Different Keystrokes for Different Folks: Visualizing Crowdworker Querying Behavior

Rodger Benham
RMIT University
Melbourne, Australia

Joel Mackenzie
University of Melbourne
Melbourne, Australia

J. Shane Culpepper
RMIT University
Melbourne, Australia

Alistair Moffat
University of Melbourne
Melbourne, Australia

ABSTRACT

Search engine users retrieve relevant information for an *information need* using keyword queries. Different users may have similar information needs, but use different query terms. The resulting *user query variations* can provide a wealth of useful information to IR researchers. Most recently, the keystroke-level telemetry data gathered as part of the CC-News-En collection provides important insights into how users create queries for a search task, at a level of detail not possible using a normal query log. In this demo, we present an interactive tool that enables practitioners to visualize users formulating queries. Our new tool is a temporal simulation of the typing behavior of crowdworkers, grouped by information need. It provides the ability to directly compare the cognitive behavior of multiple users simultaneously, and observe how query keyword selection and ordering happens before a final query is submitted to a search engine. To demonstrate the benefit of our tool, we include a qualitative study of four different user behavior patterns which were observed in the CC-News-En collection.

ACM Reference Format:
Rodger Benham, Joel Mackenzie, J. Shane Culpepper, and Alistair Moffat. 2021. Different Keystrokes for Different Folks: Visualizing Crowdworker Querying Behavior. In *Proceedings of the 2021 ACM SIGIR Conference on Human Information Interaction and Retrieval (CHIIR '21), March 14–19, 2021, Canberra, ACT, Australia.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3406522.3446054

1 INTRODUCTION

Test collections containing user query variations (UQVs) facilitate exploration of several interesting facets of the end-to-end search process by providing sets of queries corresponding to the same information need. Not only do these collections demonstrate that there are many different and valid query representations for an information need, but also they can be used to show that single-shot query optimization is akin to model over-fitting, and may not adequately capture the most salient aspects of an information need [11]. The first widely available collection of user query variations, UQV100 [2] features 100 topics with a mean of 72.5 queries per topic before spelling normalization. Although the effect of multiple query variations on improving effectiveness has been well understood since the early 1990s [4], UQV100 inspired a renewed interest in rank fusion [3, 5, 7, 9], automated query generation [6], and failure analyses, with the common goal of better understanding why human and machine-generated queries can have very different performance characteristics [15]. Following on from UQV100, the recent CC-News-En collection [16] contains 10,437 queries collected against 173 topics and includes 10,089 keystroke-level telemetries.

In this paper, we present the *Common Crawl News Query Explorer*,[1] which enables information retrieval practitioners to observe and compare human query formulation *a posteriori*. It uses keystroke-level interactions from the CC-News-En collection to display real-time querying behavior, along with associated metadata describing each crowdworker task. Different configuration options are provided such as specifying the number of queries to display, whether to synchronize when each user first starts typing, and adjusting playback speed to be faster or slower.

Query tools have facilitated visualization in the past, but to our knowledge, no readily available tool exists that allows playback of all editing interactions during query formulation. Scells and Zuccon [19] propose a tool that assists users to generate complex systematic review queries; it was recently extended by Li et al. [14], but does not facilitate visualizing the query writing process across a pool of searchers. Hoeber et al. [13] describe a tool to visualize the query space of a Google session, to aid query editing.

Our objective here is to enable researchers to more comprehensively study the factors and cognitive decisions made by users, and thus to better understand query variability.

2 QUERY GENERATION VISUALIZATION

This section describes the CC-News-En data, and the technical details of the *Common Crawl News Query Explorer*.

Query Formulation Data. A common approach used to simulate information needs in crowdworking environments involves creating a *backstory* which outlines some requirement for information [1]. The backstory is a concise summary of a single information need and is somewhat different from the usual use of the word *topic* within the IR community. A topic may contain one or more *facets* that describe small differences of a more general concept, and is a neutral statement of need. A backstory is formulated as a single-faceted need – worded in a manner that makes the search goal precise, and typically couched in personal terms so as to engage with the crowdworker, often, for example, commencing with "You" or "Your". Crowdworkers are then requested to read the backstory and formulate the query that they would issue to a search provider to address that need. This approach was used to generate the UQV100 queries, [2], and the recent community question answering logs of Biega et al. [8].

CHIIR '21, March 14–19, 2021, Canberra, ACT, Australia
© 2021 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-8055-3/21/03.
https://doi.org/10.1145/3406522.3446054

[1] https://rodgerbenham.github.io/ccnews-explorer; https://youtu.be/DS79s-8-4bU

Table 1: Trimmed data for a user typing the word `apple` (topic 88). This example shows a typographical error, and then a correction. Note the presence of key rollover in actions 9 to 12, when the user has two keys depressed at the same time. Times are in milliseconds.

Action	Time Delta	Key- Value	Query Box
7	144	▼ Key-E	`app`
8	72	▲ Key-E	`appe`
9	34	▼ Key-K	`appe`
10	2	▼ Key-L	`appek`
11	46	▲ Key-K	`appekl`
12	0	▲ Key-L	`appekl`
13	361	▼ Backspace	`appekl`
14	74	▲ Backspace	`appek`

Most recently, Mackenzie et al. [16] released CC-News-En, an English news resource that includes 10,437 user query variations over 173 news topics. The CC-News-En collection was constructed via a slightly different query elicitation approach, the use of an article *summary*. The idea here is to provide the crowdworker with a brief outline of a news story – as might be seen in the scrolling "ticker" at the bottom of a TV screen while another story is being presented in full, or as might be headlined on the radio on the half-hour as a teaser for the in-depth bulletin thirty minutes later – and have them take on the role of a person wanting to find the corresponding full story. In this approach, a range of factors can be varied as the queries are being collected. In particular, summaries of three different lengths (a title sentence, a short summary, and a long summary) varied the density of information given to the workers [17]; and the summaries were either shown to workers as *images of text* or presented as *spoken audio renditions*. The aim was to vary the query formulation process and to guard against *availability effects*, where workers may be biased to use the same terms as present in the summary. Further details are provided by Mackenzie et al. [16].

Crowdworker Data. Our visualization tool makes use of crowdworker data from the CC-News-En collection. Of particular interest is the `cc-news-query-keystroke.tsv` file, which contains *keystroke-level* data for a subset of 10,089 UQVs[2], collected using the Amazon Mechanical Turk interface, with keystrokes in the query input box captured using the Javascript events `onkeyup` and `onkeydown`. At the beginning of each query sequence, the time taken since page load was recorded. Then, the interval between each subsequent `keydown` and `keyup` event was recorded, along with the Javascript keycode and the current contents of the query box. Table 1 shows a trimmed representation of a partially typed query, illustrating the data that was collected.

The logs also contain metadata elements, including:

- Summary format: one of $\{\textit{text}, \textit{audio}\} \times \{\textit{title}, \textit{short}, \textit{long}\}$, noting how the summary was presented to that worker;
- Device Type: whether the worker completed the task on a desktop, tablet, or mobile device; and
- Browser Type: the web browser the worker used for that task.

The keystroke actions and metadata require 90 MiB in total.

Common Crawl News Query Explorer. To interactively explore the query variations and their key-stroke data we developed a responsive web application, thereby avoiding the need for complex hardware or sophisticated API support.

Implementation Details. The display tool uses React 16[3] as a Javascript framework to structure the user interface, selected for its ability to efficiently update the DOM inside of UI components without forcing re-rendering of other parts of the viewport – a strength that is emphasized when multiple queries are animated concurrently. To facilitate concurrent rendering (topic 1 has 115 queries with keystroke data), we use the `requestAnimationFrame()` API. It allows the browser viewport to be rendered as quickly as the hardware and browser can support (similar to a *game loop*), including dropping frames when the UI cannot be updated in time.[4] The website is hosted as a static webpage on GitHub Pages (with no back-end, thereby avoiding hosting fees), exploiting the fact that the user does not mutate the data being displayed.

Pre-Processing. To facilitate *lazy loading* and improve the user experience, the keystroke data (see Table 1) was pre-processed, and the initial 90 MiB partitioned into independent files. These are asynchronously fetched when requested by the interface. To further reduce the bandwidth requirement, the Javascript `keydown` events were filtered out – they do not change the contents of the query box. The animation data is reduced to a pipe-delimited string, representing instances of waiting time for the next action, and the query box instance contents at that point of time. To integrate into the React Javascript framework and avoid the computational overheads of conversion into Javascript objects in the front-end, the data is represented in JSON and dynamically loaded into variables via HTTP requests. After pre-processing, the total data size was reduced to 34 MiB, with each topic using between 108 KiB and 348 KiB of disk space.

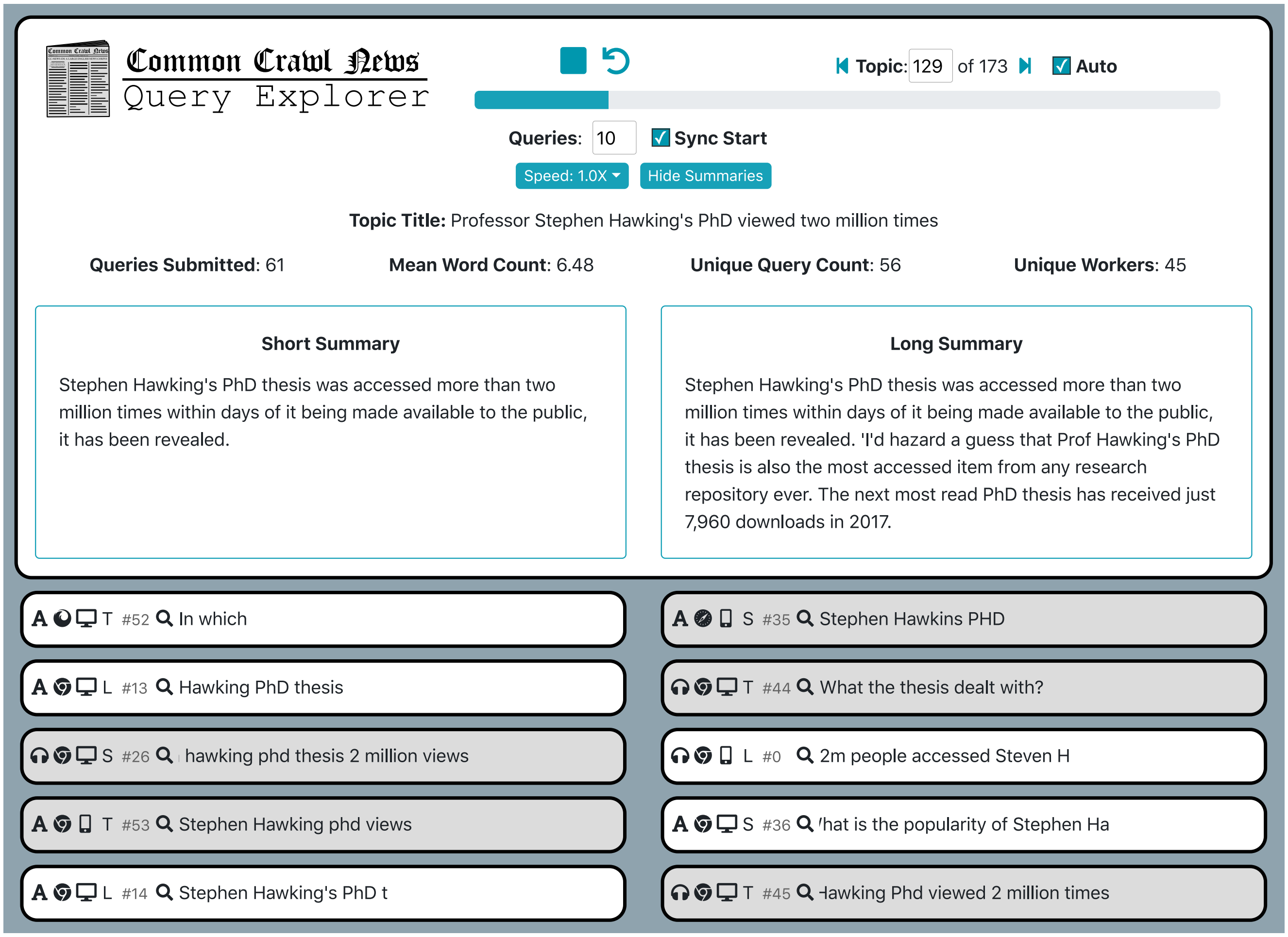
Interface. Figure 1 shows the interface. The title of the current topic is shown at the top of the page, with playback controls nearby. In the left-most portion of the controls, the user can start, pause, and restart the playback. On the right, the topic selector allows changes of topic, both by number or by next/back arrows. When "Auto" is selected the next topic will be displayed after playback has finished for the present one. Below the topic selector is a progress bar, which shows how much time is remaining to finish rendering the typing data of the slowest query on display. Beneath the progress bar (or in the right-most part of the control panel on a wider screen) the user can specify how many queries they would like to view, whether their input times should begin concurrently with "Sync Start", the current playback speed (with options of 0.2×, 0.5×, 1×, 2×, 5×), and whether the topic summaries should be displayed.

The set of workers' queries determined by those settings appear underneath that top panel. Each query box has a metadata component on the left to describe that query's unique identifier context

[2] Available from http://go.unimelb.edu.au/u3nj

[3] React: https://reactjs.org/

[4] `requestAnimationFrame` MDN web docs: https://developer.mozilla.org/en-US/docs/Web/API/window/requestAnimationFrame



Common Crawl News
Query Explorer

■ ↺ ⏮ Topic: 129 of 173 ⏭ ☑ Auto

Queries: 10 ☑ Sync Start

Speed: 1.0X ▾ Hide Summaries

Topic Title: Professor Stephen Hawking's PhD viewed two million times

Queries Submitted: 61 **Mean Word Count**: 6.48 **Unique Query Count**: 56 **Unique Workers**: 45

Short Summary

Stephen Hawking's PhD thesis was accessed more than two million times within days of it being made available to the public, it has been revealed.

Long Summary

Stephen Hawking's PhD thesis was accessed more than two million times within days of it being made available to the public, it has been revealed. 'I'd hazard a guess that Prof Hawking's PhD thesis is also the most accessed item from any research repository ever. The next most read PhD thesis has received just 7,960 downloads in 2017.

A ◉ 🖥 T #52 🔍 In which

A ◉ 🖥 L #13 🔍 Hawking PhD thesis

🎧 ◉ 🖥 S #26 🔍 hawking phd thesis 2 million views

A ◉ 📱 T #53 🔍 Stephen Hawking phd views

A ◉ 🖥 L #14 🔍 Stephen Hawking's PhD t

A ◉ 📱 S #35 🔍 Stephen Hawkins PHD

🎧 ◉ 🖥 T #44 🔍 What the thesis dealt with?

🎧 ◉ 📱 L #0 🔍 2m people accessed Steven H

A ◉ 🖥 S #36 🔍 'hat is the popularity of Stephen Ha

🎧 ◉ 🖥 T #45 🔍 -lawking Phd viewed 2 million times

Figure 1: The *Common Crawl News Query Explorer* interface, showing the partial query playback of ten queries for topic 129 of the CC-News-En collection. The icons at the left side of each query box are described in Table 2. Query boxes are shaded once the input query has reached its final state. The control panel at the top of the page controls playback state and speed, and describes the selected topic, including its textual summary and some basic statistics.

Table 2: Icons describing the worker context for each query.

Metadata	Icons	Descriptions
Modality	🎧, A	Audio, Text
Browser	◉, ◉, e, ◉	Chrome, Firefox, Edge, Safari
Device	🖥, ▯, ▯	PC, Mobile, Tablet
Summary	T, S, L	Title, Short, Long

(see Table 2 for a legend, and also revealed on mouse hover); to the right, the query text is animated into the viewport. When each query animation completes, its box turns gray.

3 QUALITATIVE ANALYSIS

To accompany the *Common Crawl News Query Explorer*, we now outline several observations made using the new interface. These anecdotes show that the final query submitted by the user is only a fraction of the information available via more detailed search engine instrumentation, with a wide variety of formulation techniques observed during query construction. We report the topic and unique query identifier for each example as a topic-#query pair, to allow our observations to be recreated by other users of the exploration tool (in some cases, after having increased the number of queries being displayed).

Input Techniques. Recent work has explored user typing behavior both in terms of typing tests [12], and in the context of web search [18]. Here, we present a snapshot of interesting user typing behaviors. Firstly, we found that desktop users exhibited behavior in line with typical keyboard use, including the use of *N*-key rollover, which was recently found to be prevalent in stronger typists [12]. Secondly, we observed that desktop workers used keyboard shortcuts (such as `Ctrl+x` and `Ctrl+v`) to rearrange query terms. Also, a subset of users opted to use the `CapsLock` key to upper-case their characters (instead of the more common `Shift+key` method).

Mobile and tablet workers often made use of automatic term suggestion and spelling correction tools which are now present on most virtual keyboards. These particular cases are characterized by either several characters or even whole terms appearing in a single key-press event. Consider the following example (42-#57):

```
... on British ⟶ ... on British airways
```

Here, only a single keystroke was observed between the two states, indicating that the worker accepted a suggestion for the term "airways" that been provided in the context of their query text. Similarly, mobile workers also used automatic dictation tools – characterized by a series of terms appearing at once without interruption. Consider the following example (111-#0), which captures the first three actions from a worker:

```
First ⟶ First casino ⟶ First casino in
```

We conjecture that the mobile device would not automatically suggest the term "First" without prior context, indicating to us that the user is dictating to the device through an audio channel.

Mishearing Audio Summaries. We observed that users listening to audio topics occasionally misheard them, resulting in unusual terms in their subsequent query. For example, topic 31 quotes President Barack Obama discussing a cyberattack; one worker (31-#43) entered the query:

```
A bomber to carry out huge cyberattack,
```

mistaking the name "Obama" with "a bomber". We observed three similar instances in topic 88, which describes how *Lattice Data*, a "dark data" AI company, was acquired by Apple:

```
... acquired larry's data ...      (88-#25)
... duck data company              (88-#27)
... acquiring latex data           (88-#10)
```

Again, we presume these particular users misheard "lattice" as "larry's" and "latex"; and, perhaps more unfortunately, "dark" as "duck". Other researchers have suggested that speaking style can have a large effect on the utility of spoken retrieval systems [10, 20], and it would be interesting to explore how mishearing audio affects both user and system performance in such contexts.

Pondering, or Validating? Our tool also allows user dwell time to be examined. While most users have some initial dwell time while reading or listening to the summary and formulating a query, there are also several cases where users have long dwell times *during* query generation. For example (144-#11):

```
... Steven Hawkins ⟶ [114 secs] ⟶ ... Stephen Hawkins
```

While the data collected was not granular enough to determine what happened during this almost two-minute pause, we assume that the worker may have used some external resource to validate their spelling. However, in checking "Stephen", they did not notice that "Hawking" was also incorrectly typed as "Hawkins". Another interesting case showed a worker seemingly second-guessing themselves (144-#20). In this example, right arrows represent insertion, left arrows represent deletion, and for brevity, the individual keystrokes are not shown:

```
[dwell: 22 seconds]
 ⟶ Stephen Hawking on A
 ⟵ Stephen Hawking on
 ⟶ Stephen Hawking on artificial inte
 ⟵ Stephen Hawking on artif
 ⟶ Stephen Hawking on artifcial
 ⟵ Stephen Hawking on
 ⟶ Stephen Hawking on artificial intelligence
[dwell: 31 seconds]
 ⟶ Stephen Hawking on control of artificial ...
```

Again, while the reason behind this behavior was not instrumented, it suggests that the user is both formulating and typing the query at the same time, resulting in several interesting edits.

Observing Crafty Crowdworkers. Some of the queries in the CC-News-En dataset were submitted by *crafty crowdworkers* [16], who used automatic tooling to game the task. One such example shows a crowdworker pasting external results from a web search engine, and subsequently deleting extraneous text which was accidentally copied across (166-#47):

```
[dwell: 77 seconds]
 ⟶ Search Results Web results French raid in ...
 ⟵ French raid in Mali ...
```

In future work, it would be interesting to see if crowdworker keystroke data could be used to detect these types of behavior.

4 CONCLUSION

We have described the *Common Crawl News Query Explorer*, an interactive browser-based tool for visualizing user querying behavior in the context of the recent CC-News-En collection [16]. Our tool provides a fascinating (indeed, somewhat addictive) way for search practitioners to visualize how crowdworkers formulate queries, by animating associated keystroke-level telemetry data. As well as displaying the querying behavior of these crowdworkers, our tool also enables insights and comparisons to be made between the various input modes, device types, and task specifics such as how the information need was conveyed to each worker. To accompany the description of the tool, we have provided a set of initial observations which highlight the diversity of approaches deployed by users when formulating queries.

In future work, we plan to extend these initial observations to construct a detailed model of query formulation behavior. The potential to create a comprehensive taxonomy of user query behavior is an exciting opportunity – and may advance our understanding of important search techniques such as query auto-completion and query rewriting.

Acknowledgments. This work was partially supported by Australian Research Council Grant DP190101113. The first author was supported by an RMIT VCPS.

REFERENCES

[1] P. Bailey, A. Moffat, F. Scholer, and P. Thomas. User variability and IR system evaluation. In *Proc. SIGIR*, pages 625–634, 2015.

[2] P. Bailey, A. Moffat, F. Scholer, and P. Thomas. UQV100: A test collection with query variability. In *Proc. SIGIR*, pages 725–728, 2016.

[3] P. Bailey, A. Moffat, F. Scholer, and P. Thomas. Retrieval consistency in the presence of query variations. In *Proc. SIGIR*, pages 395–404, 2017.

[4] N. J. Belkin, C. Cool, W. B. Croft, and J. P. Callan. The effect of multiple query variations on information retrieval system performance. In *Proc. SIGIR*, pages 339–346, 1993.

[5] R. Benham and J. S. Culpepper. Risk-reward trade-offs in rank fusion. In *Proc. Aust. Doc. Comp. Symp.*, pages 1.1–1.8, 2017.

[6] R. Benham, J. S. Culpepper, L. Gallagher, X. Lu, and J. Mackenzie. Towards efficient and effective query variant generation. In *Proc. DESIRES*, pages 62–67, 2018.

[7] R. Benham, J. Mackenzie, A. Moffat, and J. S. Culpepper. Boosting search performance using query variations. *ACM Trans. Inf. Sys.*, 37(4):41.1–41.25, 2019.

[8] A. J. Biega, J. Schmidt, and R. S. Roy. Towards query logs for privacy studies: On deriving search queries from questions. In *Proc. ECIR*, pages 110–117, 2020.

[9] M. Catena and N. Tonellotto. Multiple query processing via logic function factoring. In *Proc. SIGIR*, pages 937–940, 2019.

[10] A. Chuklin, A. Severyn, J. R. Trippas, E. Alfonseca, H. Silen, and D. Spina. Using audio transformations to improve comprehension in voice question answering. In *Proc. CLEF*, pages 164–170, 2019.

[11] J. S. Culpepper. Single query optimisation is the root of all evil. In *Proc. DESIRES*, page 100, 2018.

[12] V. Dhakal, A. M. Feit, P. O. Kristensson, and A. Oulasvirta. Observations on typing from 136 million keystrokes. In *Proc. CHI*, pages 646.1–646.12, 2018.

[13] O. Hoeber, X.-D. Yang, and Y. Yao. Visualization support for interactive query refinement. In *Proc. WI*, pages 657–665, 2005.

[14] H. Li, H. Scells, and G. Zuccon. Systematic review automation tools for end-to-end query formulation. In *Proc. SIGIR*, pages 2141–2144, 2020.

[15] B. Liu, N. Craswell, X. Lu, O. Kurland, and J. S. Culpepper. A comparative analysis of human and automatic query variants. In *Proc. ICTIR*, pages 47–50, 2019.

[16] J. Mackenzie, R. Benham, M. Petri, J. R. Trippas, J. S. Culpepper, and A. Moffat. CC-News-En: A large English news corpus. In *Proc. CIKM*, pages 3077–3084, 2020.

[17] D. Maxwell, L. Azzopardi, and Y. Moshfeghi. A study of snippet length and informativeness: Behaviour, performance and user experience. In *Proc. SIGIR*, pages 135–144, 2017.

[18] K. Ong, K. Järvelin, M. Sanderson, and F. Scholer. QWERTY: The effects of typing on web search behavior. In *Proc. CHIIR*, pages 281–284, 2018.

[19] H. Scells and G. Zuccon. searchrefiner: A query visualisation and understanding tool for systematic reviews. In *Proc. CIKM*, pages 1939–1942, 2018.

[20] P. Thomas, M. Czerwinski, D. McDuff, N. Craswell, and G. Mark. Style and alignment in information-seeking conversation. In *Proc. CHIIR*, pages 42–51, 2018.