SearchLog: A Web Browser Extension for Capturing Search Logs in Laboratory Studies

Jiaman He
RMIT University
Naarm / Melbourne, Australia
jiaman.he@student.rmit.edu.au

Riccardo Xia
Independent Researcher
Bologna, Italy
tcxia2001@gmail.com

Dana McKay
RMIT University
Naarm / Melbourne, Australia
dana.mckay@rmit.edu.au

Damiano Spina
RMIT University
Naarm / Melbourne, Australia
damiano.spina@rmit.edu.au

Johanne R. Trippas
RMIT University
Naarm / Melbourne, Australia
j.trippas@rmit.edu.au

Abstract

Natural search logs are valuable for studying search behavior in information seeking settings. We present *SearchLog*, an easy-to-install web browser extension for collecting natural search logs during lab-based studies. *SearchLog* enables participants to search the open web in a browser while recording structured interaction data from the mouse, keyboard, search activity, and browser state modules. The extension captures clicks, scrolling, hovered text, typed words, search queries, result rankings, AI-generated summaries when available, tab activity, and window changes. A local Flask backend stores each session as an ordered JSON event stream, including HTML snapshots and preprocessed search result data for later analysis. These logs can be used to derive measures such as query reformulation, page visits, dwell time, scroll behavior, tab switching, search path complexity, and exposure to AI-generated search content. By supporting natural browser-based search with structured experimental metadata, *SearchLog* provides a reusable resource to study search behavior across traditional and AI-enhanced search interfaces. SearchLog is available at https://github.com/peanutH/SearchLog-chromium-extension.

CCS Concepts

• **Information systems** → **Users and interactive retrieval**.

Keywords

Search Logs, Browser Extension, User Behavior, AI-Generated Summaries

ACM Reference Format:
Jiaman He, Riccardo Xia, Dana McKay, Damiano Spina, and Johanne R. Trippas. 2026. SearchLog: A Web Browser Extension for Capturing Search Logs in Laboratory Studies. In *Proceedings of the 35th ACM International Conference on Information and Knowledge Management (CIKM '26), November 07–11, 2026, Rome, Italy*. ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3799682.3840250

CIKM '26, Rome, Italy

ACM ISBN 979-8-4007-2539-5/2026/11
https://doi.org/10.1145/3799682.3840250

1 Introduction

Understanding how and why people search requires studying their behaviors throughout the search process, not only the final answers they find. Search behaviors can reveal users' motivations, intentions, and decision-making during information seeking [4]. Prior work has shown that signals such as eye tracking, search logs, and physiological signals can help researchers understand users' knowledge, goals, personal characteristics, satisfaction, and search strategies [9, 13, 14, 18, 23, 24]. Among these signals, search logs have been widely used in studies of information-seeking behavior [9, 12, 17, 22, 23, 32, 34, 35, 37, 42]. They offer an unobtrusive method for collecting large-scale search data from many system users, capturing interactions such as submitted queries, viewed results, clicks, and visited pages.

In laboratory studies, researchers may want participants to search naturally on the open web rather than use a custom search system [12, 36]. This is important because real-world search behavior is increasingly complex. Users may type and reformulate queries, view search result pages, open multiple pages, switch between tabs, scroll through content, return to earlier sources, and compare information across websites [2, 6, 8, 11, 19, 30]. More recently, search engines have also begun to include AI-generated summaries, which introduce new ways for users to inspect, evaluate, and use search results [1, 40]. Capturing these actions is therefore important for studying how people search, evaluate, and use information in realistic online environments.

Collecting search logs can be challenging in practice. Existing approaches often use custom scripts, manual observation, screen recordings, browser-based research tools, or controlled search interfaces [6, 8, 25, 30]. These approaches can require extra setup, programming expertise, or manual review, and controlled interfaces may not fully reflect open web search behavior [2, 37]. General web analytics tools also mainly support website-level analysis, while researchers often need logs organized by participant, session, and task across websites and search engines.

This need has become more important because search result pages now show more than ranked links. They may also include AI-generated summaries, such as Google AI Overviews, that provide information directly on the results page. As a result, users may read and use information without opening external websites. Logs that only record queries, clicks, and visited pages may therefore miss important parts of the search experience

To support this data collection, we present *SearchLog*,[1] an easy-to-install web extension that supports multiple Chromium-based browsers (e.g., Google Chrome, Opera, Brave, Microsoft Edge) for recording natural search behavior in laboratory studies. *SearchLog* lets participants search in a familiar browser while recording browser actions, page interactions, search result rankings, and, when available, an AI-generated summary. Currently, *SearchLog* supports two major commercial search engines (i.e., Google and Microsoft Bing) and detects their AI-generated summaries (i.e., AI Overviews, including conversations with AI Mode, and Copilot Search). Other search engines can be easily included using the same approach as described in Section 3 when needed as a future extension.

This paper makes the following contributions:

- We introduce *SearchLog*, an easy-to-install web extension for collecting natural search logs during laboratory-based information-seeking experiments. A demo video is available.[2]
- We define a structured logging schema that captures browser state, browser events, page-level interactions, page information, search-result rankings, and AI-generated summaries across participants' search sessions.

2 Related Work

Interaction Logs in Information Seeking Research. Interaction logs are widely used to examine how users search, browse, compare options, evaluate information, and make decisions [16, 38]. These logs are useful not only for interactive information retrieval and information-seeking research, but also for studies of consumer intent, online decision-making, and other web-based behavioral tasks [10, 15]. Common behavioral signals include submitted queries, clicked results, visited pages, dwell time, and scrolling behavior. These logs allow researchers to study the search process beyond final task outcomes, including how users formulate information needs, inspect results, reformulate queries, and compare information across sources [3, 9, 27, 35].

In laboratory studies, however, collecting such logs in a natural search environment is not always convenient. Researchers may rely on custom logging scripts, screen recordings, browser histories, or manually coded observations. These approaches can be useful, but they often require technical expertise, substantial post-processing, or study-specific infrastructure. Controlled search interfaces can provide detailed logs, but they may also limit the naturalness of participants' searches on the open web. These limitations motivate the development of reusable tools that support structured logging while allowing participants to use familiar browser-based search environments.

Browser-Based Logging Tools. Several systems have been developed to support logging in web search and browsing studies. We have summarized some tools/systems in Table 1. HCI Browser was designed to help researchers administer web search studies and collect browser event data, including visited websites, searches, windows, clicks, and scrolling events [7]. YASBIL provides a browser-extension-based logging solution together with a WordPress plugin for capturing browsing activity during user studies [5]. Other systems have supported search and browsing research through custom search panels, exploratory search interfaces, or controlled experimental platforms [31].

These tools demonstrate the value of browser-based logging for information seeking research. However, many existing systems are tied to a specific interface, platform, or study workflow. In contrast, *SearchLog* is designed as an easy-to-install Chromium extension for laboratory studies in which researchers want participants to search naturally on the open web. Rather than replacing the participant's search environment with a custom interface, *SearchLog* records browser-level and page-level interactions during an explicitly started study session.

Logging Search Result Pages with AI-generated Summaries. Search result pages now show more than ranked links. They also include features such as AI-generated summaries that provide information directly on the page. As a result, users may read and use information on the results page without clicking external websites, making query-, click-, and URL-based logs insufficient to capture the full search process [20, 21, 41].

SearchLog addresses this setting through a Chromium extension that is easy to install and collects richer browser-based search logs for laboratory studies. Compared with recent logging tools shown in Table 1, *SearchLog* captures a broader set of events, including browser and page interactions, submitted queries, result rankings, and AI-generated summary contents when available. These logs are linked with participant, session, task, and timestamp metadata, enabling researchers to analyze behavior across both traditional and AI-enhanced search interfaces. Rather than introducing a new search interface, *SearchLog* provides a practical and reusable tool for collecting structured logs from natural web search.

3 System Design and Logged Data

3.1 Architecture

SearchLog is an operating system-independent toolkit with a client and a server component. The client consists of an extension for Chromium-based browsers that captures users' interactions. The server is responsible for tracking sessions and receiving and storing logged data. We depict the workflow *SearchLog* follows in Figure 1.

The extension records browser-level and page-level information. We organize the logged data into four modules, with each module capturing a different type of activity. Page-level data includes interactions with web pages, such as mouse actions, keyboard input, and search engine content. These data are collected using JavaScript event listeners and by analyzing the web page's Document Object Model (DOM) structure and parsing scripts injected into the page. Browser-level data includes actions related to browser windows and tabs, collected via the browser APIs. Figure 2 shows the data structure, and the code repository provides implementation details.

The local server is implemented in Flask and provides endpoints to start a session, receive logged data, and stop a session. When a session starts, the server creates a unique session ID and stores all received data in a dedicated local folder.

Before a study session, the researcher installs the *SearchLog* Chromium extension and starts the local server on the experiment

[1] Code is available at https://github.com/peanutH/SearchLog-chromium-extension
[2] Demo video is available at https://youtu.be/nXHDAVfp7Zk

Table 1: Comparison of logging features across *SearchLog* and related exploratory search systems. ✓indicates that the feature is explicitly logged or supported by the system.

Tool	Clicks	Scrolling	Mouse Movement	Hovered Text/Elements	Typed Text	Queries	SERP Rankings	Tabs/ Windows	AI Summary Extraction
SearchBar [29]	–	–	–	–	–	✓	–	–	–
TrackThinkTS [26]	–	✓	–	–	–	✓	–	✓	–
Search-Logger [33]	✓	–	–	–	–	✓	–	✓	–
SearchPanel [31]	✓	–	–	–	–	✓	✓	–	–
OrgBox [39]	✓	–	✓	✓	–	✓	–	–	–
LogUI [28]	✓	✓	✓	✓	–	–	–	✓	–
***SearchLog* (Ours)**	✓	✓	✓	✓	✓	✓	✓	✓	✓

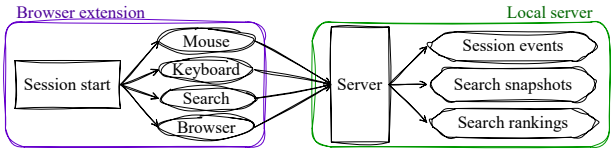

Figure 1: Workflow of *SearchLog*. After the researcher starts a session, participants naturally search in the web browser while the extension records their search and browser interactions. The local server saves the event logs and snapshots for later analysis.

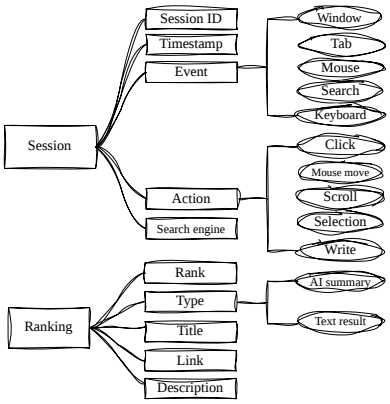

Figure 2: Structure of the data stored by *SearchLog*.

machine. The researcher then starts logging through the extension dialog, while participants use the browser naturally during the search task. After the task, the researcher stops logging through the extension dialog. Each session is saved as an ordered JSON event log, with search rankings and HTML snapshots stored separately. Figure 3 shows an example event log. Detailed installation and usage instructions are provided in the GitHub repository.

3.2 Session Lifecycle

During a lab experiment, participants use a Chromium-based browser with *SearchLog* installed to complete search tasks. A session has four main steps.

Session start. The researcher starts the local server and begins logging through the extension dialog. The participant then starts


```
{
  "session_id": "1779009704_a1cb7ffe",
  "timestamp": 1779009711393,
  "event": "keyboard",
  "action": "write",
  "typed": "how solar panels work"
},
{
  "session_id": "1779009704_a1cb7ffe",
  "timestamp": 1779009712330,
  "event": "search",
  "search_engine": "google",
  "query": "how solar panels work",
  "filename_html": "1779009712330_how solar panels work.html",
  "filename_ranking": "1779009712330_how solar panels work.json"
},
{
  "session_id": "1779009704_a1cb7ffe",
  "timestamp": 1779009713110,
  "event": "mouse",
  "action": "mouse_move",
  "mouse_from": { "x": 923, "y": 386 },
  "mouse_to": { "x": 921, "y": 387 },
  "hovered_text": ""
}
```


Figure 3: Example of event logs produced by *SearchLog*.

the search task. Researchers should inform participants what data are collected and when logging starts and stops. In our setup, data are collected only during active study sessions and are stored locally.

Logging. During the task, participants search and browse as usual. *SearchLog* records their interactions in temporal order and saves events to disk as they occur, reducing the risk of data loss.

Session end. When the task ends, the researcher stops logging through the extension dialog. The system closes the session and removes injected code from the browser. A new session can then be started for the next participant.

Artifacts for analysis. Each session produces a local folder containing interaction logs, search result rankings, and page HTML snapshots. These files can be used to study behavior such as task time, page visits, dwell time, tab switching, scrolling, and query events.

SearchLog is designed for researchers who need to collect natural search logs during laboratory-based information-seeking experiments. Instead of asking participants to use a custom search interface, *SearchLog* allows them to search on the open web in

a Chromium-based browser. During the study session, the extension records browser and page interactions, including search result rankings and, when available, an AI-generated summary.

Dry Run. To prevent logging errors or failures caused by interface changes or unexpected issues, we provide a dry-run function that simulates a search session. This allows users to verify that *SearchLog* is functioning correctly before beginning actual data collection.

Supported Browsers and Search Engines. *SearchLog* is an extension for Chromium-based browsers, including Google Chrome, Microsoft Edge, Opera, and Brave. As search engines, we currently support Google and Bing, with the option to add others.

Maintenance. By how *SearchLog* handles browser events and search engines, maintenance is required if the interfaces change. In particular, if the vendor of a search engine changes the layout of the page, *SearchLog* has to be updated accordingly to correctly extract data from the new layout.

4 Technical Validation

Experiment Setup. We conducted a small technical validation experiment using Google and Bing to examine whether *SearchLog* correctly records the main events generated during browser-based search sessions. The validation used non-sensitive demonstration tasks rather than real participant data. In each session, a tester started the local backend and browser extension, completed a predefined set of search actions, and then stopped the session.

The validation covered six test scenarios: (1) basic search, (2) multi-tab search, (3) mouse interaction, (4) keyboard input, (5) AI-generated summary capture, and (6) session-boundary handling. These scenarios were designed to cover common search behaviors, including submitting and reformulating queries, viewing search engine result pages, opening search results, scrolling web pages, hovering over text, typing in input fields, switching between tabs, and interacting with AI-generated summaries when available. We also included a session-boundary test to verify that logging stops after the session ends and that sensitive inputs, such as password fields, are masked. The detailed validation scripts and the corresponding search logs are provided in the GitHub evaluation folder.

Evaluation Criteria. After each validation session, we inspected the generated log files to check whether *SearchLog* correctly recorded the expected events in temporal order. We verified that it created the required session files, captured browser, page, and search-specific interactions, logged AI-generated summaries when present, and stopped logging after the session ended. This evaluation tested the functionality and reliability of the logging pipeline. A real-time logging example is shown in our demo video.

Results. We observed that *SearchLog* always correctly initializes and terminates a session, properly creating the required files and directories on disk. Mouse and keyboard logs are correctly captured. Mouse movements are saved as start and end coordinates, each scroll is recorded, and hovered text is identified. Keyboard presses are recorded and grouped into a single entry if they occur within a certain time frame. Commands, such as selection and copy-paste, are also recognized. Input inserted in password fields is correctly hidden in the logs. When querying Google and Bing, a search event is correctly produced in the logs, and the HTML snapshot is saved. The parsed search rankings and AI-generated summary are also correctly saved to disk and are coherent with the page's content. Browser-level events involving tab and window operations are also correctly captured and recorded. We report a summary in Table 2.

Table 2: Summary of technical validation results.

Component	Result
Session control	Starts and ends correctly.
Clicks	Captures position.
Scrolling	Captures start and end position.
Mouse Movement	Captures start and end position.
Hovered Text/Elements	Captures text content.
Typed Text	Captures keys and commands.
Queries	Detects search engine queries.
SERP Rankings	Collects ranking correctly.
Tabs/Windows	Captures open, close, focus, and loading.
AI Summary Extraction	Detects content of AI summaries.
HTML snapshots	HTML files saved correctly.

5 Final Remarks

SearchLog is released as a reusable research toolkit under Apache 2.0 license, including the Chromium extension, local Flask backend, documentation, and sample logs from non-sensitive demonstration tasks. Each session produces a local folder containing an ordered event stream, search result data, AI-generated summaries when available, and HTML snapshots. The documentation describes installation, session workflow, output structure, and example scripts for deriving common search behavior measures.

Ethical Considerations. Because natural search logs may contain sensitive information, including URLs, page titles, typed text, search queries, AI-generated summaries, and page snapshots, *SearchLog* is designed for explicitly started laboratory sessions rather than continuous background monitoring. Researchers should clearly inform participants about what is collected, when logging starts and stops, how data are stored, and how withdrawal or deletion requests are handled. Study protocols should avoid personal accounts, passwords, or highly private search tasks, and researchers should collect only the data needed for their study.

Limitations and Future Work. *SearchLog* currently has several limitations. It is designed for Chromium-based browsers and does not support Firefox and Safari without rewriting. In addition, search result extraction may require ongoing maintenance if commercial search engines update their page layouts. In future work, we plan to extend *SearchLog* to support more search engines (e.g., DuckDuckGo, Yahoo!, Baidu, Yandex, etc.) and to log interactions with LLM-based systems, such as ChatGPT, Copilot, Gemini, or Claude, using a similar approach that analyzes the page DOM and extracts relevant interaction content.

Acknowledgments

This research is supported by Australian Research Council (ARC) Centre of Excellence for Automated Decision-Making and Society (ADM+S, CE200100005).

GenAI Usage Disclosure

During the preparation of this work, the author used ChatGPT in order to assist with language editing and improving the clarity of portions of the manuscript. The AI tools were not used to generate data, conduct analyses, or make scientific decisions. After using this tool/service, the author reviewed and edited the content as needed and take full responsibility for the content of the published article.

References

[1] Sara Allawati, Dana McKay, Mark Sanderson, Paul Thomas, and Johanne Trippas. 2026. An Eye Tracking Study: Are AI Overviews Changing Search Behavior?. In *Proceedings of the 49th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 29–40. doi:10.1145/3805712.3809530

[2] Anne Aula, Rehan M. Khan, and Zhiwei Guan. 2010. How Does Search Behavior Change as Search Becomes More Difficult?. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '10)*. ACM, 35–44. doi:10.1145/1753326.1753333

[3] Nicholas J Belkin. 1980. Anomalous States of Knowledge as A Basis for Information Retrieval. *Canadian journal of information science* 5, 1 (1980), 133–143.

[4] Nicholas J. Belkin. 2008. Some(what) Grand Challenges for Information Retrieval. *SIGIR Forum* 42, 1 (June 2008), 47–54. doi:10.1145/1394251.1394261

[5] Nilavra Bhattacharya and Jacek Gwizdka. 2021. YASBIL: Yet another search behaviour (and) interaction logger. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2585–2589.

[6] Robert Capra. 2009. HCI Browser: A Tool for Studying Web Search Behavior. In *Proceedings of the Workshop on Understanding the User: Logging and Interpreting User Interactions in Information Search and Retrieval (UIIR '09)*. 38–41.

[7] Robert Capra. 2010. HCI browser: A tool for studying web search behavior. *Proceedings of the American Society for Information Science and Technology* 47, 1 (2010), 1–2.

[8] Robert Capra. 2011. HCI Browser: A Tool for Administration and Data Collection for Studies of Web Information Seeking. In *Human-Computer Interaction. Users and Applications*. Lecture Notes in Computer Science, Vol. 6764. Springer, 189–198. doi:10.1007/978-3-642-21708-1_30

[9] Michael J Cole, Chathra Hendahewa, Nicholas J Belkin, and Chirag Shah. 2015. User activity patterns during information search. *ACM Transactions on Information Systems (TOIS)* 33, 1 (2015), 1–39.

[10] Sharad Goel, Jake M Hofman, Sébastien Lahaie, David M Pennock, and Duncan J Watts. 2010. Predicting consumer behavior with Web search. *Proceedings of the National academy of sciences* 107, 41 (2010), 17486–17490.

[11] Laura A. Granka, Thorsten Joachims, and Geri Gay. 2004. Eye-Tracking Analysis of User Behavior in WWW Search. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '04)*. ACM, 478–479. doi:10.1145/1008992.1009079

[12] Jiaman He. 2026. User Search Behavior and Knowledge Effects in Hybrid Search Environment. In *Proceedings of the 49th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, Melbourne, Australia. doi:10.1145/3805712.3808356

[13] Jiaman He, Zikang Leng, Dana McKay, Johanne R Trippas, and Damiano Spina. 2025. Characterising Topic Familiarity and Query Specificity Using Eye-Tracking Data. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2602–2606.

[14] Jiaman He, Marta Micheli, Damiano Spina, Dana McKay, Johanne R Trippas, and Noriko Kando. 2026. Characterizing Personality from Eye-Tracking: The Role of Gaze and Its Absence in Interactive Search Environments. In *Proceedings of the 2026 Conference on Human Information Interaction and Retrieval*. 193–203.

[15] Bernard J Jansen. 2006. Search log analysis: What it is, what's been done, how to do it. *Library & information science research* 28, 3 (2006), 407–432.

[16] Bernard J Jansen and Amanda Spink. 2006. How are we searching the World Wide Web? A comparison of nine search engine transaction logs. *Information processing & management* 42, 1 (2006), 248–263.

[17] Bernard J. Jansen, Amanda Spink, and Tefko Saracevic. 2000. Real Life, Real Users, and Real Needs: A Study and Analysis of User Queries on the Web. *Information Processing & Management* 36, 2 (2000), 207–227. doi:10.1016/S0306-4573(99)00056-4

[18] Kaixin Ji, Danula Hettiachchi, Flora D. Salim, Falk Scholer, and Damiano Spina. 2024. Characterizing Information Seeking Processes with Multiple Physiological Signals. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Washington DC, USA) *(SIGIR '24)*. Association for Computing Machinery, New York, NY, USA, 1006–1017. doi:10.1145/3626772.3657793

[19] Thorsten Joachims, Laura Granka, Bing Pan, Helene Hembrooke, and Geri Gay. 2005. Accurately Interpreting Clickthrough Data as Implicit Feedback. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '05)*. ACM, 154–161. doi:10.1145/1076034.1076063

[20] Dmitry Lagun, Chih-Hung Hsieh, Dale Webster, and Vidhya Navalpakkam. 2014. Towards Better Measurement of Attention and Satisfaction in Mobile Search. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR '14)*. ACM, 113–122. doi:10.1145/2600428.2609631

[21] Jane Li, Scott B. Huffman, and Akihito Tokuda. 2009. Good Abandonment in Mobile and PC Internet Search. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '09)*. ACM, 43–50. doi:10.1145/1571941.1571951

[22] Yidong Liang, Zhijing Wu, Yuchen He, Fengming Liang, Kexin Liu, and Jiaxin Mao. 2025. A Flexible User Study Platform for Generative Information Retrieval. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '25)*. ACM, 4066–4070. doi:10.1145/3726302.3730140

[23] Jingjing Liu, Chang Liu, and Nicholas J Belkin. 2016. Predicting information searchers' topic knowledge at different search stages. *Journal of the Association for Information Science and Technology* 67, 11 (2016), 2652–2666.

[24] Mengyang Liu, Jiaxin Mao, Yiqun Liu, Min Zhang, and Shaoping Ma. 2019. Investigating Cognitive Effects in Session-level Search User Satisfaction. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*. 923–931. doi:10.1145/3292500.3330981

[25] Ammar Makhlouf, Yutaka Arakawa, and Yutaka Watanabe. 2022. A Privacy-Aware Browser Extension to Track User Search Behavior for Programming Learning. In *Mobile and Ubiquitous Systems: Computing, Networking and Services (MobiQuitous 2021)*. Springer, 684–694. doi:10.1007/978-3-030-94822-1_51

[26] Jihed Makhlouf, Yutaka Arakawa, and Ko Watanabe. 2021. A privacy-aware browser extension to track user search behavior for programming course supplement. In *International Conference on Mobile and Ubiquitous Systems: Computing, Networking, and Services*. Springer, 783–796.

[27] Gary Marchionini. 1995. *Information Seeking in Electronic Environments*. Cambridge university press.

[28] David Maxwell and Claudia Hauff. 2021. LogUI: contemporary logging infrastructure for web-based experiments. In *European Conference on Information Retrieval*. Springer, 525–530.

[29] Dan Morris, Meredith Ringel Morris, and Gina Venolia. 2008. SearchBar: a search-centric web history for task resumption and information re-finding. In *Proceedings of the SIGCHI conference on human factors in computing systems*. 1207–1216.

[30] Srishti Palani and Steven P. Dow. 2025. Contextualizing the Role of Web Search in Creative Workflows: Insights from a Longitudinal Study. In *Proceedings of the 2025 ACM SIGIR Conference on Human Information Interaction and Retrieval (CHIIR '25)*. ACM.

[31] Pernilla Qvarfordt, Simon Tretter, Gene Golovchinsky, and Tony Dunnigan. 2014. Searchpanel: framing complex search needs. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*. 495–504.

[32] Craig Silverstein, Monika Henzinger, Hannes Marais, and Michael Moricz. 1999. Analysis of a Very Large Web Search Engine Query Log. *SIGIR Forum* 33, 1 (1999), 6–12.

[33] Georg Singer, Ulrich Norbisrath, Eero Vainikko, Hannu Kikkas, and Dirk Lewandowski. 2011. Search-logger analyzing exploratory search tasks. In *Proceedings of the 2011 ACM Symposium on Applied Computing*. 751–756.

[34] Johanne R. Trippas, Sara Fahad Dawood Al Lawati, Joel Mackenzie, and Luke Gallagher. 2024. What do Users Really Ask Large Language Models? An Initial Log Analysis of Google Bard Interactions in the Wild. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2024)*. 2703–2707. doi:10.1145/3626772.3657914

[35] Johanne R. Trippas, Luke Gallagher, and Joel Mackenzie. 2024. Re-evaluating the Command-and-Control Paradigm in Conversational Search Interactions. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management (CIKM '24)*. 2260–2270. doi:10.1145/3627673.3679588

[36] Kelsey Urgo and Jaime Arguello. 2025. Search as learning. *Foundations and Trends® in Information Retrieval* 19, 4 (2025), 365–556.

[37] Tung Vuong, Miamaria Saastamoinen, Giulio Jacucci, and Tuukka Ruotsalo. 2019. Understanding User Behavior in Naturalistic Information Search Tasks. *Journal of the Association for Information Science and Technology* 70, 11 (2019), 1248–1261. doi:10.1002/asi.24201

[38] Xuanhui Wang and ChengXiang Zhai. 2007. Learn from web search logs to organize search results. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*. 87–94.

[39] Austin R Ward and Robert Capra. 2021. OrgBox: Supporting cognitive and metacognitive activities during exploratory search. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*. 2570–2574.

[40] Claire Wardle, Shaydanay Urbani, and Eric Wang. 2025. Evolving Health Information–Seeking Behavior in the Context of Google AI Overviews, ChatGPT, and Alexa: Interview Study Using the Think-Aloud Protocol. *Journal of Medical Internet Research* 27 (2025), e79961.

[41] Zhijing Wu, Mark Sanderson, B. Barla Cambazoglu, W. Bruce Croft, and Falk Scholer. 2020. Providing Direct Answers in Search Results: A Study of User Behavior. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management (CIKM '20)*. ACM, 1635–1644. doi:10.1145/3340531.3412017

[42] Junhao Zhang and Haiming Liu. 2025. Theory-Based User Search Behaviour Modelling and Understanding through Search Log Analysis. In *Proceedings of the 2025 ACM SIGIR Conference on Human Information Interaction and Retrieval (CHIIR '25)*. ACM. doi:10.1145/3698204.3716462