility applications. Conversely, researchers have also developed methods for generating UI interaction sequences based on natural language task descriptions leveraging transformers [21] and LLMs [26]. To produce multimodal summarizations, this approach prompts an LLM to generate a natural language representation that interleaves timestamped verbal, interaction, and design data into a multimodal transcript.

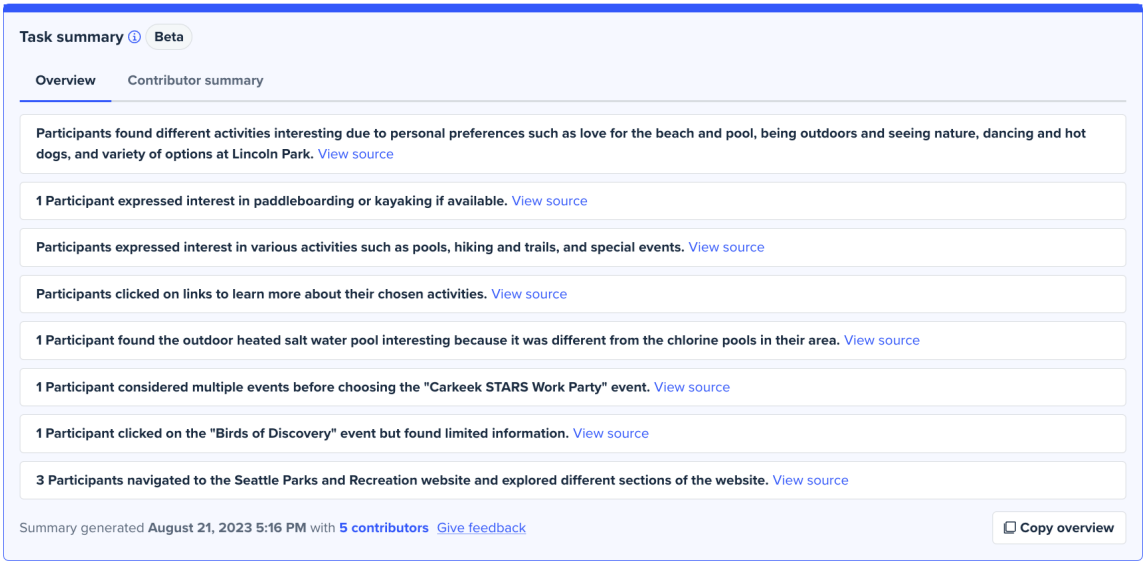

Fig. 2. The 'Overview' section displays an interactive list of insights related to the behavior, verbal, or design datastreams of one or more user sessions.

3 MULTIMODAL INSIGHT SUMMARIZATION IN ACTION

When conducting qualitative usability research, UX practitioners must analyze many aspects of each user session before synthesizing results, including: What the user said while performing a task What actions the user performed while attempting a task; The sentiment expressed by the user during the task; and What design elements the user interacted with in order to perform the task.

Consider the example of a UX practitioner assessing the user experience of the Seattle Department of Parks and Recreation website with the goal of improving navigability. The practitioner may recruit several usability study participants to "reserve a tennis court, pickleball court, or sports field," then review the user sessions to extract insights related to the experience. The practitioner would then need to spend significant time reviewing video data, taking notes, and qualitatively coding the sessions, including both the user's actions and their verbal feedback.

The proposed system instead feeds verbal, behavioral, and design datastreams into an LLM to generate a combined 'behavioral-verbal' transcript for each user session and insights synthesized from all user sessions. Practitioners can analyze test results within the insight summarization UI and export groups of the insights through a "Copy" button to add to a research report. In addition, the system affords drill-downs where practitioners can review the raw data for insight accuracy.

3.1 Insights Overview Interface

The practitioner begins on the "Overview" section, where they can review a list of insights related to the behavior, verbal, or design datastreams of one or more user sessions (Figure 2).

For this example, the practitioner may be interested in the users' success at the task of reserving a tennis or pickleball court, as well as how challenging they found it. The practitioner knows that to reserve a court, one must navigate to the "Sports" page. An insight tells them that "4 participants navigated to the "Sports" section on the website," indicating that four of five participants were likely successful. However, verbal feedback indicates that the process was not easy. Other insights include: "2 participants expressed uncertainty or confusion during their navigation," and "1 participant commented on the multiple clicks required to access the reservation site". However, once they found information, the users found it to be sufficient: "3 participants expressed satisfaction with the information they found."

In addition, the practitioner will want to know how users navigated the website's design. In this case, one insight shows that "2 participants used the search function" and "1 participant expressed a preference for using the search function." The practitioner may infer from the uncertainty users expressed and user comments on the complexity of the navigation (the "many clicks" to access the reservation site) that users opt to search rather than browse through the site. This may inform future decisions, whether investing in search more heavily in the future, or simplifying the navigation process.

3.2 View Source Interface

The system affords several means of checking the insights' accuracy.

Each insight has a link enabling UX practitioners to 'View source'. This leads to a screen that displays supporting thumbnails (Figure 3). Each thumbnail represents a clip within a session and is annotated with the behavioral and/or verbal transcript that informed the insight. In addition, the practitioner can click to watch the session video, which is timestamped to the parts of the session that led to the insight (Figure 4). They will also see a transcript of all think-aloud data next to the video.

Watching the video allows the practitioner to verify the insight and review any datastreams not directly included in the insight. For instance, for the insight "2 participants used the search function," a practitioner may check that two participants did, in fact, perform a search, but they might also review any comments the participants made while performing the search. Or, for the insight "2 participants expressed uncertainty or confusion during their navigation," the practitioner may wish to identify the specific screens or design elements about which the participants expressed uncertainty.

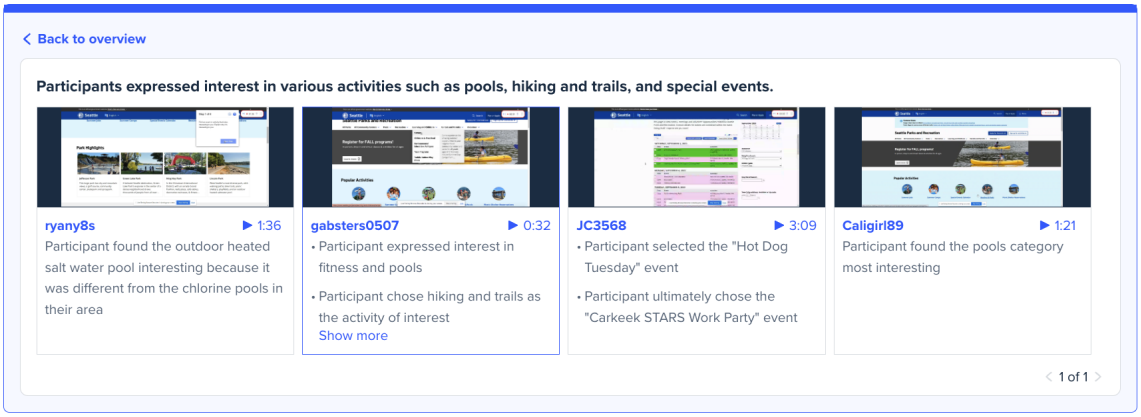

Fig. 3. Clicking 'View source' exposes annotated, timestamped videos that inform the insight, enabling the UX practitioner to quickly and easily verify the accuracy of the insight.

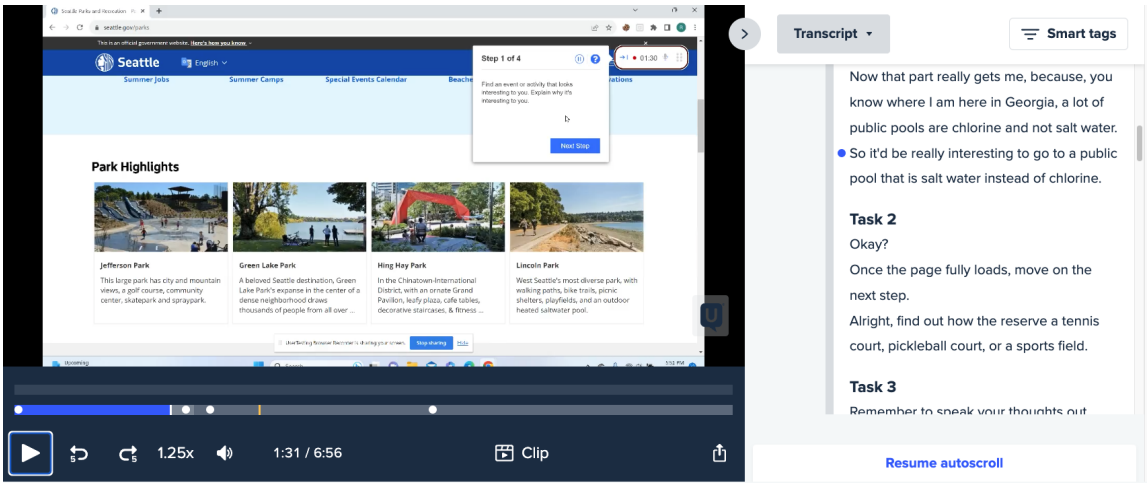

Fig. 4. Practitioners can watch the session video and review transcript of all think-aloud data.

3.3 Contributor Summary Interface

The UX practitioner can also verify the insights' accuracy by reviewing a combined behavioral-verbal transcript for each participant (Figure 5). Each line in the sequence links to the relevant timestamp in their session video, so the practitioner can review the participant's video along with a transcript of anything the participant said. Practitioners may also scan through the summary instead of watching the video, or use it for a starting point for manual analysis, as it mimics how a UX practitioner might take notes while reviewing a recorded session.

4 GENERATING THE MULTIMODAL INSIGHTS SUMMARY

To generate a multimodal insights summary, we first capture data from individual participant UX test sessions: UI design, think-aloud transcript, and event data. We then combine these datastreams into a multimodal 'transcript' of

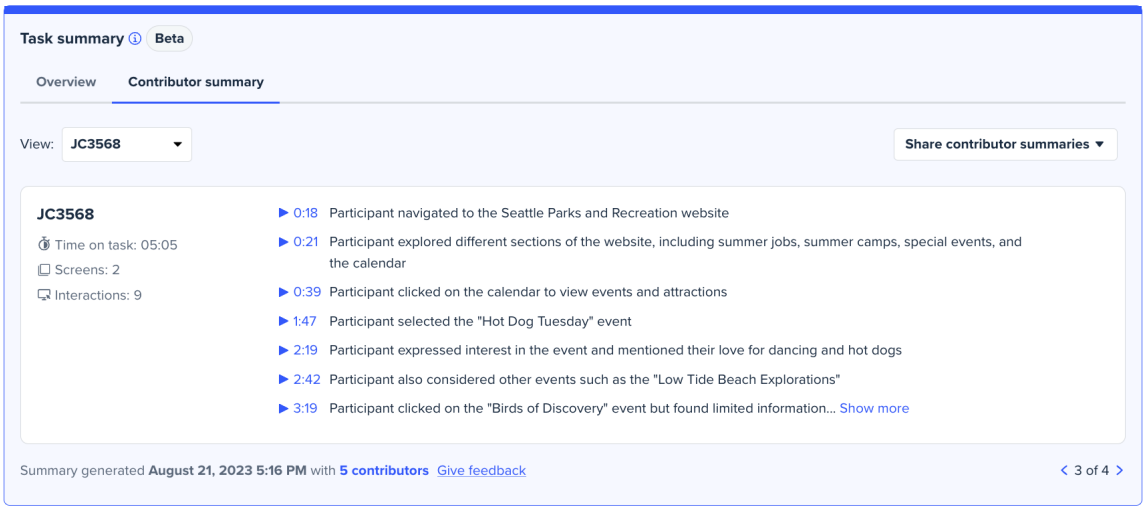

Fig. 5. Practitioners can review a combined behavioral-verbal transcript for each participant with timestamped links to the participant's session videos.

the individual session using an LLM. Finally, the multimodal transcripts from all user sessions are concatenated and processed through an LLM prompt to output the insights summary (Figure 6).

4.1 Data Stream Capture for Multimodal Insights Summary

Our data collection tool captures and synchronizes behavioral, audio data, screen capture data, and the webpage's Document Object Model (DOM). We employ a browser extension for real-time recording of event, DOM, and audio data during web-based user sessions.

Event Stream. The event stream records both user-level interactions like clicks and scrolls, and system-level activities such as page loads and form submissions. Every event is tagged with its specific type and timestamp. Given the high frequency of certain events like `scroll` and `mousemove`, a debouncing strategy is applied to filter out redundant events, thereby reducing noise in the data stream.

Design Layout and Rendertime Properties. The DOM is captured as a hierarchical tree structure that mirrors the browser's representation. Upon page load, an initial snapshot of the entire DOM is taken. Subsequent captures focus solely on changes—added, removed, or modified nodes. Each node carries metadata such as its type, attributes, and spatial bounds relative to the page.

Screen Capture. The video stream captures the participant's screen during the test. The video is recorded alongside the other data streams. Each video snippet is used to generate screenshots whenever an event occurs. The screenshots are assigned to the event that triggered them to be used in displaying the participant's screen as a data source in the final visualization alongside the full recording.

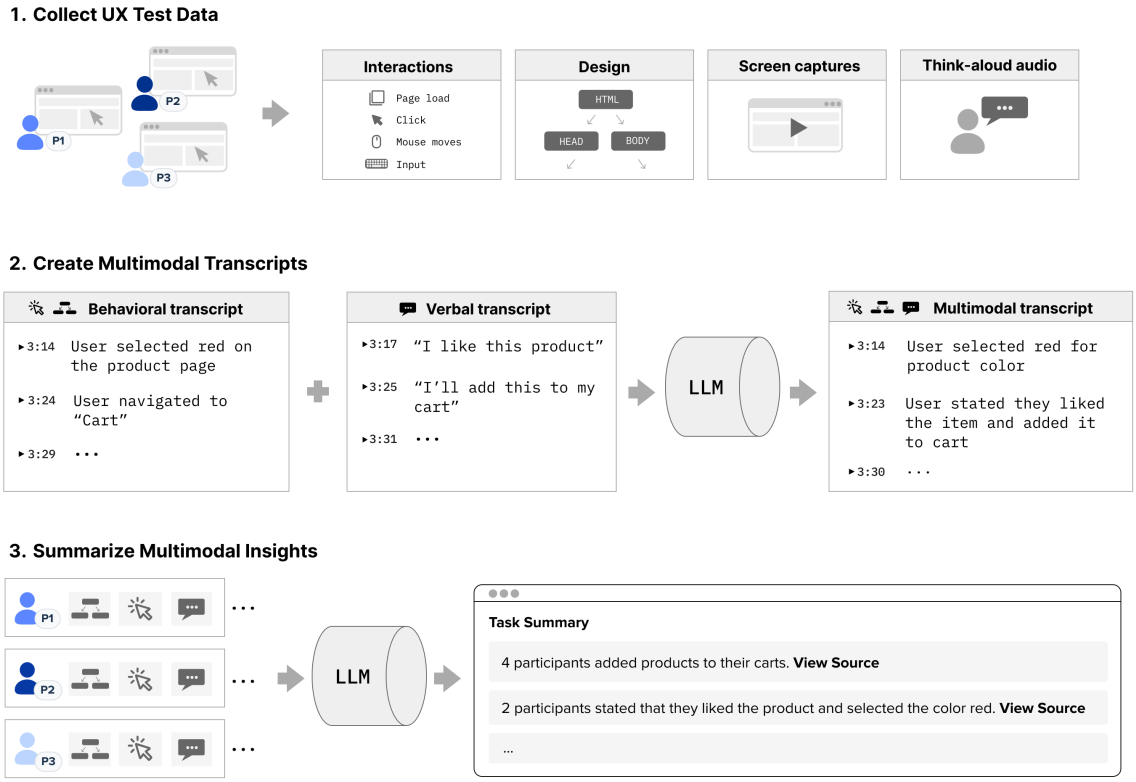

Fig. 6. To generate multimodal insights summary, we capture behavioral, audio data, screen capture data, and the webpage's Document Object Model (DOM) during each UX research session. These data streams are aggregrated and prompted through an LLM to generate a multimodal transcript of a single test session. The multimodal transcripts from all sessions are then concatenated and prompted through an LLM to generate the final summary.

Think-aloud audio. The audio stream captures the participant's think-aloud audio transcript data during the test, is recorded alongside the other data streams. Each audio snippet is timestamped to enable chronological alignment with other streams.

The recorded events, DOM snapshots, and audio snippets are chronologically aligned into sequences of "data moments". While these sequences can be extensive, not every data moment is pertinent for generating the multimodal insights summary. Hence, a filtering process retains only significant events like page loads, clicks, and form submissions. Furthermore, consecutive events of similar types, such as `input` and `scroll`, are consolidated to produce concise, non-redundant sequences.

Chronological alignment of events, DOM captures, and audio data forms the basis for generating multimodal transcripts that are ideal for further analysis and visualization, including Sankey diagrams. Although we do capture screenshots, they are used solely for display purposes in the final UI visualization, and not for data analysis.

4.2 Creating Multimodal Transcripts for Individual Sessions

Integrating Behavioral and Audio Transcripts. This multimodal approach aims to capture the full spectrum of user behavior by blending what actions users take with why they take them, as the combined behavioral-verbal transcript allows us to infer user intent.

The behavioral transcript encapsulates the user's interactions and the associated DOM information within a session. Each interaction is recorded as a text string, detailing the event type, timestamp, and pertinent DOM elements like the clicked element, complete webpage URL, and title. With experimentation, we found that the inclusion of full URLs and query parameters proves beneficial; for instance, an LLM can identify that a user applied filters on a shopping website merely by analyzing the URL's query parameters.

The inclusion of DOM elements serves a dual purpose. First, it adds a layer of context to the user's actions, helping the LLM to generate more accurate and nuanced insights. Second, it allows for greater granularity in understanding the user's interactions, making it possible for the LLM to pinpoint exactly where on a webpage the user engaged. This level of detail proves valuable when interpreting complex user journeys, enabling practitioners to discern not just what happened, but why it happened, thereby enriching the quality of the generated insights.

The audio transcript converts session-recorded think-aloud audio into text, segmenting it into timestamped sentences. Both transcripts are then chronologically integrated based on their timestamps to form the LLM prompt. When timestamp conflicts arise, the conflicting entries are simply placed adjacently, as it doesn't affect the LLM output. A unique identifier tags each integrated event to be referenced later in the output.

Generating a Multimodal Transcript via an LLM. The unified transcript serves as the input to an LLM, which is tasked with condensing multifaceted participant behavior into a list-formatted summary. By incorporating unique identifiers from the original data into each line item, the LLM ensures direct traceability to the source material. This feature is vital for subsequent data verification and analysis. We have found out that an LLM often excels at synthesizing multimodal data, such as user actions and spoken words, to produce cohesive insights. For instance, it can identify that a user is interested in a specific product by merging clicked DOM elements and spoken expressions.

In order to improve output quality, we found that explicitly marking sections with delimiters such as <PATHS> and <SUMMARY> helps in minimizing unintended or out-of-context outputs. Furthermore, the example serves as a tangible reference point for the LLM, influencing the quality and style of its output. Additionally, the choice of instructional language is crucial for shaping the LLM's output. The selection between terms like "participant" and "user" subtly guides the terminology the LLM adopts. The wording of the prompt should be considered carefully to ensure the LLM's output aligns with the desired language of the final visualization.

```
Below is a path a participant took during a UX test.
Summarize the path the participant took and include the notable data points found.
Rephrase the summary to be easy to digest and quickly understand in list format.
---
## Example ##
"Go check out the product on the below page. Explain why you chose the product."
<PATH>
24: Clicked on 'Red color' on 'Product 314' page (https://www.site.com/product/314)
25: Scrolled down on 'Product 314' page (https://www.site.com/product/314?color=red)
```

```
26: Said "I liked the red color"
27: Clicked on 'Add to cart' button on 'Product 314' page (https://www.site.com/product/314?color=red)
<SUMMARY>
- Participant clicked on the red color option on the product page [24]
- Participant liked the red color option and added the product to cart [26, 27]
---
"Instruction"
<PATH>
...
<SUMMARY>
```

Compiling the Multimodal Transcript. A multimodal transcript is a specialized data structure that holds the summarized insights in list form, while preserving references to the original data points. To assemble this data structure, the LLM output is divided into distinct line items. Unique identifiers within each item are extracted to link back to the original data. These identifiers are then purged from the text, yielding a clean list of insights. The resulting multimodal transcript not only feeds into the next phase of generating insights but also serves as a more digestible session summary to be displayed in the final visualization, allowing for a session-level overview. An example multimodal transcript output is shown below:

```
...
{ "id": 1,
  "text": "Participant clicked on the red color option on the product page",
  "sources": [{"type": "event", "id": 24}]},
{ "id": 2,
  "text": "Participant liked the red color option and added the product to cart",
  "sources": [{"type": "event", "id": 26}, {"type": "sentence", "id": 27}]}
...
```

4.3 Aggregating and Extracting Insights Across Multiple Sessions

Assembling Multiple Multimodal Transcripts. First we concatenate individual multimodal transcripts to create a unified input prompt to the LLM. This aggregated document comprises the multimodal transcripts from each session, while preserving unique identifiers for traceability back to individual sessions and events.

Generating Multi-Session Insights via an LLM. The assembled transcript is the starting point for the LLM, which is set to generate a list of key findings that span multiple user sessions. The LLM is prompted to consider commonalities, differences, and anomalies among sessions to give a full picture of user behavior and experiences. In our experiments, for example, it can spot if a product is frequently searched for, or if different search terms lead to the same product being clicked on. The output keeps the multi-faceted nature of the data, including both user actions and spoken words, but presents them in a more straightforward format.

Clear markers are added between each session's transcript to help the LLM identify the start and end points of each session. The prompt can be adjusted by changing or adding questions, allowing for more targeted insights. Using

placeholders for participant IDs improves the ability to trace back to the original data and makes it easier to present and store the findings later on.

The example prompt below acts as a guide for this process. It includes key questions and an example summary to help focus the LLM's analysis. Special tags like <CONTRIBUTOR_PATHS> and <SUMMARY> are also included in the prompt to keep the LLM on track, enhancing both the reliability and repeatability of the insights it produces.

```
Given the paths that contributors of a UX study took to achieve a task,
write a short holistic summary of bullet points to answer the below questions:


- Are there any differences and similarities in how contributors navigated to achieve the task?
- Were there any points where contributors expressed any comments such as sentiment or emotion?
- Did the contributors encounter any issues, blockers, or struggled during the task?
---
## Example ##
"Go check out the product on the below page. Explain why you chose the product."
<CONTRIBUTOR_PATHS>
participant_123:
1: Participant clicked on the red color option on the product page
2: Participant liked the red color option and added the product to cart
--
participant_456:
3: Participant dislikes the red option and opted instead for the blue option
4: Participant added the product to cart
<SUMMARY>
- [participant_123] and [participant_456] both added the product to cart [2, 4]
- While [participant_123] liked the red color option, [participant_456] opted for the blue option [1, 3]
---
"Instruction"
<CONTRIBUTOR_PATHS>
...
<SUMMARY>
```

Compiling the Multi-Session Insights Summary. The LLM output is parsed to separate individual line items. The unique identifiers in each line item are extracted and mapped back to their corresponding original events in a two step process linking to an individual line item in the respective sessions' multimodal transcript and then to the original source event. This process, known as "walking the references," ensures each insight is anchored in the raw data, thereby enhancing its credibility and interpretability for practitioners. This final step outputs a list of insights that span multiple UX test sessions, along with references to individual source data rooted in video recordings of the test session and in other visualizations such as path flow diagrams.

```
...
{ "id": 1,
```

```
  "text": "[participant_123] and [participant_456] both added the product to cart",
  "sources": [{"type": "event", "id": 24}, {"type": "event", "id": 25}]},
{ "id": 2,
  "text": "While [participant_123] liked the red color option, [participant_456] opted for the blue
      option",
  "sources": [{"type": "sentence", "id": 26}, {"type": "sentence", "id": 27}, {"type": "event", "id": 26}]}
...
```

4.4 Implementation

Data Collection. To capture video, event and DOM data for user sessions, we built a Chrome extension. The extension provides a basic UI for the user to start and stop the data collection. Apart from the UI, the extension has two main parts: a *content script* that is injected into the web pages that we record, and collects event and DOM data; and a *background script* that runs at a global browser level. The background script keeps track of page loads, injects our content script into web pages, handles the video recording using Chrome's `desktopCapture` API, and sends back the collected data to a server.

User and system events are collected by registering appropriate *event listeners* in the `window` and `document` objects. To capture DOM and the changes in the DOM over time, the extension's content script initially traverse the DOM tree, starting at the `document` node, and records a DOM event with this data. Then, it registers a `MutationObserver` [11] to listen to changes in the DOM.

Preparing and Prompting Data for LLM Insights. In implementing the generation of multimodal insights summaries, we leveraged both the GPT-3.5 16k and GPT-4 models via the OpenAI API. We faced constraints related to context size: 16,384 tokens for the GPT-3.5 16k model and 8,192 tokens for the GPT-4 model. These limits affected the number of sessions and content size we could include in a single task summary prompt. For sessions exceeding the token limit, we marked them as skipped and continued processing other sessions. Additionally, the API's rate limits necessitated a queuing system to manage requests. To optimize costs and rate-limit availability—given our processing of up to X UX studies per day—we primarily used GPT-3.5 models for individual session summaries. We reserved the more capable GPT-4 model for generating the final, aggregated multi-session insights.

Visualization UI. The web UI to visualize the Multimodal Insights Summary is built using React. The UI fetches cleaned session data for all sessions belonging to a test. The visualization is placed next to path flow diagrams and incorporates links to the sessions' source video in order to provide context for the insights. Drilldowns for each insight that show the source data along with screenshots are provided. When interacting with the drilldowns various UI elements such as within accompanying path flows are highlighted to provide context and link multiple visualizations together. The UI also allows users to download the insights as a CSV file format for further analysis.

5 EVALUATING MULTIMODAL INSIGHT SUMMARIZATION

Between August 30, 2023 and September 11, 2023, UX practitioners created 2,662 UX studies eligible for insights summarization. Once a study is complete, practitioners could choose to watch videos sequentially or review a 'Results' page, which included the option to generate an insights summary. During the deployment, 1,600 practitioners reviewed

the Results page, and 166 of them (10.4%) created 283 insights summaries. Practitioners also had the option to provide written feedback on the summary through a link, and 17 did so.

In addition, we gathered think-aloud feedback from UX practitioners who interacted with a prototype of the insights summarization.

5.1 Deployment feedback

Practitioners expressed that the insights sped their workflow: “This really helps with the fast data analysis and synthesis per task” (P4) and “I can see [this feature] being really helpful” (P15). Others requested more convenient means to export the insights into their reports or documentation, "It’d be nice to just be able to quick copy one [insight] at a time [in addition to the “Copy all” button]!” (P9), indicating that, for this practitioner, some insights were useful or relevant and others were not. Similarly, P7 wrote, “I would like to be able to edit this, fine tune, remove what is not correct and then… download.”

Some practitioners commented directly on accuracy (P3, P15, P16). Of these, two noted issues related to counting (P15, P16): “The source link cited six participants as having done the thing mentioned in the insight. Three of those six cited actually did the thing, the other three did not (and interestingly, three additional participants did the thing but were not included in the insight summary)” (P15). As LLMs have limited performance with arithmetic tasks, future implementations could take a rule-based approach for summing user feedback. Transcription errors also contributed to inaccurate insights: P3 wrote, “the response ‘one [tester] expressed annoyance at a small cost’ was actually the tester saying that they had a small cough and it was annoying.”

Finally, two practitioners commented on the content of the output, indicating that some users may benefit from being able to customize prompts. P10 suggested that the insights be less tailored to UX specifically, “I’d love to use this for marketing research, ad tests, etc… the output makes me think this is super tailored to UX research only,” while P1 wrote “I was expecting a summary in a narrative format… [with] recommendations as a conclusion.”

5.2 Unmoderated study

To understand more fully how UX practitioners might want to interact with the insights summary, we ran unmoderated user sessions with UX practitioners recruited from attendees of a UX research-focused professional conference. Participants were given a laptop that recorded their think-aloud data and screen interactions. Participants were allowed to freely explore a pre-generated insights summary and asked to speak their thoughts aloud while doing so. They were then asked to rate the usefulness of both the insights overview and the participant summaries on a 5-point likert scale, and finally asked what, if anything, they might want to change about the summary.

5.2.1 Respondents. We recruited participants from the attendees of a UX research-oriented professional conference in August 2023. Four UX practitioners opted to participate, three female, one male. Three reported their title as UX researcher, and the last reported a UX designer title.

Three of the four participants described using AI in their workflow, with two relying on ChatGPT for tasks like drafting emails or usability tests, and one relying on the AI summarization feature of a UX testing platform. One (P4) reported using no AI in their workflow due to their company’s data privacy policy.

5.2.2 Results. Participants expressed excitement about the potential for insights summarization, with two noting that the summary was “great overall” (P1, P4). All participants rated both the insights and participant summaries as useful or very useful.

Participants' opinions diverged, however, on which aspects of the insights summary were most useful. P4 specifically commented on the participant summaries: "I like the ability to kind of go through what the contributors are seeing, showing more," and the ability to verify accuracy through the video drilldowns. P2, conversely, commented "this [the participants summaries] is digging into stuff and that's fine, but, but this [the overview] is the actually useful part… like, if I'm gonna do a summary, I want the whole overview summary much more."

Two participants (P3, P4) indicated a need to customize and/or prioritize insights. P4 stated, "[if the summary] didn't quite get what I wanted it to, I would want the ability to either refresh it or… edit it," so that the summary would only display useful, relevant insights. They went on to describe saving or "pinning" specific insights then refreshing the rest to optimize the summary. P3 noted that, "the attitudinal stuff [that the summary generates] is cool, but it's not really the most important with our area of work." Instead, their team prioritizes task success, and customizing the summary accordingly would make it more useful.

P1 expressed interest in connecting insights between different tasks, remarking that auto-tagging or categorizing insights or entire summaries would be useful: "I wonder if [the summaries] need to be organized as like groups or if there needs to be some sort of defined tags… you can easily search for specific task summaries," as this would allow them to make connections between the results of tasks from different UX tests.

6 DISCUSSION AND FUTURE WORK

Feedback gathered from the deployment and unmoderated study indicates several avenues for future development: The ability for each practitioner to curate the insights produced for an individual task; increased user control over the prompt; and querying relevant insights. Each of these would improve support for practitioners' workflows. In addition, more affordances for exporting and sharing insights and related material could benefit users, allowing them to 'close the loop' and complete their workflow more quickly.

Future research could also explore the extent to which basing summarizations off of multimodal data streams mitigates risk of hallucination, and whether, as the number of datastreams or volume of data increases, LLMs are able to find insights human practitioners miss.

6.1 User-controlled insight curation

Users rarely reported that the insights' content was inaccurate during the deployment, but those that did observe inaccuracies reported frustration. In addition, practitioners did not find all insights generated by the system equally relevant; P7, for instance, wished to remove or hide certain insights.

A future implementation could allow users to 'hide' or remove specific insights, and future work could explore the impact of this affordance on users' preferences for a higher-recall system, as users would be able to curate which insights were shown.

The inability to collect user feedback on individual insights is a limitation of the current system. Hiding an insight could be used as an implicit feedback loop that the insight is unsatisfactory—whether irrelevant, uninteresting, or inaccurate. Similarly, allowing users to 'pin' insights could be used as an implicit endorsement of an insights' value.

Users could also be prompted for their motivation for hiding an insight (i.e., indicate whether the insight was "Inaccurate" or "Not relevant"). This data could then be used to further understand what types of insights are useful in a UX context, fine-tune LLM prompts to produce better insights, and custom-rank insights on a per-company or -individual basis, prioritizing the insights most likely to be useful to a specific user or team.

6.2 Custom prompting

The proposed system fully automates the insight generation process. Increased control could improve user satisfaction. Practitioners' intentions vary when creating UX tests, and those intentions are not always obvious from the task assigned to participants. A practitioner may, for instance, ask users to navigate a website while thinking aloud, but primarily be interested in organic user reactions to the homepage. Because of these kinds of variations in the way practitioners use the system, a single prompt cannot meet the needs of all users in all disciplines.

Allowing user input for prompting would allow practitioners to direct the LLM to produce insights most pertinent to their specific discipline or use case—for instance, the participant who noted that their team primarily needed information about task success could prompt the LLM accordingly.

Non-experts struggle with prompt creation [29]; however, instead of allowing a user full control over the prompt, a range of pre-generated prompts could be provided, or free-form user input could be included as an instruction. Either of these approaches would maintain output integrity while allowing some user control.

Future work could explore how extensive user instructions should be to optimize insight relevance.

6.3 Querying insights

As all datastreams are represented in natural language, future iterations of the system could support searching: Practitioners could search for mentions of pricing, for instance, or engagement with specific features or design elements (e.g., "Take a quiz"). The system could then return topic-specific insights.

This would give practitioners another means of insight customization, and would allow practitioners to verify the system's recall by checking whether the output is comprehensive.

6.4 Closing the loop

User feedback indicates that individual practitioners and teams can have unique needs, both for UX test goals and level of summary detail. Regardless of the test goal or practitioner's analysis preference, however, most practitioners must share UX test findings with their teams or stakeholders, either via a written report or slides. If the system can produce more relevant and comprehensive insights, practitioners can more quickly add those insights to their reports or slides, completing the UX feedback collection workflow.

In addition, the proposed system only allows practitioners to copy/export groups of insights, or individual participant summaries. Feedback indicates that practitioners would like to be able to more easily copy individual insights, or share clips or collections of clips. Future implementations could incorporate affordances for these actions.

6.5 Multimodal datastreams and hallucination

We hypothesize that summarization of multimodal data reduces LLM hallucination compared to summarization of single data streams alone; with more streams of data offering more context, the LLM may not need to supply as much 'missing' material. Feedback from the deployment reported no hallucinations beyond miscounting. Future work could more rigorously explore the occurrence of hallucinations for single vs. dual datastream summarization, or whether reduction in hallucinations is observed with the incorporation of additional datastreams.

6.6 Scaling qualitative analysis

Analysis time is the key limiting factor for qualitative research. Every additional user session adds 20 minutes or more of video review, plus additional time investment for qualitative coding and synthesis. In addition, the more unstructured qualitative data a study generates—and the the more kinds of data considered—the more challenging theme extraction and synthesis become for human analysts.

With sufficiently comprehensive, accurate, and verifiable multimodal insights, analysis time will no longer limit scale for qualitative research, broadening possibilities and scope for future UX practitioners in the future.

REFERENCES

[1] [n. d.]. AI Powered Qualitative Research in Notably | Notably — notably.ai. https://www.notably.ai/features/notably-ai-research. [Accessed 15-09-2023].

[2] [n. d.]. Customer Insights Hub — Dovetail — dovetail.com. https://dovetail.com/. [Accessed 15-09-2023].

[3] [n. d.]. dscout | Flexible, remote, in-context user research — dscout.com. https://dscout.com/. [Accessed 15-09-2023].

[4] [n. d.]. Maze | The continuous product discovery platform — maze.co. https://maze.co/. [Accessed 15-09-2023].

[5] [n. d.]. Maze AI | Elevate Your Product Research — maze.co. https://maze.co/ai/. [Accessed 15-09-2023].

[6] [n. d.]. Note and insight summaries — dovetail.com. https://dovetail.com/help/summarizing-your-data/. [Accessed 15-09-2023].

[7] [n. d.]. Sprig | User Insights Platform for Teams Building Better Product Experiences — sprig.com. https://sprig.com/. [Accessed 15-09-2023].

[8] [n. d.]. Sprig AI Analysis - GPT-Powered Real-Time Product Insights | Sprig — sprig.com. https://sprig.com/ai-analysis. [Accessed 15-09-2023].

[9] [n. d.]. Synthesis Platform for User Research | Notably — notably.ai. https://www.notably.ai/. [Accessed 15-09-2023].

[10] [n. d.]. UserTesting Human Insight Platform | Improve Customer Experience (CX) — usertesting.com. https://www.usertesting.com/. [Accessed 15-09-2023].

[11] 2020. MutationObserver. https://developer.mozilla.org/en-US/docs/Web/API/MutationObserver.

[12] Tanja Blascheck, Markus John, Kuno Kurzhals, Steffen Koch, and Thomas Ertl. 2015. VA 2: a visual analytics approach for evaluating visual analytics applications. *IEEE transactions on visualization and computer graphics* 22, 1 (2015), 61–70.

[13] Parmit K Chilana, Jacob O Wobbrock, and Andrew J Ko. 2010. Understanding usability practices in complex domains. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2337–2346.

[14] Biplab Deka, Zifeng Huang, Chad Franzen, Jeffrey Nichols, Yang Li, and Ranjitha Kumar. 2017. ZIPT: Zero-Integration Performance Testing of Mobile App Designs. In *Proc. UIST*. 727–736.

[15] Biplab Deka, Zifeng Huang, and Ranjitha Kumar. 2016. ERICA: Interaction mining mobile apps. In *Proc. UIST*. ACM, 767–776.

[16] Mingming Fan, Ke Wu, Jian Zhao, Yue Li, Winter Wei, and Khai N Truong. 2019. VisTA: Integrating machine intelligence with visualization to support the investigation of think-aloud sessions. *IEEE transactions on visualization and computer graphics* 26, 1 (2019), 343–352.

[17] Asbjørn Følstad, Effie Law, and Kasper Hornbæk. 2012. Analysis in practical usability evaluation: a survey study. In *proceedings of the SIGCHI conference on human factors in computing systems*. 2127–2136.

[18] Perttu Hämäläinen, Mikke Tavast, and Anton Kunnari. 2023. Evaluating large language models in generating synthetic hci research data: a case study. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 1–19.

[19] JongWook Jeong, NeungHoe Kim, and Hoh Peter In. 2020. Detecting usability problems in mobile applications on the basis of dissimilarity in user behavior. *International Journal of Human-Computer Studies* 139 (2020), 102364.

[20] Luis A Leiva, Asutosh Hota, and Antti Oulasvirta. 2022. Describing ui screenshots in natural language. *ACM Transactions on Intelligent Systems and Technology* 14, 1 (2022), 1–28.

[21] Yang Li, Jiacong He, Xin Zhou, Yuan Zhang, and Jason Baldridge. 2020. Mapping natural language instructions to mobile UI action sequences. *arXiv preprint arXiv:2005.03776* (2020).

[22] Yang Li, Gang Li, Luheng He, Jingjie Zheng, Hong Li, and Zhiwei Guan. 2020. Widget Captioning: Generating Natural Language Description for Mobile User Interface Elements. arXiv:2010.04295 [cs.LG]

[23] S. McDonald, H. M. Edwards, and T. Zhao. 2012. Exploring Think-Alouds in Usability Testing: An International Survey. *IEEE Transactions on Professional Communication* 55, 1 (2012), 2–19. https://doi.org/10.1109/TPC.2011.2182569

[24] Mie Nørgaard and Kasper Hornbæk. 2006. What do usability evaluators do in practice? An explorative study of think-aloud testing. In *Proceedings of the 6th conference on Designing Interactive systems*. 209–218.

[25] Fabio Paternò, Antonio Giovanni Schiavone, and Antonio Conti. 2017. Customizable automatic detection of bad usability smells in mobile accessed web applications. In *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services*. 1–11.

[26] Bryan Wang, Gang Li, and Yang Li. 2023. Enabling conversational interaction with mobile ui using large language models. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 1–17.

[27] Bryan Wang, Gang Li, Xin Zhou, Zhourong Chen, Tovi Grossman, and Yang Li. 2021. Screen2words: Automatic mobile UI summarization with multimodal learning. In *The 34th Annual ACM Symposium on User Interface Software and Technology*. 498–510.

[28] Ziang Xiao, Xingdi Yuan, Q Vera Liao, Rania Abdelghani, and Pierre-Yves Oudeyer. 2023. Supporting Qualitative Analysis with Large Language Models: Combining Codebook with GPT-3 for Deductive Coding. In *Companion Proceedings of the 28th International Conference on Intelligent User Interfaces*. 75–78.

[29] JD Zamfirescu-Pereira, Richmond Y Wong, Bjoern Hartmann, and Qian Yang. 2023. Why Johnny can't prompt: how non-AI experts try (and fail) to design LLM prompts. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 1–21.

[30] Xiaoyi Zhang, Lilian de Greef, Amanda Swearngin, Samuel White, Kyle Murray, Lisa Yu, Qi Shan, Jeffrey Nichols, Jason Wu, Chris Fleizach, Aaron Everitt, and Jeffrey P. Bigham. 2021. Screen Recognition: Creating Accessibility Metadata for Mobile Applications from Pixels. arXiv:2101.04893 [cs.HC]