Does More Context Help? Effects of Context Window and Application Source on Retrieval Performance

TUNG VUONG, University of Helsinki, Finland
SALVATORE ANDOLINA, University of Palermo, Italy and University of Helsinki, Finland
GIULIO JACUCCI, University of Helsinki, Finland
TUUKKA RUOTSALO, University of Helsinki, Finland and University of Copenhagen, Denmark

We study the effect of contextual information obtained from a user's digital trace on Web search performance. Contextual information is modeled using Dirichlet-Hawkes processes and used in augmenting Web search queries. The context is captured by monitoring all naturally occurring user behavior using continuous 24/7 recordings of the screen and associating the context with the queries issued by the users. We report a field study in which 13 participants installed a screen recording and digital activity monitoring system on their laptops for 14 days, resulting in data on all Web search queries and the associated context data. A query augmentation model was built to expand the original query with semantically related terms. The effects of context window and source were determined by training context models with temporally varying context windows and varying application sources. The context models were then utilized to re-rank the query augmentation model. We evaluate the context models by using the Web document rankings of the original query as a control condition compared against various experimental conditions: 1) a search context condition in which the context was sourced from search history, 2) a non-search context condition in which the context was sourced from all interactions excluding search history, 3) a comprehensive context condition in which the context was sourced from both search and non-search histories, and 4) an application-specific condition in which the context was sourced from interaction histories captured on a specific application type. Our results indicated that incorporating more contextual information significantly improved Web search rankings as measured by the positions of the documents on which users clicked in the search result pages. The effects and importance of different context windows and application sources, along with different query types are analyzed, and their impact on Web search performance is discussed.

CCS Concepts: • **Information systems** → **Query reformulation**.

Additional Key Words and Phrases: Web search, contextual information, digital user behavior, query augmentation, context window, application source

ACM Reference Format:
Tung Vuong, Salvatore Andolina, Giulio Jacucci, and Tuukka Ruotsalo. 2021. Does More Context Help? Effects of Context Window and Application Source on Retrieval Performance. *ACM Transactions on Information Systems* 1, 1, Article 1 (January 2021), 41 pages. https://doi.org/10.1145/3474055

Authors' addresses: Tung Vuong, vuong@cs.helsinki.fi, University of Helsinki, , Helsinki, , , Finland; Salvatore Andolina, University of Palermo, Palermo, Italy, University of Helsinki, Helsinki, Finland, salvatore.andolina@unipa.it; Giulio Jacucci, giulio.jacucci@helsinki.fi, University of Helsinki, Helsinki, Finland; Tuukka Ruotsalo, tuukka.ruotsalo@helsinki.fi, University of Helsinki, Helsinki, Finland, University of Copenhagen, Copenhagen, Denmark.

1046-8188/2021/1-ART1 $15.00
https://doi.org/10.1145/3474055

Original query	*Contextualized query*	*Task context*
rdm	rdm **mac**	Looking up information on a tool used for Mac OSX display resolution settings.
terrace house opening new doors	terrace house opening new doors **netflix**	Searching for updated information on Netflix about the movie.
glossier cloud paint puff	glossier cloud paint puff **mymallbox**	Searching for information on a cosmetic product on the commercial website mymallbox

Table 1. Examples of how users' queries were contextualized. Each original query was augmented as a contextualized query with an additional related term that could be inferred from digital user behaviors preceding the search. A new term added to the original query is denoted in bold.

1 INTRODUCTION

According to Internet Live Stats[1], billions of Web searches are executed online each year worldwide. Web search provides access to the vast amount of information on the Web, but also supports addressing a broad range of information-intensive tasks [9]. Often, the query submitted by the user may be short and is written using the user's own vocabulary. The Web documents, however, may be written by different individuals with different vocabulary and styling [43]. This raises a fundamental problem of term mismatch in information retrieval that can negatively affect the results of Web searches [14]. Typical examples are shown in Table 1. For example, the query *"rdm"* is short, ambiguous, and does not specify a detailed description of information needs that would enable effective retrieval.

To overcome this problem, many different techniques have been proposed to improve the retrieval performance, including query expansion, which selects related terms from top-ranked documents in the initial retrieval results to expand the query [12]; query disambiguation, which identifies ambiguous terms and expands the query with synonyms [44]; and query diversification, which presents many different interpretations of the query up front and lets the user select the correct one [34]. All of these require user effort and use external data but may still not lead to successful queries. An alternative approach is to use the rich digital context of the user to try to detect the user's information needs and augment the query with contextually related terms. The context provides a rich source of information about the tasks in which the user is involved, and this information can be leveraged to augment the queries [30, 53].

The use of context in query augmentation has been extensively studied [27, 69]. Many types of context data have been considered, and numerous methods have been proposed through which contextual information has been used [16, 21]. The main approach has been to construct the models from observed past user behaviors, which are often sourced from the search engine interaction logs [21, 22, 26] or Web browsing data [3, 27, 32]. The context has been used to redefine the vocabulary in which query expansion terms were generated. For instance, Eickhoff et al. [21] considered search engine result pages of the prior query as context; the signal value was the set of terms that the user paid attention to on the pages. Then, the candidate terms for query expansion were reranked according to the semantic correlation to those contextual terms. However, Web searches are often conducted as part of a more general task [64], and therefore considering search history as the only source of context may be a factor limiting the effectiveness of query augmentation. Another approach is taking all the desktop data (documents stored on the computer) as context [16]. The authors first identified the set of terms that were closely related to the current query as candidates. Then, they restricted the candidate terms to only those appearing on the desktop. More recent

[1]http://www.internetlivestats.com/

research has also utilized data from other sources that involve a richer context. Singh et al. [56] logged user behavioral signals, including clicks and page visits, on a real-world e-commerce site to elicit user query intent. Li et al. [41] considered user context based on recently read emails. Tan et al. [59] collected recently opened documents as context for recommendations. All of these methods aimed at exploiting as much additional context as possible; however, it is still unclear if and to what extent leveraging more context sources would lead to improved search ranking.

Another important factor in personalizing search is the amount of behavioral information considered to construct the context model [7]. Most studies often use all available context as a whole without distinguishing which parts are useful. Although positive results have been reported, collecting more information may not always be necessary. An alternative approach is dividing the whole context into partitions by time. Such work includes [54, 61], which associate the current query with different amount (time periods) of history (e.g., queries or browsed Web pages) for personalized search. A more recent approach [7] detected boundaries of search sessions first, so that activities within the same session could be used to train the model. For this purpose, an early work [65] determined a search session demarcated by 30 minutes of user inactivity. Rather than emphasizing such absolute time, Koskela et al. [35] modelled the context in that most recent activity were more relevant than activities conducted some time ago by introducing a decay function. However, choosing the decay function was dependent on the experimental settings and test data used. Earlier attempts to address this challenge leveraged different representations of context including recent interactions (short-term), historical browsing contexts (long-term), or a combination of both short- and long-term contexts to personalize search results [7]. Nevertheless, the investigation of the impact of more contextual information on query augmentation has received less attention. We address this shortcoming with the research presented here by studying the relationships between various context sizes by time thresholds (10 minutes, 1 hour, and 1 day) and investigating their effects on modeling and query augmentation. We also study the effect of the context source, as the utility of contextual data may be associated with different applications, tasks, and search intents.

We model the context "comprehensively", which considers not only searching and general browsing behaviors, but also a variety of other applications. That is, we account for all user-generated content and the textual information available in all types of applications that were used preceding the search. This context contains information about a user's search intent and preference that can be used in re-ranking Web search results according to user needs. For example, Figure 1 shows a sequence of contextual information captured from a user's digital behavior and the use of that information in query augmentation. A user is engaged in a Web design task and is using CSS styling language. The context here is typically the content that was processed and produced by the user before the query, including history of previous searches, Web browsing history, past email conversations, or any text written by the user into the code editing software. The user's intent was to look for a CSS code implementing the animation effect to support the task. Conventional query augmentation would expand the original query "candle animation html" with a related term such as "flame". Here, the context model reranked the prediction terms after considering contextual information, such that the term "CSS" that was contextually related would be ranked higher. The original query was rewritten as "candle animation html css" so that it would better convey the user's search intent and lead to an improved search ranking.

By considering both the source of context and different time windows, we seek to answer the following research questions (RQs):

- **RQ1:** Does comprehensive use of context improve retrieval performance?
- **RQ2:** How does the source of contextual information affect Web search performance?

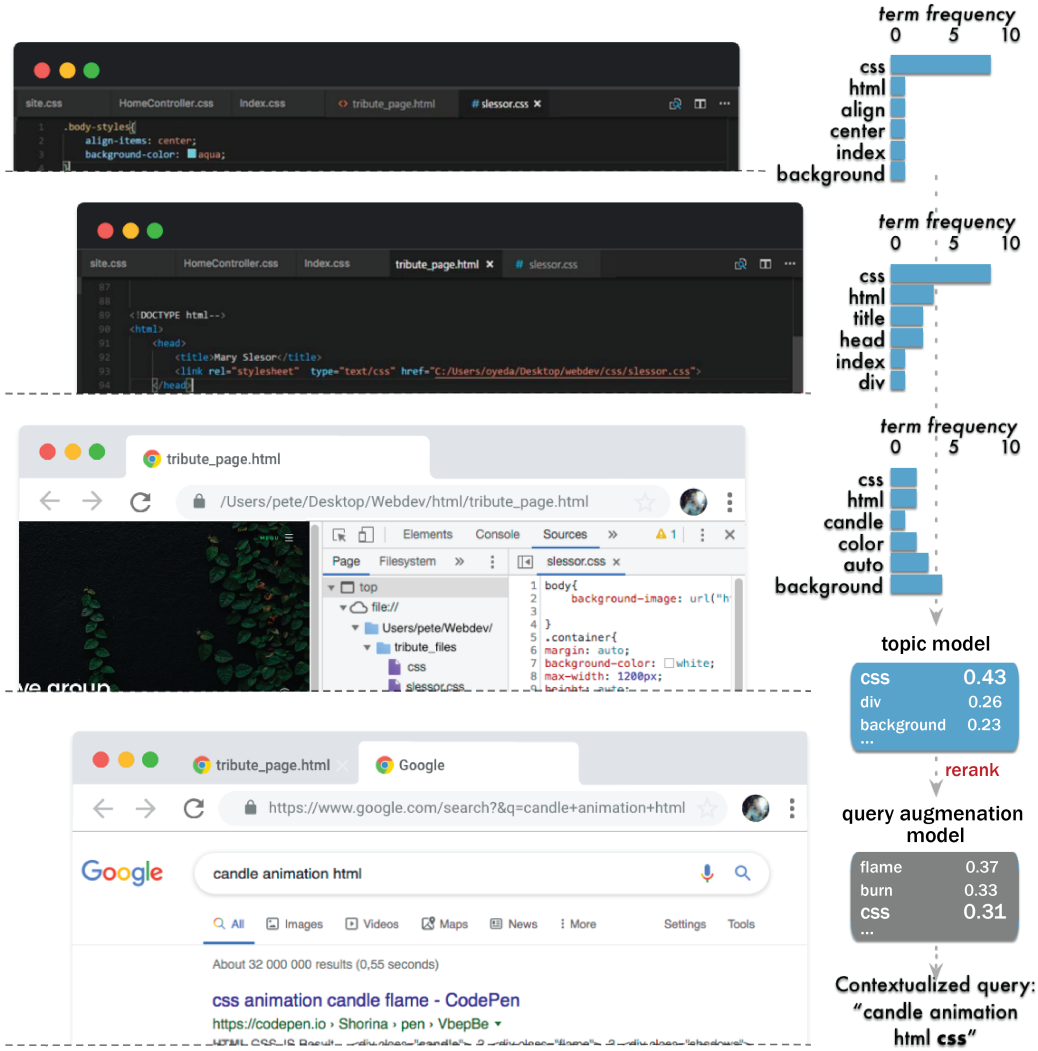


Fig. 1. An example of contextual query augmentation. The first three screen frames recorded 10 minutes before the search, show documents read by the user, and the bottom frame is the query inputted to the search engine. The context includes opened documents on other applications, and these were used to train the predictive model. The query augmentation model was re-ranked using the context model. The contextually related term "CSS" was ranked higher and used to expand the original query "candle animation html"

To answer the research questions, we report an in-the-wild data collection study monitoring the computer screens of volunteer users over the course of 14 days. The resulting data includes all content produced by the user and to which the user is exposed to across applications. User digital activities captured from a wide range of applications were classified into different sources and used to train the context models for query augmentation. Varying windows of context were also used when training the models, including digital activities that occurred within a time window of 10 minutes, 1 hour, and 1 day prior to Web searching.

We further report the results of two data-analysis experiments with respect to the two RQs. In **Experiment 1**, Web document rankings of the original query were used as control conditions and compared with context models. A query expansion model was built using the content of top-ranked search results in response to the original query. Then, the query expansion model was used to augment the original query. To investigate whether more context helps to improve query augmentation, the sources of context information and the sizes of context used to construct the context models for experimental conditions were manipulated. We tested the control condition and the query augmentation condition against various experimental conditions: 1) a search context condition in which only the search history or screen captures of the search systems were leveraged, 2) a non-search context condition in which all the sources excluding the search history were leveraged, and 3) a comprehensive context condition in which all sources of context (both search and non-search contexts) were leveraged. We evaluated the retrieval quality of the queries in different conditions using click metrics: Mean Average Precision for each query computed as the average precision of retrieving the documents that were clicked by users at top K, and Mean First Relevance for each user computed as the average ranks of the last-clicked documents in the search results list. In **Experiment 2**, we analyzed the effect of different sources of contextual information on retrieval performance. We first clustered computer applications that had similar functions into various sources. Similar to Experiment 1, we varied information from the application sources to build the context models for application-specific context conditions. We also compared the retrieval quality of each application-specific context condition against the control condition and the query augmentation condition using the same measures. Moreover, we characterized the effect of contextual query augmentation with regard to different search intents (navigational, transactional, informational) in both experiments. Our findings reveal the following:

- In general, the user's search history is a rich source of contextual information for query augmentation. However, when the search history is short or contains only a little information about the user's task-related preferences (cold start or new task), the search history can be complemented with other sources of contextual information.
- Contextual information from any type of application prior to the search is useful in augmenting queries. Therefore, contextual information should not be limited to only user interactions with the search system itself, but user context should also be modelled by considering other application sources, for example, email conversations or text observed when the user is producing or consuming content (writing or reading).
- Queries with informational intent can be successfully augmented by using contextual information, whereas queries with transactional or navigational intent are limited in how they benefit from contextual information. Informational queries are associated with re-visitation behavior; for example, searching for programming knowledge often lands on a site visited frequently or that is about a certain task (e.g., stackoverflow on Javascript programming). In contrast, queries with transactional intent are more session specific and thus are not necessarily influenced by the context. For example, the process of searching for airlines, the user may end up visiting sites of different airlines across search sessions and augmenting the query with context focusing on airlines other than the one in focus may be limited. Moreover, queries with navigational intent are better specified. In other words, the users know what they are searching for and can write successful queries up front. Therefore, contextual information has limited benefits for augmenting navigational queries.
- Different sources of contextual information may be suited to different queries. For queries with informational intent (checking facts, looking up information), recent interaction history sourced from general Web browsing, textual documents, emails, and instant messages contains

relevant context, and the model performs better than query augmentation using search history. On the other hand, to better understand user information needs in the case of transactional intent, the context sourced from a full day of user learning activities, interactions with office productivity applications or static Web pages, and electronic purchasing behavior becomes more useful.

2 BACKGROUND

In this section, we begin by giving an overview of research efforts aimed at the use of contextual information in Web search applications. Then, we review previous work on the roles of context windows and application sources in query augmentation, highlighting our contributions with respect to prior work.

2.1 Contextual information in Web search

There has been increased interest in the information retrieval community in modeling user contexts to improve various aspects of search, such as document ranking and query suggestion [36, 70]. A large body of research has explored different types of context including searching and browsing activities, and built predictive models to improve Web search performance [12, 27, 49, 70]. Xu and Croft [70] viewed terms in top-retrieved documents as context of the query and used for query expansion. A problem with this approach is that some of the expansion terms extracted from top-retrieved documents are irrelevant to the user's underlying context and thus may hurt the retrieval performance. To address this problem, Shen et al. [55] considered the user's browsing history to capture the current query context. A context model was then used to improve the query expansion. Instead of using the immediate browsing history before the search, Chirita et al. [16] considered the entire document collection stored on the user's computer as the contextual environment. However, a text document may contain many topics, and the user might only be interested in some of them. Chen et al. [15] and Buscher et al. [11], on the other hand, focused on implicit feedback on the document level, more specifically, the document parts that the user looked at, by utilizing the eye-tracking approach. This way, the viewed document parts can better represent the user's short-term context and can be used to improve the quality of retrieval by query expansion and reranking. Further studies aimed at incorporating user context for improved query suggestion and document ranking. For example, Cao et al. [13] mined recurring sequential patterns from search sessions, and this information was used to train the model for context-aware query suggestion. Mitra [47] studied session context with a distributed representation of queries and reformulations and used the learned embeddings for query prediction tasks. Zhang and Jones [72] modelled contexts comprising queries and clicks within search sessions for query suggestion, URL recommendation, and document re-ranking. All of these researchers gathered and included different kinds of context or parts of user context for personalizing search, whereas we combined all contextual signals, and as the main goal of our study, we compared the performances of the models constructed using different contexts and studied their effects on retrieval performance.

Another study related to ours is [54], which proposed a method for context-aware reranking. The current query was augmented by using contextual information and then fitted into language models for retrieval. The basic idea was to promote the context as the preceding queries and recently viewed documents within the same session. The authors evaluated their models using a small amount of session data created upon TREC collection [54]. However, in reality, user sessions for Web searches are more complex, often conducted as part of a general work task [53, 64]. Typically, the existing method assumes that the context is present within search sessions [27]. The assumption weakens for complex information needs, for which it is possible that the context has not been

previously seen in the search logs [10]. Therefore, considering only search history to model context may limit the effectiveness of query augmentation.

Unlike previous studies, we do not restrict the context to a specific type of system but use all user interactions and behaviors that naturally occur on the computer. Therefore, the context analyzed in our study is rich and includes more extensive sources that span a variety of applications.

2.2 Roles of context windows

Studies on query augmentation are not limited to proposing a new modeling approach; researchers also seek to obtain richer contextual information useful in improving Web searches [7, 32, 47]. Cao et al. [13] demonstrated how variable sizes of contextual information affected query suggestion. Recent work has shown that collecting more contextual information can improve the performance of query augmentation [67, 73].

Constructing the context size generally depends on the timespan of user behavior prior to the search [7, 31, 49]. A search session has also been widely considered as a context for personalization. A window of 30 minutes has already been used to demarcate sessions [51, 52], although it may also range from 5 minutes to several hours depending on the study settings [60]. White et al. [65] studied the utility of various kinds of contextual information based on how effectively they can be used to predict future user interest. In their study, the kind of context was determined by the number of Web pages visited by the user preceding the current page; for example, five pages visited prior to the current page were considered as the immediate session-based interaction context, while historical context included all pages visited before. The models were trained using this contextual evidence to predict the future page the user would open. The best accuracy was obtained when contextual information about the current session was used to predict interests in the following hour. However, their study focused on predicting user interest at different times in the future but did not utilize the contextual information for query augmentation.

Following the previous work, we explore the effectiveness of the context model trained based on different temporal views of interaction history on various applications and systems. Accordingly, we used three windows of context: 10 minutes, 1 hour, and 1 day. User behaviors (e.g., previous queries and page visits) 10 minutes to 1 hour prior to the current query have been used to generate short-term profiles [66], whereas a full day of behaviors can represent long-term interests [7].

Both long- and short-term behaviors have been shown to provide benefits for search personalization models [7]. Short-term behavior during the current search session has been used for search results ranking [7] and predicting future search interests [66]. Teevan et al. [61] found that the performance of their personalization algorithm improved as more information about the user's interests became available. Long-term behavior has also been used to personalize search result rankings by building long-term models of search interests [7], specifically including the use of documents that have been previously clicked, or prior queries suggesting a pursuit of similar search intent. However, little effort has been focused on exploring the impact of different sizes of histories, for example, keeping one of either short-term history or long-term history at a time and studying its effects. Meanwhile, we delve deeper into this aspect by performing experiments on the models in different conditions with different context sizes, such as a condition with a short-term history and a condition with a long-term history, and we varied the sources used to construct the model. In addition, most previous work is based mainly on the analysis of search engine logs in which the presence or absence of a term on a Web document or search engine results page (SERP) can be regarded as a signal. However, in the real-world scenario, the prior queries may not always be a relevant source to model search context. For example, the query "t-rex" was about a dinosaur, but the intent of the query was to search for a bike as the user was in the middle of a discussion with friends about sports bikes on an instant messaging application, but the search history might

not contain information about the conversation. This problem can be alleviated by offering query suggestions that guide the searcher toward popular queries frequently issued by other users [23]. It is, however, often unclear how relevant such suggestions are for the individual user. Therefore, in those cases, the search intent can only be identified through a more comprehensive context beyond the search system.

Contrary to the limited context sourced from search history, our screen recording contains a large amount of information covering search activity, and other types of activities (e.g., while writing documents or accessing intranet information) that conventional logging mechanisms can not obtain. Rather than modeling context based on the search history alone, we have developed methods to leverage a more comprehensive context from a wider range of applications to improve retrieval. Such an approach allows us to study the impact on search performance when more contextual signals considered. As part of our study, we also investigate the effect of different windows of context on model performance.

2.3 Roles of application sources

While the timespan of the behavioral information used for context construction is important, another critical aspect is the source of information used to train the model [10, 27, 28, 32]. Empirical findings by Belkin et al. [6] suggest that eliciting a variety of information needs from multiple sources of evidence concerning the relevance of documents to a query is likely to improve retrieval. Kelly et al. [33] performed user studies investigating another source of context in the search process and selecting useful sources of terms for query expansion. The authors focused on using a document-independent technique for eliciting feedback from users about their information problems, for example, asking them to report topics of their interest directly in a feedback document. They used the report to find useful terms for query expansion. However, they did not use an automatic technique to elicit information needs from a user's past behaviors, but rather relied on background and contextual information that the users provided themselves. Recent work has proposed augmenting queries for sponsored searches in which the current query was substituted by a query containing terms from the source of the advertising content, thereby increasing the number of Web ads available to users [27, 28, 45, 56]. Some studies have also focused on query augmentation for email search [38, 41]. User information needs were modelled based on the source of email conversations, and models were built to predict and augment the query [41]. When such metadata is absent, search systems will rely on different sources to model the user context; for example, queries and clicks from Web searches, user-generated content on micro-blogging posts [20, 42, 46], multimedia documents (e.g., images, videos) [48], hyperlinked Web pages [5, 37], and click-through data [5, 8, 17].

Different sources of information have been leveraged in these studies with the aim of reducing query ambiguity and vagueness and elucidating the actual intent behind the query. A less vague query has the potential to produce more relevant search results for search applications. However, no work so far has investigated how much improvement can be achieved by leveraging these specialized sources for general Web searches.

Context, application sources, and information retrieval are highly connected, as shown in many studies [14]. Some reviews and meta-analyses have tried to compare the various techniques used for query augmentation [14, 40]. However, they have mostly focused on disambiguating long and short queries, without discussing the effects of different application sources and context windows. With our research, we complete the picture by leveraging various data sources captured via a novel logging methodology by recording all information occurring on the screen, including user digital traces across a variety of applications. Screen recording can be used to model some parts of user activity that can be hard to sense in conventional logging systems, such as the content that was appearing and applications that were used before, during, and after a search. We use this rich

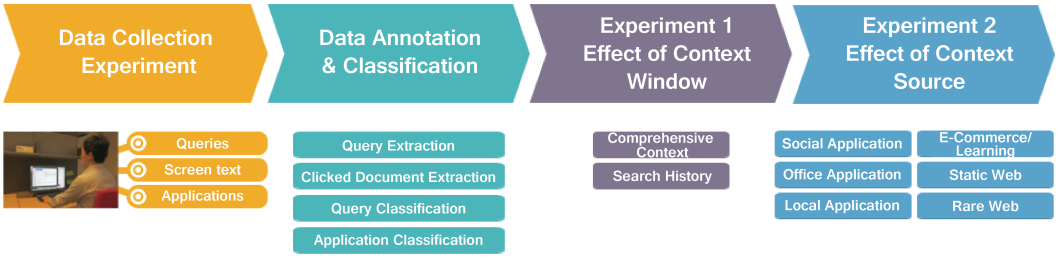


Fig. 2. The overall procedure consisted of four phases: 1) Data collection study for a duration of 14 days; 2) Data annotation and classification; 3) Experiment 1, investigating the effect of context window; and 4) Experiment 2, investigating the effect of context source on the retrieval performance.

source of context to shed light on the roles of various context sources in supporting effective query augmentation.

3 DATA COLLECTION STUDY

The purpose of the data collection study was to obtain in-the-wild search behavior and the associated contextual information. Details of the data collection study and the overall procedure of the two experiments are presented in Figure 2. We used a methodology in which the participants' laptops were continuously monitored and all digital activities logged for a duration of 14 days, including Web history and interaction history spanning a variety of applications. To capture the contents of the documents (e.g., Web pages, textual documents, and other types of files) that were examined and produced by the participants, we utilized a screen recording system that captures screen frames of active windows at 2-second intervals and stores them on the laptop's hard drive. All screen frames were converted to text using an Optical Character Recognition (OCR) system. Participants' search queries and clicked documents on the search engine result pages were extracted.

3.1 Apparatus

A custom-built screen recording system was developed for the experiment. Unlike other logging methods that have been used in traditional information retrieval studies, screen recording is not restricted to pre-specified logging functions. Apart from voice search, it can capture all possible user activities, including every input, as well as the presentation of content on the screen that occurs between the user and the computer before, during, and after searching [63].

The screen recording system captures the screen frames of active windows at 2-second intervals. The system was developed in two versions: We used Core Graphics API to implement the Mac operating system (OS) version, and we used Desktop App UI to implement the Microsoft Windows OS version. Both are native OS libraries and perform identical functions, recording and saving screen frames as images.

To produce a high-quality textual representation of the content from screen frame images, we checked the OCR accuracy using various settings for image processing. The settings that produced highly accurate recognized texts were used in our study. The settings included the pre-processing of each image using a script in the ImageMagick library[2] and converting the images to grayscale while

[2]https://imagemagick.org/script/convert.php

making the background white and the text black, then re-scaling them to 500%. The pre-processed images were converted to text using Tesseract (version 4.0[3]), which is a highly accurate OCR engine.

In addition, the system collected digital activity information, such as OS event logs that were associated with the screen frames, including the names of active applications and the Uniform Resource Locators (URLs) of active Web pages from Web browsing applications, and the timestamps when applications and documents/files were opened.

3.2 Participants

The participants were recruited by sending out invitations to mailing lists at the University of Helsinki. Only respondents who used a laptop as their main computing device and English as their main language for performing their everyday digital activities were considered eligible for the study. Another eligibility criterion for taking part in the study was having a higher education background, as we assumed that people satisfying this criterion would be more likely to use their laptops for everyday digital activities.

Thirteen participants in both university and industrial settings, and of varying professions, took part in the study. They were university students, computer scientists, and start-up entrepreneurs. Of these, six participants had bachelor's degrees, and seven participants had master's degrees. There were five males and eight females with an average age of 25 years (SD = 5). The participants had different cultural backgrounds: Six were from Nordic countries, two were from central Europe, one was from Africa, and four were from Asia.

The participants were informed of our privacy guidelines prior to joining the experiment and were told that the digital activity logs and screen recordings would be stored on their own laptops during the monitoring phase. After that, the data would be transferred to a secure server and used only for research purposes. After the experiment, the participants were compensated with 150 Euros. Consent was obtained from the participants regarding the data usage, privacy, and procedure of the experiment.

3.3 Ethics

We are very aware of the privacy implications of using screen recording data for research and have taken active steps to protect the participants. In particular, to safeguard participant privacy during the experiment, all screen frames were encrypted, stored locally on participant laptops during the recording, and never exposed to anyone except the participants themselves. Upon completing the experiment, we only archived the queries and ranking positions of the search results, removing all other personally identifiable information in the demographic information. We followed the ethical guidelines and principles of data anonymization and minimization at every stage of the data processing. The logs were archived and stored on a server protected by authentication mechanisms and firewalls. This work received ethical approval from the University of Helsinki Ethical Review Board in the Humanities and Social and Behavioural Sciences [4].

3.4 Screen Recording and Digital Activity Monitoring

Upon participants' agreeing to take part in the experiment, the screen recording and digital activity monitoring system were installed on each participant's laptop and set to run continuously in stealth mode for a duration of 14 days. The system was automatically launched whenever the laptop was turned on. To provide users with full control over the monitoring, a stopping function was also implemented. The user could stop the system from monitoring at any time simply by clicking on

[3]https://github.com/tesseract-ocr/tesseract/wiki/4.0-with-LSTM

[4]https://www.helsinki.fi/en/research/ethical-review-board-in-the-humanities-and-social-and-behavioural-sciences

the stop function. During the monitoring phase, the participants were advised to use their laptops as usual and to avoid stopping the system unless it was necessary. After the 14-day period, the participants visited our lab and the system was uninstalled from their laptops.

3.5 Results

The data collected from the 13 participants contained all Web activity logs, the actual queries submitted to the search engine, the Web pages visited, and interaction history on all applications. In total, the data collection consisted of an average 140,035 ($SD = 134,024$) screen frames per participant. This represents a history of an average of 78 hours ($SD = 73$) of computer usage per participant. We focused on unique screen frames by discarding duplicate frames or constant screen capturing of non-informative changes on the screen. This resulted in an average of 17,185 ($SD = 12,448$) screen frames per participant and an average of $1,204$ ($SD = 555$) unique documents were used over the course of two weeks. Due to a variety of online services offered on the Web, many user works were carried out not only on a local standard-alone application, but also on the Web browser. For example, Office 365 and Google Docs have the same functionalities as MS Office. Therefore, we decided to extract the domain names of the Web sites visited by the participants and considered them as separate applications. This resulted in an average of 108 ($SD = 45$) applications used per participant.

4 DATA ANNOTATION AND CLASSIFICATION

Data annotation and classification were conducted for the collected data prior to data analysis experiments. Queries and clicked documents on SERPs were extracted and classified according to their intent and application sources.

4.1 Query and Clicked Document Link Extraction

The preliminary step of data annotation and classification was to extract the participants' Web search queries from digital activity logs and screen recordings. We ran a script programmed to automatically identify all Web searches and queries from commercial search engines including Google Search, Bing Search, DuckDuckGo, Yandex, and Yahoo Search. Search engine usage was identified in the Web URLs of the collected screen frames. The queries were then extracted directly from the URLs. The corresponding clicked document links from the SERPs following the queries were also extracted.

A total of 1,518 unique Web search queries were found in the participants' screen recordings; 787 queries were identified with one or multiple clicked document links (e.g, either a user performed multiple clicks on the results at a time and viewed them individually later, or clicked a result and then came back to click another one). Of these, 645 queries had one clicked document, 123 queries had two clicked documents, 10 queries had three clicked documents, and 9 queries had four clicked documents. The remaining queries were potentially "good abandonment" or queries that may have resulted in user satisfaction without their needing to click on the SERP (e.g. SERP contains enough details to satisfy the user's information need without visiting the actual document). In addition, there were many cases in which queries were part of search sessions and had zero clicks on the results. This was because the user was not satisfied with the current search results and thus submitted more follow-up queries. However, when clicked documents were not observed, they were not part of the evaluated queries. Therefore, we only analyzed and reported the results of those queries that had clicked documents on SERP.

Intent Type	*Description*	*Examples*
Navigational	Search intent that is to navigate to a particular Website. The searcher may already have the specific words regarding the Website in mind [9].	google cloud; google drive; google sheets; docusign; slack; theverge; coursera; encyclopedia; d3js; beta freecodecamp.
Transactional	Search intent that is to locate a Website with the intent to obtain some product either electronically or a commercial real-life product [9] including purchase of a product, execution of an online application, or to download the "known" documents, images.	glossier cloud paint dusk; marlboro original jacket; flights from helsinki to rome; pdf model-driven formative evaluation of exploratory search; porvoo museum price.
Informational	Search intent that is to locate content concerning a particular topic in order to address an information need of the searcher [9]. In contrast to navigational, the query can be along a spectrum from very specific to something very vague.	remove windows media player; how many days to spend in rome; different strength of diamonds; hascode return too similar values; how to put text beside picture.

Table 2. Examples and descriptions of search intent types. The examples are taken from the participants' real-world queries extracted in the data collection study.

4.2 Query Classification

The aim of the query classification phase was to classify the extracted queries that had similar search intent into a set of target categories. The descriptions and examples of the search intents are presented in Table 2. Classification was chosen based on three basic intent types (informational, transactional, and navigational) that were extensively studied and have been widely used in many query augmentation studies [9, 14].

The intent types were assigned to the queries based on case-specific considerations. Two annotators classified the queries independently to study the inter-annotator reliability. First, the primary annotator went through each query, search results, and the clicked Web documents; and developed an initial classification scheme. Then the second annotator was called in to the task. The second annotator followed the same classification scheme and independently classified 20 queries. After that, the second annotator met with the primary annotator to discuss the intent types and revised the classification scheme. Then a random sample of 50 queries was assigned to both annotators for classification. These 50 queries were then used to compute the inter-annotator agreement using Cohen's Kappa. The test showed a high agreement between the annotators ($Kappa = 0.87$). Then the primary annotator classified the remaining queries.

4.3 Application Classification

The aim of the application classification phase was to classify applications into a set of categories based on their common functions, types, and fields of use. The application categories are presented in Table 3. The application names were extracted from the collected OS log information.

The *Social* category included applications of which the main function was to enable communication with other people (e.g., Skype, Mail, Facebook). The *E-commerce and E-learning* category typically featured websites used to support online interaction for learning and even to enable transactions (e.g., online stores, journey planner, MOOCs). Meanwhile, the *Static* category included websites that did not support transactions or online learning (e.g., personal web-blogs, on-line news services). Participants used many dedicated tools for office work, such as spreadsheets, word

Application Category	*Description*
Social	Applications and websites where main function is to enable communication between people.
Office	Applications or tools that are generally used to support office work.
Local	Local applications that are installed on the participant's computer but exclude applications categorized in the previous categories.
E-commerce/E-learning	Websites that are typically used for manifold interactions and that support interaction for learning and even enable transactions.
Static	Websites that are typically used for browsing and that do not support or encourage much other interaction for learning or transactions.
Rare	Websites used only once or twice are placed in this category.

Table 3. Application categories are data-driven. They are categorized based on common function and type of use.

processing, PowerPoint presentations, and pdf reader which were categorized into the *Office* category. Locally installed applications such as, system preferences, file explorers, and programming frameworks were grouped as the *Local* category. Finally, any website that occurred only once or twice in the recorded data was placed into the *Rare* category.

The classification followed a top-down approach [58] that was agreed upon by three researchers in our group. The researchers formed classifications for all applications in the data. Two researchers developed an initial classification scheme as follows. First, the researchers retrieved a description to each application and determined functions and features offered by the application; then applications that shared a similar set of functions and features were clustered into the same categories. The two researchers independently classified twenty applications and met to discuss the clusters and revised the classification scheme. After that, one researcher used the defined classification scheme to form clusters to the remaining applications. The two researchers assigned the labels and descriptions to the clusters and called them categories. A third researcher was invited to verify the classification scheme and check on naming conventions of the clusters. After the review and discussion on revisions, the final categories of applications were formed.

The double-blind inter-rater agreement was determined by calling an independent annotator who did not have any prior knowledge about the study or the application categories, to perform the classification. The annotator was first given the same set of descriptions and labels of the categories. The annotator followed the classification scheme determined earlier. Then, we extracted a random sample of five applications from each category and demonstrated the classification process to the annotator. A separate random sample of ten applications per category were extracted and assigned to the annotator for the actual classification. Based on Cohen's Kappa, the level of agreement was found to be high ($Kappa = 0.75$).

Overall, the application classification resulted in an average of 3,538 screen frames ($SD = 3,539$) for Social application; 5,167 screen frames ($SD = 8,433$) for Office application; 2,837 screen frames ($SD = 2,936$) for Local application; 1,755 screen frames ($SD = 2,622$) for E-commcerce/E-learning; 1,475 screen frames ($SD = 1,365$) for Static Web application; and 1,258 ($SD = 836$) screen frames for Rare Web application per participant.

4.4 Data Pre-processing

We focused on the information change on the screen, while removing terms that constantly appeared that did not provide any useful context, such as the application's title bar, menu bar, and other toolboxes. For this process, we utilized a frame difference technique in which the two adjacent screen frames were compared and the differences in pixel values were determined. That is, terms that appeared in the same pixels in the two adjacent screen frames were excluded from the OCR-processed document. After that, OCR-processed documents were pre-processed, lowercasing terms using gensim library[5], and stopword removal and lemmatization using nltk library[6].

5 CONTEXTUAL QUERY AUGMENTATION

We leveraged the recent digital activity of the user to model context and augment the current search query. The sources determined the information used to build the context models. As part of the analysis, we varied the sources used to construct the four models, which are described below.

- **Comprehensive context model**: The context model was constructed based on comprehensive contextual information and actions that occurred prior to the current query in the search. Actions comprise all OCR-processed documents (Web pages visited, textual documents opened, local files accessed, email, and instance messages recently read).
- **Search history model**: The search history model was constructed based on the user's search activity followed by a subsequent search or the current query. We applied a constraint to the data, accepting only OCR-processed documents of prior searches to train the model.
- **Non-search history model**: The non-search history model was constructed based on actions that occurred prior to the current query in the search, but excluding all search activities.
- **Application-specific model**: A model for each application type was created using the data assigned to the application category described earlier. We assumed that, if a user opened a specific application, then the application window contained useful content for modeling. All OCR-processed documents captured on that application were used to train the model.

5.1 Dirichlet-Hawkes processes

We used Dirichlet–Hawkes processes (DHP) [19] for topic modeling of search context. DHP is a time-dependent topic model that combines Dirichlet [4] and Hawkes processes [29] to uncover meaningful topics and their temporal dynamics in the temporal stream of user activities. Topic

[5]https://radimrehurek.com/gensim/
[6]https://www.nltk.org/api/nltk.corpus.html

Table 4. Main notation used in the article.

Notation	Description
T	logged timestamp associated with a user activity
q_T	a query submitted by the user at time T
q_{Tj}	jth term in query q_T
len	a context size with a threshold of 10 minutes, 1 hour, 1 day
D	set of OCR-processed documents truncated by len before time T using $d_{T-len:T}$
d_t	the tth OCR-processed document in D; ($t = 1, 2, ..., \|D\|$)
W	vocabulary of D
w_i	the ith word in W; ($i = 1, 2, ..., \|W\|$)
f_i^t	word count of w_i if w_i exists in d_t, or 0 otherwise
K	number of topics produced by DHP
z_k	the kth topic in K topics; ($k = 1, 2, ..., \|K\|$)

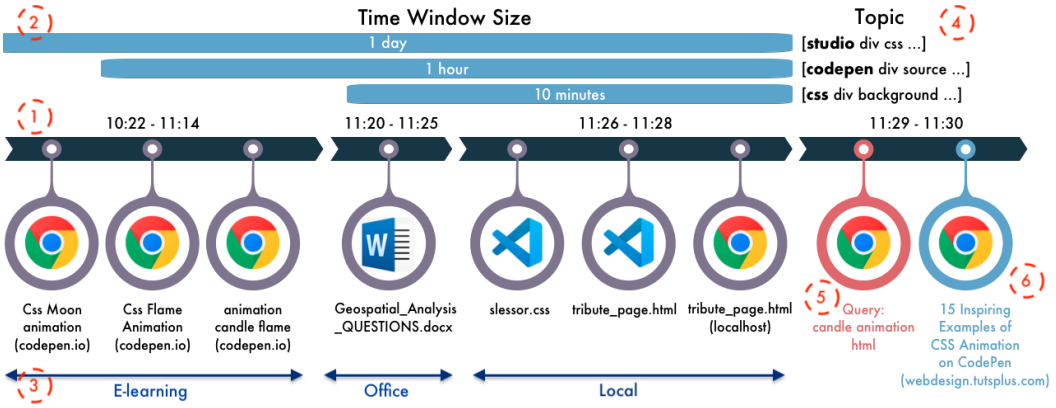


Fig. 3. An illustration of contextual query augmentation with varying context windows. 1) The user's interaction history spans the timeline from left to right. Applications used by the user are illustrated as icons, documents are illustrated as titles labelled beneath the icons. 2) Time windows are 10 minutes, 1 hour, and 1 day. 3) Applications and documents are classified under an application category. 4) Context models are constructed based on the predefined time window and the application category. 5) The context models were used to re-rank the conventional query augmentation model. For example, the original query "candle animation html" was augmented as "candle animation html *codepen*" when the model used the 1-hour context window. 6) The clicked SERP "Examples of CSS Animation on CodePen" was re-ranked.

modeling has been used in prior query augmentation research [11, 15] to infer the searchers' latent intent based on the terms to which they pay attention. A particular term or phrase that the searcher is reading in real-time can be used to derive what the user is currently interested in and is useful for query expansion.

Our pilot tests showed that DHP is particularly suitable to model the evolving nature of user periodic preference, compared with conventional topic models [15] that do not consider the temporal dynamics.

DHP was utilized to discover topic clusters from a stream of documents based on both the contents and temporal dynamics of their occurrence. The model was estimated through an online inference algorithm that jointly learns the cluster pattern and the parameters of the Hawkes processes for each cluster [19]. The use of DHP relies on the assumption that documents with similar topics emerging closely in time are related to each other. In the DHP model, each document (a file, a folder window, a text document, and a browsed Web page) was considered as an input unit. After model estimation, the resulting topic clusters are used to find useful query expansion terms.

5.2 Modeling Technique

Figure 3 illustrates an example of context modeling with varying context windows. In addition, the main notation used is described in Table 4. Given a Web query q_T inputted by the user at time T, and a collection of OCR-processed documents $D_{T-len:T}$ of a time window $len = \{10minutes, 1hour, 1day\}$ before time T were truncated and extracted for context modeling and query augmentation, our approach was based on the three following steps:

(1) Use $D_{T-len:T}$ to build a context model of the interaction history preceding the search.
(2) Use SERPs in response to query q_T to build a query augmentation model.
(3) Re-rank the query augmentation model using the context model.

Step 1: Context model. Given D, the context at each time step is defined as a vector over W terms. Each document is represented as a bag of words in which non-zero elements are the terms present in the current document. The context is stored in the matrix $X \in \mathcal{R}^{|W|\times|D|}$:

$$X = \begin{array}{c} \\ \mathrm{w}_1 \\ \mathrm{w}_2 \\ \\ \mathrm{w}_i \end{array} \begin{array}{c} \begin{array}{cccc} \mathrm{d}_1 & \mathrm{d}_2 & \dots & \mathrm{d}_t \end{array} \\ \begin{bmatrix} f_1^1 & f_1^2 & \cdots & f_1^t \\ f_2^1 & f_2^2 & \cdots & f_2^t \\ \vdots & \vdots & \ddots & \vdots \\ f_i^1 & f_i^2 & \cdots & f_i^t \end{bmatrix} \end{array},$$

where the element f_i^t is the count of w_i if w_i exists in the document d_t, or 0 otherwise; each d_t is associated with a timestamp; $|W|$ and $|D|$ are the set sizes; and $i = \{1, 2, ..., |W|\}$ and $t = \{1, 2, ..., |D|\}$.

The DHP approach projects X into a low-dimensional latent space such that co-occurring terms in documents should have similar representation. The model automatically yielded a fixed K number of topics, for which we denoted each topic as z_k. The resulting topic model assigned a membership probability distribution over the terms in each z_k, denoted as $p(w_i|z_k)$.

Given a query q_T, we obtained a score for each topic $p(z_k)$ according to the Dirichlet-Multinomial [19] log likelihood of q_T belonging to z_k as follows:

$$p(z_k) = \left(\frac{\Gamma(f^{z_k \backslash q_T} + |W|) \prod_{i=1}^{|W|} \Gamma(f_i^{z_k \backslash q_T} + f_i^{q_T} + \theta_0)}{\Gamma(f^{z_k \backslash q_T} + f^{q_T} + |W|) \prod_{i=1}^{|W|} \Gamma(f_i^{z_k \backslash q_T} + \theta_0)} \right),$$

where $f^{z_k \backslash q_T}$ is the term count of the topic z_k excluding the query q_T; $f_i^{z_k \backslash q_T}$ refers to the count of the ith term; f^{q_T} is the term count in q_T; and θ_0 is obtained from the DHP.

Intuitively, latent topics that are more semantically related to the query q_T would obtain higher scores. We then chose the top N terms from each of the K topics for each query q_T. Consequently, a total of $N * K$, forming W^{DHP}, which was selected for contextual query augmentation. For each term w_i, the score is calculated as below:

$$p(w_i|w_i \in W^{DHP}) = \sum_{k=1}^{K} p(w_i|z_k) \cdot p(z_k)$$

Step 2: Query augmentation model. Given q_T, we retrieved 1000 ranked SERPs using Microsoft Bing Web Search API v7[7]. A word embedding approach [39] was utilized to find terms that were semantically related to q_T for query augmentation. The Word2Vec Continuous Bag-of-Words model [57] was trained using the contents of SERPs. We used the content and comment extractors[8] of the Dragnet [50] to extract textual contents from SERPs. Words in the SERP collection are represented by a vector $\vec{w}$ embedded in a vector space. Similarities between terms are defined by cosine similarities between these vectors. Therefore, the score of term w_i given q_T in the SERP collection is:

$$p_{Vec}(w_i, q_T) = exp(cos(\vec{w}, \vec{q}_T)),$$

where $\vec{q}_T$ a term vector of q_T.

This approach scores terms by their semantic similarity to q_T as a whole. An alternative approach involves treating each term q_{Tj} in the query as a vector [1, 18], denoted as $\vec{q}_{Tj}$, and rank terms in

[7]https://azure.microsoft.com/en-us/services/cognitive-services/bing-web-search-api/
[8]https://github.com/dragnet-org/dragnet

the SERP collection according to $cos(\vec{w}, \vec{q}_{Tj})$. These similarity scores were summed and softmax-normalized to yield another score for term w_i:

$$p_{Sum}(w_i, q_T) = \sum_{q_{Tj} \in q_T} cos(\vec{w}, \vec{q}_{Tj})$$

We sum-normalize these terms' scores produced by the two approaches $\{Vec, Sum\}$ to obtain the a combined score distribution over terms, denoted as W^{SERP}. For each $w_i \in W^{SERP}$, the score is calculated as below:

$$p(w_i|w_i \in W^{SERP}) = p_{Vec}(w_i, q_T) + p_{Sum}(w_i, q_T)$$

Step 3: Re-rank the query augmentation model using the context model. The model producing W^{SERP} contains useful information for inferring users' search intention, but does not have contextual information. Therefore, we integrated W^{DHP} with W^{SERP} by union of the two sets of terms. If a term appearred in both W^{DHP} and W^{SERP}, we multiplied its score from the two sets, or assigned a 0 score otherwise. The formula for re-ranking query augmentation terms is as follows:

$$p(w_i) = p(w_i|w_i \in W^{DHP}) \cdot p(w_i|w_i \in W^{SERP})$$

Terms were ranked by sorting $p(w_i)$ in descending order, that is, terms that are more semantically related to the user context would be ranked higher. We chose the top N terms to expand the original query.

6 CONDITIONS

To study whether more context helps in query augmentation, we tested the model in varying conditions: the control condition, the query augmentation condition, the search context condition, the non-search context condition, the comprehensive context condition, and the application-specific context condition. Table 5 shows model configurations in these conditions, which are described in more detail below.

- **Control condition.** The initial ranking from the Bing search engine was used as a control condition. Rankings were obtained by sending a search request using the original query to Bing API to retrieve 1000 ranked Web documents.
- **Query augmentation (QAug) condition.** A query augmentation model with no contextual information was utilized in this condition.
- **Search context condition.** Search history was leveraged for contextual query augmentation. In this condition, a search history model was utilized.
- **Non-search context condition.** Interaction history excluding the searches was leveraged for contextual query augmentation. In this condition, a non-search history model was utilized.
- **Comprehensive context condition.** All interaction history was leveraged for contextual query augmentation. In this condition, a comprehensive context model was utilized.
- **Application-specific context condition.** Application-specific interaction history was leveraged for contextual query augmentation. In this condition, an application-specific model was utilized.

7 EXPERIMENT 1: EFFECT OF CONTEXT WINDOW

The purpose of the first experiment was twofold: first, to study the effect of a context window on the model performance in general; and second, to understand the effect of a context window on the model performance with respect to various search intents.

		Experiment 1					Experiment 2					
		Control	*QAug*	*Search*	*Non-search*	*Comprehensive*	*Social*	*Office*	*Local*	*E-commerce*	*Static Web*	*Rare Web*
Source	Search history	-	-	✓	-	✓	-	-	-	-	-	-
	Social	-	-	-	✓	✓	✓	-	-	-	-	-
	Office	-	-	-	✓	✓	-	✓	-	-	-	-
	Local	-	-	-	✓	✓	-	-	✓	-	-	-
	E-commerce/learning	-	-	-	✓	✓	-	-	-	✓	-	-
	Static Web	-	-	-	✓	✓	-	-	-	-	✓	-
	Rare Web	-	-	-	✓	✓	-	-	-	-	-	✓
Model	Context model	-	-	✓	✓	✓	✓	✓	✓	✓	✓	✓
	Context sizes	-	-	✓	✓	✓	✓	✓	✓	✓	✓	✓
	Query augmentation model	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	Bing API	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Table 5. Configurations of each compared condition. Contextual query augmentation model with context sources: search history and application-specific interaction histories are additive.

7.1 Measures

We used Mean Average Precision (*MAP*) at K as an evaluation metric because it has been widely used in query augmentation research [11, 15]. For each query q_T, we first calculated $Prec@K$ which is the fraction of clicked documents within the top K results. Then, we averaged all $Prec@K$ scores where each K is in the range of 1 to K. Finally, we computed a mean of all the queries' averages of $Prec@K$ to obtain *MAP*. More specifically, the *MAP* for each user is computed as follows:

$$MAP = \frac{1}{|Q|}\sum_{q_t \in Q}\frac{1}{K}\sum_{k=1}^{K} Prec@K(q_T),$$

where Q is a set of queries, K is chosen as $K = 5$.

In addition to *MAP*, we also used mean first rank (*MFR*) to evaluate the models. *MFR* is measured as the average of the rank positions of first relevant documents of all queries. In this research, we used the last-clicked documents when computing *MFR* as we believe the last clicked document is more likely to be relevant than the first. *MFR* is a commonly used metric as the quality criterion for measuring the retrieval performance in Web search [25]. We computed *MFR* by ranks of the top-100 retrieved documents, which means *MFR* ranges between 1 to 100 and a lower *MFR* was better. If the last-clicked document was not within top 100 documents, we considered its rank as 101. Given the set of queries Q, the MFR for each user is defined as follows:

$$MFR = \frac{1}{|Q|}\sum_{i=1}^{|Q|} rank_i,$$

where $rank_i$ is the rank position of the last clicked document for the i-th query.

7.2 Significance Tests

We applied a paired-samples t-test to seek significant differences between the compared models. Bonferroni correction was also applied to adjust for multiple comparisons. We used *MAP* and *MFR* as dependent variables; we used the control condition, the query augmentation condition, the conditions with contextual query augmentation using varying context windows (10 minutes, 1 hour, 1 day) as independent variables. In addition, the Shapiro-Wilk tests for normality were also conducted. Statistical analyses were performed with RStudio software (version 1.1.463)[9].

7.3 Number of query expansion terms

We examined retrieval effectiveness of our models using the top one, two, and three query expansion terms. The performances of both query augmentation and contextual query augmentation models degraded when the number of expansion terms increased (ref. Table 12 in the Appendix). We observed that when expanding with more than one term, the original query became over-specified, thus introducing undesirable side effects on retrieval performance. For example, if the query *"apache lucene tutorial"*, if it is augmented as *"apache lucene tutorial* ***core****"* which is a more accurate representation of user information need and requirement, more of the retrieved documents will be related to tutorials of lucene core. However, if it is augmented with more than one expansion term, as *"apache lucene tutorial* ***core demo package****"*, few of the retrieved documents will be related to tutorials, but most of them pertain to the demo module of lucene. As this experiment showed that the performance can be improved greatly when using one expansion term, we used one expansion term for the rest of Experiment 1 and also for Experiment 2.

7.4 Results

We performed a comparison of the retrieval performance of the control condition with no query augmentation, the query augmentation condition with no contextualization, the query augmentation in search context condition, the query augmentation in non-search context condition, and the query augmentation in comprehensive context condition. Table 6 shows the results of this comparison for each of the conditions using each context window (10 minutes, 1 hour, and 1 day). Figure 4 presents the percentage of queries that performed well or worse on query augmentation in terms of *MFR*. Evaluation measures were computed over each participant and the results were averaged. The standard deviations of the means are also reported.

7.4.1 Overall Performance. Contextual signals are useful in query augmentation, as indicated in the results that *MAP* and *MFR* of query augmentation in the context conditions (search history model, non-search history model, and comprehensive model) were significantly higher than the control condition. However, no significant difference was found between the control condition and the query augmentation condition when no context was considered.

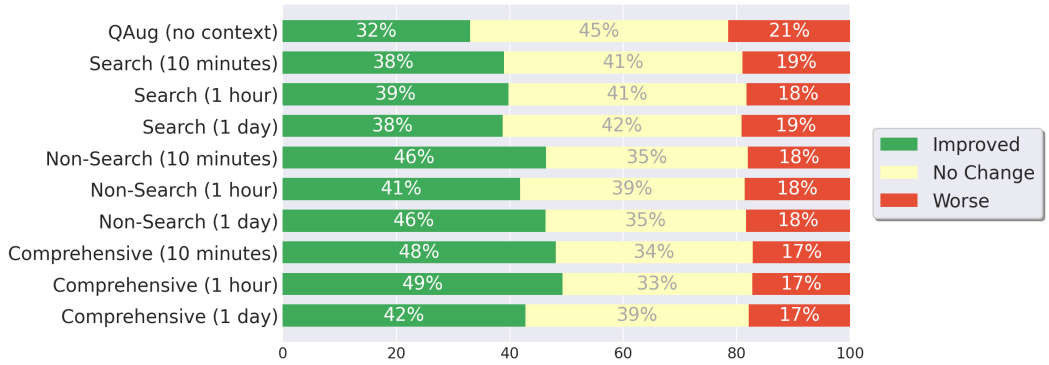
Furthermore, comprehensive use of context generally has a positive effect when using a short context window. This can be seen from the fact that the improvement in *MAP* for the query augmentation in comprehensive context condition is more substantial, compared with the query augmentation in search context condition. More precisely, when using a 10-minute context window, the query augmentation in search context condition has a lower performance, possibly because the search history at that time point is sparse ($MAP = 0.155$, $p = 0.02$, $W = 0.92$). Retrieval performance for the model in either the non-search context condition or the comprehensive context condition showed a significantly better performance. *MAP*s are 0.180 ($p = 0.00002$, $W = 0.97$) and 0.179 ($p = 0.00006$, $W = 0.94$) respectively. The differences between the non-search context condition

[9]https://www.rstudio.com/

Table 6. Retrieval performance of the context model, search history model, and the measure in terms of *MAP* and *MFR*. All experiments were performed with context windows of 10 minutes, 1 hour, and 1 day. Values in parentheses are standard deviations.

	MAP			*MFR*		
context window	10 minutes	1 hour	1 day	10 minutes	1 hour	1 day
Init (control) (i)	0.121 (0.058)			58.2 (12.5)		
QAug (a)	0.142 (0.050)			50.7 (13.3)		
vs. (i) p-value	*1*			*1*		
Search (s)	**0.155 (0.048)**	**0.168 (0.048)**	**0.161 (0.052)**	**45.5 (13.7)**	**44.4 (14.1)**	**45.1 (14.4)**
vs. (i) p-value	***0.02***	***0.0005***	***0.002***	***0.004***	***0.001***	***0.001***
vs. (a) p-value	*1*	*0.2*	*1*	*0.1*	*0.08*	*1*
Non-search (ns)	**0.180 (0.051)**	**0.165 (0.043)**	**0.163 (0.043)**	**41.2 (10.4)**	**43.8 (12.3)**	**42.4 (10.9)**
vs. (i) p-value	***0.00002***	***0.002***	***0.008***	***0.0004***	***0.001***	***0.0006***
vs. (a) p-value	***0.01***	*0.6*	*1*	***0.02***	*0.06*	*0.2*
vs. (s) p-value	***0.002***	*1*	*1*	*1*	*1*	*1*
Comprehensive	**0.179 (0.051)**	**0.178 (0.049)**	**0.173 (0.048)**	**39.8 (11.1)**	**40.0 (9.4)**	**41.2 (13.7)**
vs. (i) p-value	***0.00006***	***0.00008***	***0.0004***	***0.0002***	***0.0006***	***0.0004***
vs. (a) p-value	***0.02***	***0.04***	***0.01***	***0.004***	***0.01***	***0.001***
vs. (s) p-value	***0.008***	*1*	*1*	*0.2*	*1*	*1*
vs. (ns) p-value	*1*	*0.2*	*1*	*1*	*1*	*1*

Fig. 4. The percentage of queries have ranks of last-clicked documents changed from the control condition (ref. *MFR* in Table 6)



and the search context condition; and between the comprehensive context condition and the search context condition were also significant; p-values are 0.002 and 0.008 respectively.

Results for longer contexts (1 hour and 1 day) show that the performance of the search history model improved as more contextual information became available. *MAP*s are 0.168 ($W = 0.93$) and 0.161 ($W = 0.91$) for the query augmentation in search context condition when using 1-hour and 1-day respectively, and were significantly higher than in the control condition ($p = 0.0005$ and $p = 0.002$). In addition, the query augmentation in non-search context condition also has a higher performance, compared to the control condition. *MAP*s are 0.165 ($W = 0.92$) and 0.163 ($W = 0.94$) for the non-search context model with 1-hour and 1-day respectively. Differences between the non-search context condition and the control condition were significant ($p = 0.002, p = 0.008$). Likewise, the query augmentation in comprehensive context condition also performed better than

that in the control condition. MAPs are 0.178 ($W = 0.91$) and 0.173 ($W = 0.94$) for the comprehensive context model when using 1-hour and 1-day context windows, respectively. Differences between the comprehensive context condition and the control condition were also significant ($p = 0.00008$ and $p = 0.0004$). However, differences among the conditions with the models using different contextual signals (search, non-search, and comprehensive) were not significant. No significant differences were found among the contextual query augmentation using different context windows. Therefore we can conclude that using more extensive sources is quite beneficial but can only help to address the sparsity of short-term context in search history.

From another perspective, the results for *MFR* confirm the effectiveness of the comprehensive context model. With varying context windows, the query augmentation in comprehensive context condition performed better than both the query augmentation condition with no contextualization and the control condition. In general, the results show that the model in comprehensive context condition improved search ranking by ranking the last-clicked document at higher position (*MFR* = 39.8, 40.0, 41.2 for 10-minute, 1-hour, 1-day context window respectively), whereas, *MFR* for the model in search context condition has a lower performance (*MFR* = 45.5, 44.4, 45.1 for 10-minute, 1-hour, 1-day context window respectively). We did not see improvements with the search history model over the query augmentation without contextualization model.

These results can be further explained by inspecting the percentage of queries for which the *MFR* performance is improved or worse than the control condition in Figure 4. For the model in comprehensive context condition, 48%, 49%, and 42% of the queries showed an improved performance when using 10-minute, 1-hour, and 1-day context windows respectively, whereas for the model in the search context condition, the percentage of queries positively affected by query augmentation remains almost the same, despite the growing amount of contextual information used. Around 38%-39% of queries have the rankings of the last-clicked documents improved in search context condition.

Such results suggest that the comprehensive context can provide more useful information for improving retrieval performance than using the search history alone. We believe this improvement

Table 7. Retrieval performance by search intent types in terms of *MAP* and *MFR*.

	MAP			*MFR*		
context window	10 minutes	1 hour	1 day	10 minutes	1 hour	1 day
Init (control) (i)	0.079 (0.042)			64.2 (14.1)		
QAug (a)	0.122 (0.040)			52.9 (13.5)		
vs. (i) p-value	*0.2*			*0.3*		
Search (s)	0.128 (0.048)	**0.141 (0.038)**	**0.135 (0.049)**	**48.8 (13.4)**	**47.7 (14.7)**	**49.1 (13.6)**
vs. (i) p-value	*0.07*	***0.001***	***0.01***	***0.01***	***0.01***	***0.01***
vs. (a) p-value	*1*	*0.5*	*1*	*0.5*	*1*	*1*
Non-search (ns)	**0.148 (0.046)**	**0.132 (0.031)**	**0.134 (0.032)**	**42.6 (10.8)**	**46.1 (13.3)**	**46.4 (9.5)**
vs. (i) p-value	***0.0004***	***0.0002***	***0.0003***	***0.003***	***0.005***	***0.004***
vs. (a) p-value	*0.9*	*1*	*1*	*0.2*	*0.3*	*0.5*
vs. (s) p-value	*1*	*1*	*1*	*0.9*	*1*	*1*
Comprehensive	**0.149 (0.052)**	**0.144 (0.054)**	**0.147 (0.047)**	**41.9 (11.1)**	**42.1 (11.1)**	**40.5 (11.4)**
vs. (i) p-value	***0.001***	***0.004***	***0.0001***	***0.001***	***0.003***	***0.00006***
vs. (a) p-value	*0.6*	*1*	*1*	*0.09*	*0.07*	***0.0004***
vs. (s) p-value	*0.9*	*1*	*1*	*0.3*	*1*	***0.01***
vs. (ns) p-value	*1*	*1*	*1*	*1*	*1*	***0.03***

(a) Informational intent

	MAP			*MFR*		
context window	10 minutes	1 hour	1 day	10 minutes	1 hour	1 day
Init (control) (i)	0.120 (0.121)			61.1 (19.2)		
QAug (a)	0.166 (0.121)			49.4 (24.8)		
vs. (i) p-value	*1*			*1*		
Search (s)	0.186 (0.130)	**0.202 (0.125)**	0.192 (0.126)	**41.9 (22.4)**	**42.8 (23.0)**	**40.2 (23.9)**
vs. (i) p-value	*0.09*	***0.01***	*0.07*	***0.03***	***0.01***	***0.03***
vs. (a) p-value	*1*	*1*	*1*	*1*	*1*	*1*
Non-search (ns)	**0.218 (0.127)**	**0.206 (0.112)**	**0.201 (0.133)**	**37.0 (17.4)**	**42.8 (23.5)**	**37.3 (21.4)**
vs. (i) p-value	***0.0006***	***0.04***	***0.04***	***0.02***	***0.04***	***0.04***
vs. (a) p-value	*0.9*	*0.9*	*1*	*0.9*	*1*	*0.5*
vs. (s) p-value	*0.9*	*1*	*1*	*1*	*1*	*1*
Comprehensive	**0.220 (0.127)**	**0.223 (0.128)**	**0.196 (0.136)**	**36.6 (17.6)**	**35.6 (17.1)**	**40.9 (21.4)**
vs. (i) p-value	***0.0006***	***0.0009***	***0.01***	***0.01***	***0.01***	***0.01***
vs. (a) p-value	*0.9*	*0.9*	*1*	*0.8*	*0.6*	*1*
vs. (s) p-value	*0.8*	*1*	*1*	*1*	*1*	*1*
vs. (ns) p-value	*1*	*1*	*1*	*1*	*1*	*1*

(b) Transactional intent

	MAP			*MFR*		
context window	10 minutes	1 hour	1 day	10 minutes	1 hour	1 day
Init (control) (i)	0.238 (0.085)			44.2 (10.2)		
QAug (a)	0.177 (0.084)			44.9 (11.0)		
vs. (i) p-value	*1*			*1*		
Search (s)	0.181 (0.073)	0.190 (0.079)	0.189 (0.076)	39.7 (11.7)	37.8 (12.8)	37.4 (15.1)
vs. (i) p-value	*1*	*1*	*1*	*0.7*	*0.1*	*1*
vs. (a) p-value	*1*	*1*	*1*	*0.7*	*0.9*	*1*
Non-search (ns)	0.214 (0.081)	0.200 (0.077)	0.190 (0.092)	38.7 (11.3)	40.4 (12.4)	37.8 (13.0)
vs. (i) p-value	*1*	*1*	*1*	*0.9*	*1*	*0.2*
vs. (a) p-value	*1*	*1*	*1*	*1*	*1*	*1*
vs. (s) p-value	*1*	*1*	*1*	*1*	*1*	*1*
Comprehensive	0.201 (0.083)	0.200 (0.083)	0.199 (0.096)	35.2 (10.9)	37.3 (12.1)	38.1 (10.2)
vs. (i) p-value	*1*	*1*	*1*	*0.09*	*0.6*	*0.1*
vs. (a) p-value	*1*	*1*	*1*	*0.1*	*0.5*	*0.1*
vs. (s) p-value	*1*	*1*	*1*	*1*	*1*	*1*
vs. (ns) p-value	*1*	*1*	*1*	*1*	*1*	*1*

(c) Navigational intent

is due to the additional information from non-search sources that is incorporated into the model. In fact, the non-search context information is more beneficial when the model uses a short context window (10-minute). As can be seen in the results, 46% of queries have the *MFR* performance improved when a 10-minute context window is used. The difference in *MFR* between the non-search context condition and the query augmentation condition was also significant ($p = 0.02, W = 0.96$). However, the differences among the context conditions (search, non-search, comprehensive context condition) were not significant. Therefore, we believe that the model using search history alone is already effective in improving the ranking of the last-clicked document. Integration of search

context with non-search context may produce a better performance; in particular, immediate (short-term) non-search contexts are most useful in improving query augmentation.

7.4.2 Intent-level Analysis. In this section, we explore the correlation between the query intents and retrieval performance of query augmentation with the model using different context windows. The results are shown in Table 7 and Figure 5.

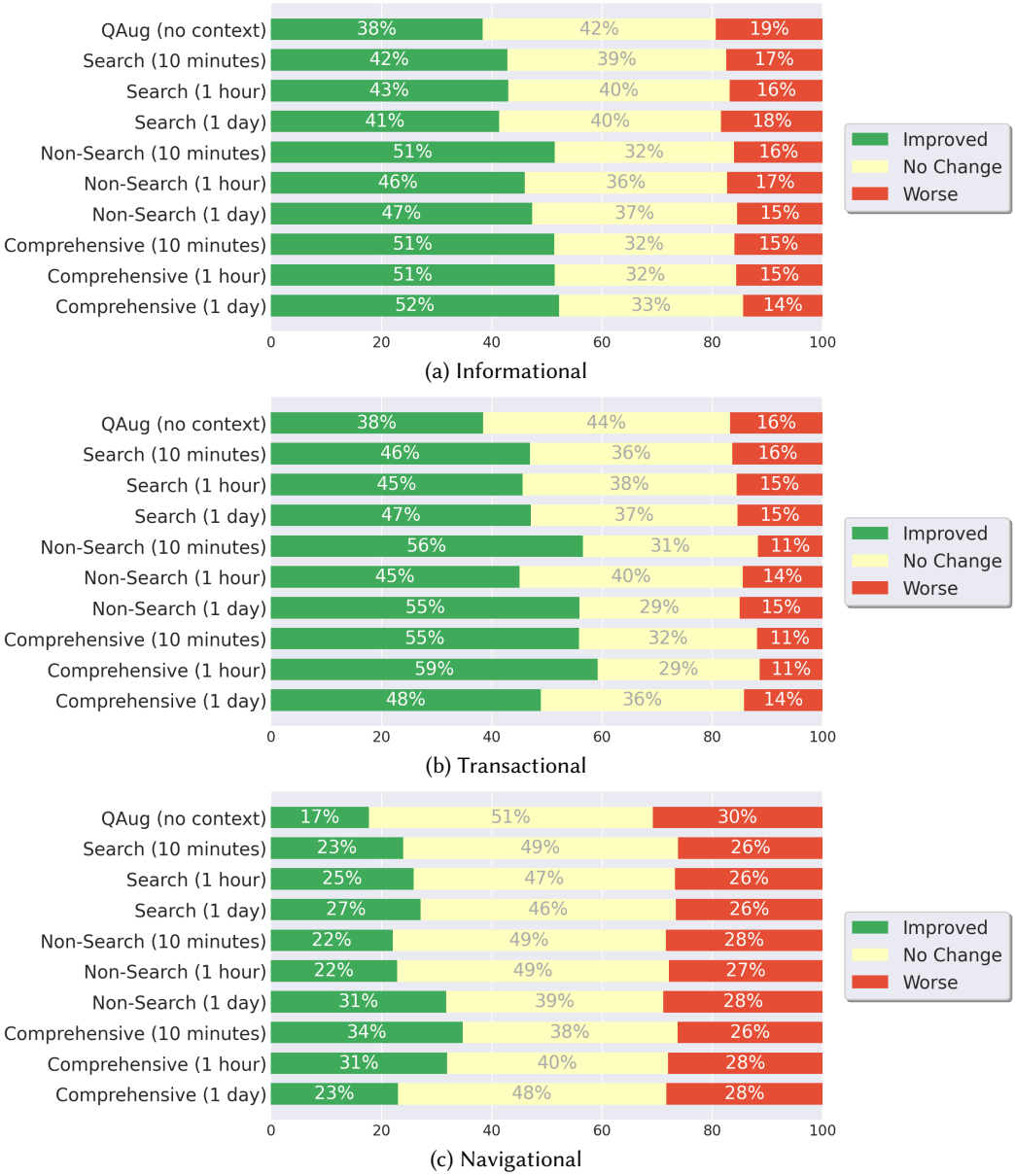


Fig. 5. The percentage of queries, by search intents, with ranks of last-clicked documents changed from the control condition (ref. *MFR*s in Table 7a, 7b, 7c).

Here, we see that, in general, query augmentation performed well for queries with informational and transactional intents (Tables 7a, 7b). However, we did not see any improvement with queries with navigational intent (Table 7c); this is due to the fact that navigational queries are well-specified, which mean that query augmentation may not have an effect on this type of query.

The results for query augmentation for informational intent shown in Table 7a indicates *MAPs* are 0.141 and 1.35 for the model in the search context condition using 1-hour and 1-day context windows ($p = 0.001, 0.01$; $W = 0.93, 0.92$ respectively). For queries with transactional intent, we found an improvement with the search history model using 1-hour context window ($MAP = 0.202$, $p = 0.01, W = 0.93$). The results show that the model in the search context condition improved the performance when increasing the amount of search history being taken into account, as the context became rich. On the other hand, we can see an overall positive effect of the comprehensive context model across varying context windows with significant improvement over the control condition. However, the difference in *MAP* between the comprehensive context condition and the search context condition was not significant.

Furthermore, in terms of *MFR*, all the models in context conditions (search, non-search, comprehensive) using 10-minute and 1-hour context windows performed equally well and improved over the control condition for queries with informational and transactional intents. Over 41% of queries with informational intent (Figure 5a) and over 45% of queries having transactional intent (Figure 5b) performed well on augmentation in the search context condition. Paired t-tests indicate the differences between each context condition (search, non-search, comprehensive) and the control condition were significant ($p < 0.04$). Increasing the amount of comprehensive context information (1-day) led to a greater improvement in *MFR* for the queries with informational intent (Table 7a). *MFR* is 40.5 ($W = 0.96$) for the model using comprehensive context with a 1-day context window; this was also true for over 52% of queries with informational intent for which *MFR* was better than that of the control condition (Figure 5a). The difference between the comprehensive context condition and the search context condition was significant ($p = 0.01$). This suggests that for better ranking of the last-clicked documents, query augmentation would most benefits the queries with informational intent when long-term contexts from search and non-search histories are combined.

8 EXPERIMENT 2: EFFECT OF APPLICATION SOURCE

The purpose of the second experiment was to understand how the source of contextual information affects Web search performance. More simply, it tested whether the conditions with the models using different application-specific context generated based on the six sources of contextual information can be useful in improving the quality of search results.

8.1 Measures and Significance Tests

Similarly to the first experiment, the main evaluation criterion in this experiment was retrieval quality. We also used *MAP* and *MFR* to measure the effectiveness of contextual query augmentation.

A paired-samples t-test was applied to find significant differences between the compared models. We also used *MAP* and *MFR* as dependent variables. We used the condition with the model using application-specific context, the condition with the model using search history, the query augmentation when no context was considered, and the control condition without query augmentation as independent variables; we also used context windows and search intent types as independent variables.

Table 8. Retrieval performance of application-specific context models in terms of *MAP* and *MFR*. All experiments were performed with context windows of 10 minutes, 1 hour, and 1 day. Values in parentheses are standard deviations.

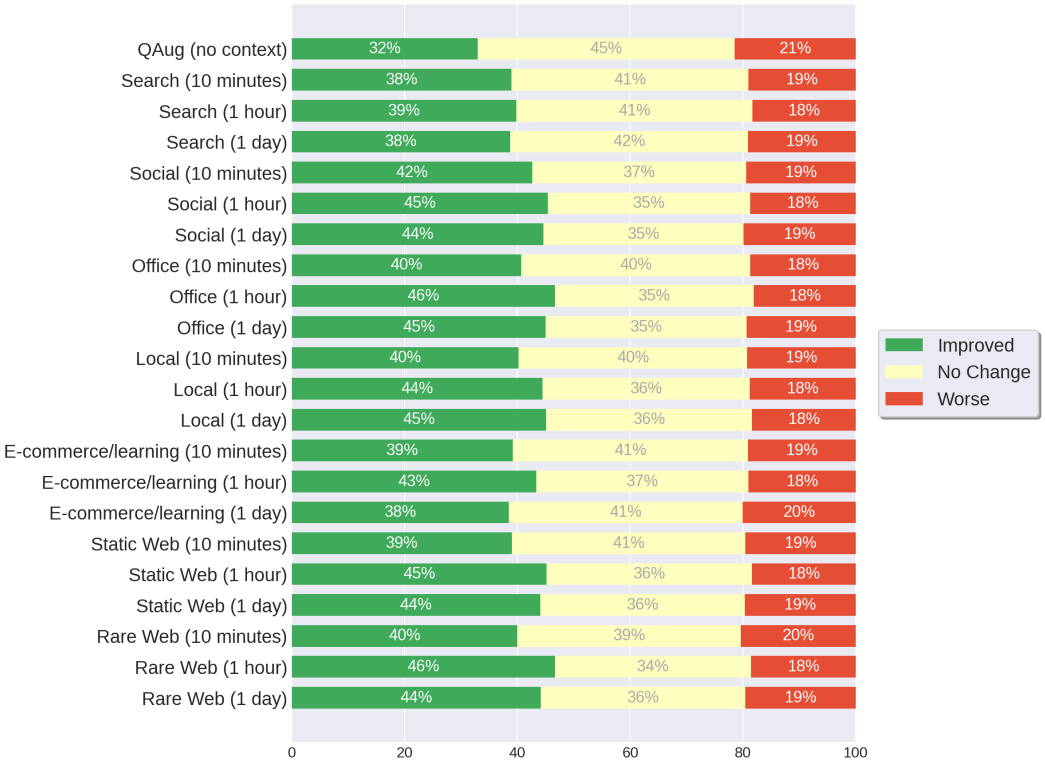
	MAP			*MFR*		
context window	10 minutes	1 hour	1 day	10 minutes	1 hour	1 day
Init (control) (i)	0.121 (0.058)			58.2 (12.5)		
QAug (a)	0.142 (0.050)			50.7 (13.3)		
Search (s)	0.155 (0.048)	0.168 (0.048)	0.161 (0.052)	45.5 (13.7)	44.4 (14.1)	45.1 (14.4)
Social	**0.157 (0.042)**	**0.171 (0.045)**	**0.166 (0.047)**	**44.2 (10.6)**	**42.8 (8.9)**	**43.2 (10.4)**
vs. (i) p-value	***0.02***	***0.001***	***0.01***	***0.002***	***0.0004***	***0.0002***
vs. (a) p-value	*1*	***0.05***	*0.6*	*0.1*	*0.1*	*0.2*
vs. (s) p-value	*1*	*1*	*1*	*1*	*1*	*1*
Office	**0.164 (0.051)**	**0.172 (0.052)**	**0.163 (0.054)**	**44.4 (13.1)**	**41.5 (10.7)**	**42.5 (11.3)**
vs. (i) p-value	***0.002***	***0.0004***	***0.01***	***0.01***	***0.0002***	***0.0004***
vs. (a) p-value	*1*	***0.02***	*0.2*	*0.1*	***0.02***	*0.1*
vs. (s) p-value	*1*	*1*	*1*	*1*	*1*	*1*
Local	**0.156 (0.051)**	**0.167 (0.054)**	**0.167 (0.52)**	**45.1 (13.9)**	**43.7 (10.1)**	**42.0 (10.2)**
vs. (i) p-value	***0.01***	***0.001***	***0.002***	***0.006***	***0.0008***	***0.0002***
vs. (a) p-value	*1*	*0.08*	*0.08*	*0.1*	*0.6*	*0.06*
vs. (s) p-value	*1*	*1*	*1*	*1*	*1*	*1*
E-commerce	**0.159 (0.051)**	**0.173 (0.054)**	**0.170 (0.048)**	**45.0 (13.7)**	**43.7 (10.1)**	**45.6 (14.3)**
vs. (i) p-value	***0.01***	***0.0006***	***0.01***	***0.002***	***0.0006***	***0.002***
vs. (a) p-value	*0.4*	***0.02***	***0.01***	*0.1*	*0.4*	*0.8*
vs. (s) p-value	*1*	*1*	*1*	*1*	*1*	*1*
Static Web	**0.161 (0.056)**	**0.167 (0.053)**	**0.165 (0.056)**	**45.5 (14.6)**	**42.8 (10.4)**	**43.6 (11.4)**
vs. (i) p-value	***0.02***	***0.006***	***0.01***	***0.01***	***0.0002***	***0.001***
vs. (a) p-value	*0.4*	*0.08*	*0.1*	*0.6*	*0.1*	*0.4*
vs. (s) p-value	*1*	*1*	*1*	*1*	*1*	*1*
Rare Web	**0.168 (0.047)**	**0.174 (0.051)**	**0.162 (0.054)**	**43.5 (13.2)**	**41.6 (9.1)**	**43.5 (10.8)**
vs. (i) p-value	***0.002***	***0.002***	***0.06***	***0.0006***	***0.0001***	***0.0004***
vs. (a) p-value	*0.08*	***0.002***	*1*	*0.1*	***0.04***	*0.2*
vs. (s) p-value	*1*	*1*	*1*	*1*	*1*	*1*

8.2 Results

In total, six different models with different types of contextual information were trained: *Social Application* model, *Office Application* model, *Local Application* model, *E-commerce/e-learning* model, *Static Web* model, and *Rare Web* model. An overview of the main results is shown in Table 8.

8.2.1 Overall Performance. The results, in general, indicate that the query augmentation in search context condition had a lower performance than the model in other application-specific conditions. The search history model was not able to outperform query augmentation with no contextualization, whereas other application-specific models consistently improved the performance when larger context windows were used. For instance, the results for the models in four application-specific context conditions (Social Application, Office, E-commerce, and Rare Web) show that increasing the amount of contextual information (by increasing the context window to 1 hour) led to greater improvements in *MAP*. The differences between those application-specific context conditions and the query augmentation condition were significant with $p < 0.05$.

Fig. 6. The percentage of queries in application-specific context conditions with ranks of last-clicked documents changed from the control condition (ref. *MFR* in Table 8)



The model in the E-commerce/learning context condition reached much higher *MAP*, in contrast to other application sources. Surprisingly, the model was most effective when it used larger context windows. From the results, we can see that the E-commerce/learning model steadily increased its performance as more contextual information became available and reached high *MAP*s in the 1-hour and 1-day context windows ($MAPs = 0.173, 0.170$; $W = 0.94, 0.92$; and $p = 0.0006, 0.01$ respectively). On the other hand, the search history model did not improve its performance over increasing context windows as the search sessions may not have contained some of relevant information that was only available in other specific-application context.

MFR results show similar improvements over the control condition for the models in application-specific context conditions (p values < 0.01). Furthermore, the Local context condition and Rare Web context condition have highest *MFR* values when the models used a 1-hour context window and outperformed the query augmentation when no context was considered. *MFR*s are 41.5, 41.6 for the Local context condition and Rare Web context condition ($p = 0.02, 0.04$ respectively). We also observed that 46% of queries in those context conditions showed a better performance than the control condition (ref. Figure 6). This suggests that the positive effect of using contextual information comes from the use of interaction data captured from non-search applications.

8.2.2 Intent-level Analysis. The results in Tables 9, 10, 11 and Figures 7, 8, 9 show that the improvements for different types of queries are respective to the amount of application-specific contexts used. In general, queries with informational and transactional intents achieved significant

Table 9. The results for informational intent pertaining to the retrieval performance of application-specific context models.

	MAP			*MFR*		
context window	10 minutes	1 hour	1 day	10 minutes	1 hour	1 day
Init (control) (i)	0.079 (0.042)			64.2 (14.1)		
QAug (a)	0.122 (0.040)			52.9 (13.5)		
Search (s)	0.128 (0.048)	0.141 (0.038)	0.135 (0.049)	48.8 (13.4)	47.7 (14.7)	49.1 (13.6)
Social	**0.134 (0.033)**	**0.145 (0.031)**	**0.146 (0.035)**	**46.6 (9.5)**	**44.7 (9.0)**	**45.2 (8.1)**
vs. (i) p-value	***0.003***	***0.0008***	***0.002***	***0.02***	***0.003***	***0.004***
vs. (a) p-value	*1*	*1*	*0.1*	*0.7*	*0.1*	*0.8*
vs. (s) p-value	*1*	*1*	*1*	*1*	*1*	*1*
Office	**0.138 (0.042)**	**0.143 (0.036)**	**0.132 (0.042)**	**47.0 (13.4)**	**45.5 (15.4)**	**46.9 (12.4)**
vs. (i) p-value	***0.001***	***0.0002***	***0.01***	***0.02***	***0.008***	***0.006***
vs. (a) p-value	*1*	*1*	*1*	*0.08*	*0.3*	*0.8*
vs. (s) p-value	*1*	*1*	*1*	*1*	*1*	*1*
Local	**0.130 (0.045)**	**0.135 (0.041)**	**0.140 (0.039)**	**48.5 (13.9)**	**49.2 (14.9)**	**47.6 (12.5)**
vs. (i) p-value	***0.03***	***0.002***	***0.002***	***0.02***	***0.05***	***0.01***
vs. (a) p-value	*1*	*1*	*0.7*	*0.08*	*1*	*1*
vs. (s) p-value	*1*	*1*	*1*	*1*	*1*	*1*
E-commerce	0.125 (0.042)	**0.138 (0.039)**	**0.138 (0.033)**	**49.2 (12.9)**	50.1 (13.8)	**48.9 (10.6)**
vs. (i) p-value	*0.07*	***0.001***	***0.005***	***0.04***	*0.06*	***0.01***
vs. (a) p-value	*1*	*1*	*1*	*1*	*1*	*1*
vs. (s) p-value	*1*	*1*	*1*	*1*	*1*	*1*
Static Web	0.128 (0.049)	**0.137 (0.045)**	**0.136 (0.042)**	49.4 (13.7)	**49.1 (15.4)**	**48.8 (12.6)**
vs. (i) p-value	*0.07*	***0.008***	***0.01***	*0.06*	***0.02***	***0.03***
vs. (a) p-value	*1*	*1*	*1*	*1*	*1*	*1*
vs. (s) p-value	*1*	*1*	*1*	*1*	*1*	*1*
Rare Web	**0.129 (0.042)**	**0.136 (0.046)**	0.127 (0.050)	**44.9 (10.2)**	**44.4 (9.6)**	**46.2 (9.6)**
vs. (i) p-value	***0.007***	***0.002***	*0.09*	***0.001***	***0.0001***	***0.002***
vs. (a) p-value	*1*	*1*	*1*	*0.3*	*0.3*	*1*
vs. (s) p-value	*1*	*1*	*1*	*1*	*1*	*1*

improvements over the control condition, whereas we did not see improvement for queries having navigational intent. Obviously, majority of queries with navigational intent already had optimal value for a click metric (most clicks at position 1), because the user already had a particular URL to find and was able to specify that in a query. However, even though the application-specific context used was fragmented, for instance, accounting for user behavior on a particular type of application, we still observed significant improvements in *MFR* for all application-specific models, which affected over 47% of all queries with informational intent (for example, query augmentation in the Social context condition using 1-hour and 1-day contexts) and up to 55% of all queries with transactional intent (for example, query augmentation in the Office context condition and in the Static Web context condition using 1-hour and 1-day contexts).

Another important observation in the results for informational intent (Table 9) is that the improvements in *MAP* for the query augmentation with the model using short-term information (10-minute) in four context conditions (Social, Office, Local, and Rare Web) performed better than the control condition ($p < 0.03$) whereas the model in the search context condition did not improve over the control condition. *MAP*s are 0.134, 0.138, 0.130, 0.129 (W values > 0.91) for the models using 10-minute window in the Social, Office, Local, and Rare Web context conditions, respectively.

Fig. 7. The results for informational intent pertaining to the percentage of queries in application-specific context conditions with ranks of last-clicked documents changed from the control condition (ref. *MFR* in Table 9)

Finally, the results for transactional intent in Table 10 show consistent improvements in different application-specific context conditions with the model using a 1-day context window, apart from the Social context condition. The differences between each application-context condition and the control condition were statistically significant (p values < 0.01). However, no significant differences were found between each application-specific context condition and the search context condition. The result suggests that the models incorporating application-specific contexts and search context may have had access to equally effective information, and as expected, their performances were not statistically different.

9 FINDINGS

We demonstrated the advantage of our contextual query augmentation methods in retrieval performance. The results revealed interesting dependencies between the context windows and application sources. In the following, we distill generalizable findings from the results and reflect on the research questions defined earlier.

RQ1. Does comprehensive use of context improve retrieval performance? Yes, as indicated, the context from more extensive sources improved the effectiveness of query augmentation.

Finding 1: Current user modeling approaches in previous studies [26, 47, 49] based on simple behavioral traits, such as clicks or dwell-time on search systems have been shown to be effective, but

Table 10. The results for transactional intent pertaining to the retrieval performance of application-specific context models.

	MAP			*MFR*		
context window	10 minutes	1 hour	1 day	10 minutes	1 hour	1 day
Init (control) (i)	0.120 (0.121)			61.1 (19.2)		
QAug (a)	0.166 (0.121)			49.4 (24.8)		
Search (s)	0.186 (0.130)	0.202 (0.125)	0.192 (0.126)	41.9 (22.4)	42.8 (23.0)	40.2 (23.9)
Social	0.193 (0.127)	**0.205 (0.126)**	0.194 (0.146)	44.4 (20.7)	45.1 (19.5)	43.4 (21.7)
vs. (i) p-value	*0.06*	***0.03***	*0.09*	*0.1*	*0.1*	*0.1*
vs. (a) p-value	*1*	*1*	*1*	*1*	*1*	*1*
vs. (s) p-value	*1*	*1*	*1*	*1*	*1*	*1*
Office	**0.195 (0.136)**	**0.202 (0.142)**	**0.203 (0.138)**	**38.0 (21.4)**	**38.7 (19.6)**	**37.1 (21.4)**
vs. (i) p-value	***0.01***	***0.009***	***0.01***	***0.05***	***0.02***	***0.03***
vs. (a) p-value	*1*	*1*	*1*	*1*	*1*	*0.5*
vs. (s) p-value	*1*	*1*	*1*	*1*	*1*	*1*
Local	0.190 (0.128)	**0.202 (0.122)**	**0.207 (0.131)**	**41.7 (23.1)**	**41.8 (20.5)**	**37.3 (21.5)**
vs. (i) p-value	*0.06*	***0.02***	***0.01***	***0.02***	***0.007***	***0.03***
vs. (a) p-value	*1*	*1*	*1*	*1*	*1*	*0.5*
vs. (s) p-value	*1*	*1*	*1*	*1*	*1*	*1*
E-commerce	0.186 (0.126)	**0.205 (0.126)**	**0.215 (0.129)**	**41.0 (22.7)**	**38.8 (20.8)**	**37.9 (23.9)**
vs. (i) p-value	*0.06*	***0.02***	***0.004***	***0.02***	***0.01***	***0.03***
vs. (a) p-value	*1*	*1*	*1*	*0.4*	*0.1*	*0.5*
vs. (s) p-value	*1*	*1*	*1*	*1*	*1*	*1*
Static Web	0.189 (0.126)	**0.202 (0.124)**	**0.197 (0.143)**	**42.2 (23.2)**	**37.3 (18.9)**	**37.9 (21.6)**
vs. (i) p-value	*0.06*	***0.03***	***0.01***	***0.03***	***0.002***	***0.01***
vs. (a) p-value	*1*	*1*	*1*	*1*	*1*	*1*
vs. (s) p-value	*1*	*1*	*1*	*1*	*1*	*1*
Rare Web	**0.215 (0.119)**	**0.217 (0.114)**	**0.202 (0.119)**	**40.1 (24.5)**	**39.4 (22.4)**	**36.8 (19.9)**
vs. (i) p-value	***0.01***	***0.007***	***0.01***	***0.05***	***0.03***	***0.01***
vs. (a) p-value	*1*	*1*	*1*	*1*	*1*	*0.4*
vs. (s) p-value	*0.5*	*0.7*	*1*	*1*	*1*	*1*

are still missing many important sources for context modeling. We find from the results that search history is an effective source of contextual information, but comprehensive context can be used to complement or replace search history when extensive search history is not available. The finding suggests that there are many useful sources of context and that if they are available to the search system, it may be possible to address many cold-start problems [24]. In contrast, incorporating a longer search history is effective in improving retrieval performance as more context information becomes available, with an increase in click metrics: 13% improvement in *MAP* (from 0.128 to 0.141 with 1-hour context window) (ref. Table 6) and by fetching the last-clicked document to a higher position for up to 39% of the queries (1-hour context window) (ref. Figure 4). However, additional source information did not improve results when more extensive search history was available.

Finding 2: Queries with informational intent could be impacted by the use of more contextual information. Context models based on long-term behavior from more extensive sources presented a significant opportunity to improve search performance for queries with informational intent. For example, 51% of queries with informational intent received more improvements in terms of *MFR* with the model using the comprehensive context than the model using search history (ref. Table 7a), but only long-term behavior was useful. On the other hand, the use of more contextual

Fig. 8. The results for transactional intent pertaining to the percentage of queries in application-specific context conditions with ranks of last-clicked documents changed from the control condition (ref. *MFR* in Table 10)

information did not provide more benefits, compared to the model using search history when the queries were of transactional intent. Although the proportion of transactional queries affected by comprehensive contextualization was relatively large (55% - 59% of queries had an improved ranking in the comprehensive context condition, ref. Table 7b and Figure 5b), the improvement in terms of *MFR* was not significant. In addition, the use of contextual information in query augmentation did not help in improving retrieval quality for queries with navigational intent (ref. Table 7c). A similar finding has been reported in the prior work [14] in which query augmentation had no effect on searches with navigational intent.

RQ2. Does the source of contextual information affect Web search performance? Yes, the results in Experiment 2 show that contexts derived from different application sources affected retrieval performance but the results varied across the context windows and the search intents.

Finding 3: We found that the different application sources of context information are all important, but we did not find differences to support any specific source that would improve performances across different queries. Therefore, it seems that the user context should not be limited to the information available on the search systems themselves, but there are many equally good sources of contextual information that can be leveraged for query augmentation. However, the best-performing contextual sources for modeling varied in their performance with respect to temporal length of context. Different context windows should be assigned to each source depending on whether

Table 11. The results for navigational intent pertaining to the retrieval performance of application-specific context models.

	MAP			*MFR*		
context window	10 minutes	1 hour	1 day	10 minutes	1 hour	1 day
Init (control) (i)	0.238 (0.085)			44.2 (10.2)		
QAug (a)	0.177 (0.084)			44.9 (11.0)		
Search (s)	0.181 (0.073)	0.190 (0.079)	0.189 (0.076)	39.7 (11.7)	37.8 (12.8)	37.4 (15.1)
Social	0.184 (0.079)	0.200 (0.082)	0.192 (0.080)	39.4 (12.5)	39.3 (11.1)	39.3 (15.6)
vs. (i) p-value	*0.6*	*1*	*1*	*1*	*0.1*	*1*
vs. (a) p-value	*1*	*1*	*1*	*1*	*1*	*1*
vs. (s) p-value	*1*	*1*	*1*	*1*	*1*	*1*
Office	0.193 (0.076)	0.208 (0.086)	0.197 (0.086)	39.6 (10.5)	36.8 (13.2)	38.1 (15.4)
vs. (i) p-value	*1*	*1*	*1*	*0.1*	*0.07*	*0.9*
vs. (a) p-value	*1*	*0.3*	*0.1*	*1*	*0.8*	*1*
vs. (s) p-value	*1*	*1*	*1*	*1*	*1*	*1*
Local	0.187 (0.074)	0.196 (0.086)	0.193 (0.085)	38.9 (11.1)	38.1 (12.4)	35.4 (12.9)
vs. (i) p-value	*1*	*1*	*1*	*0.3*	*0.06*	*0.06*
vs. (a) p-value	*1*	*1*	*1*	*0.3*	*1*	*0.2*
vs. (s) p-value	*1*	*1*	*1*	*1*	*1*	*1*
E-commerce	0.199 (0.082)	0.213 (0.092)	0.194 (0.086)	38.2 (12.2)	37.3 (11.9)	40.2 (15.7)
vs. (i) p-value	*1*	*1*	*1*	*0.06*	*0.06*	*1*
vs. (a) p-value	*1*	*0.6*	*1*	*1*	*0.5*	*1*
vs. (s) p-value	*1*	*1*	*1*	*1*	*1*	*1*
Static Web	0.199 (0.086)	0.194 (0.075)	0.194 (0.081)	38.4 (12.0)	38.5 (12.6)	38.6 (13.9)
vs. (i) p-value	*1*	*1*	*1*	*0.1*	*0.2*	*0.7*
vs. (a) p-value	*1*	*1*	*1*	*0.4*	*1*	*3*
vs. (s) p-value	*1*	*1*	*1*	*1*	*1*	*1*
Rare Web	0.210 (0.081)	0.213 (0.082)	0.196 (0.085)	40.5 (11.4)	39.5 (12.9)	40.2 (14.5)
vs. (i) p-value	*1*	*1*	*1*	*1*	*1*	*1*
vs. (a) p-value	*0.2*	*1*	*1*	*1*	*1*	*1*
vs. (s) p-value	*1*	*1*	*1*	*1*	*1*	*1*

the system is contextualizing a Web search or recommending documents that are relevant to the immediate situation, the current work task, or the user's long-term interests. For example, Table 8 demonstrates that a certain amount of contexts with a particular application source are required to obtain high retrieval performance (e.g., the Social and Office application models should use short-term behavior, whereas the E-commerce/learning model should use long-term behavior).

Finding 4: The observed variation in model performance for each of the three context windows suggests that different sources of contextual information may be suited to different queries. For example, for a query with informational intent, the search system should leverage short-term contextual information sourced from email conversations, local documents, or one-time Webpage visits. One possibility is that, because the goal of informational tasks is to find useful information for the current primary task, in which the information provided by contextual snippets of the task is more relevant for queries with informational intents, such as details of conversations or editing documents would be related to the site content that the user would want to locate from the search results. This finding aligns with the early finding [35], in which previously seen documents before

Fig. 9. The results for navigational intent pertaining to the percentage of queries in application-specific context conditions with ranks of last-clicked documents changed from the control condition (ref. *MFR* in Table 11)

the query while the user engaged in an information-intensive task could be used to model search intent, leading to improved quality of retrieval results. However, if the system needs to better understand, model, and serve searchers' information needs for a query with transactional intent, long-term behaviors from online learning, e-commerce transactions, office productivity software, or static Web pages should be used.

10 DISCUSSION AND CONCLUSION

We studied the effectiveness of different sources of information and their lengths for context modeling and query augmentation. The data were collected using a novel methodology via the continuous capturing of participants' computer screens over 14 days. Our research improves our understanding of the importance of different types of context and application sources in retrieval performance.

10.1 Implications

The implications of our findings are striking, as they reveal interdependencies between the retrieval performance of query augmentation and the type of contextual signals (long- or short-term, various application sources) used to construct the predictive model. We foresee implications for using contextual information to support researchers in designing user studies and experiments, and for practitioners designing information access systems, as well as privacy preservation strategies.

10.1.1 Designing user studies and experiments. The results indicate that simple search history and Web browsing actions may not be representative of users' real underlying behaviors. This may have set limits on the current experimental paradigm and the datasets used in information retrieval experimentation. Our findings showed that user search intent within ten minutes could be predicted by various contextual information sourced from local documents and applications, suggesting that user topical interest may not change within a short period of time. The high effectiveness of comprehensive context in predicting user information needs within ten minutes may be due to its consideration of the current primary task. As comprehensive context contains more information than search context, it is more likely able to include more task information that could appear in other non-search applications. The effectiveness of comprehensive contexts in predicting longer-term search intent for queries with informational intent is likely due to the ability to access large amounts of long-term information of a user. Such knowledge can help researchers to design experiments with more face validity. Given the assigned work task and its potential queries, contextual evidence sourced from different local applications and Web services can be leveraged and an appropriate duration for the task can be established accordingly to improve the lab-based experiments.

10.1.2 Designing information access systems. There are differences between query intents and applications that affect the kind of context useful for modeling user information needs. Our findings showed that there are distinct query sets for which different context models and sources perform most effectively, suggesting that query information is likely important in selecting sources and temporal contextual lengths. The richer models that we developed can be used to interpret a user's search intent for a wide variety of search applications, including proactively retrieving information of likely interest to the user, suggesting useful queries contextually, or document ranking and filtering. Search systems could also use the context model and assign a source and context window based on the query to improve the quality of search rankings, by promoting results that are consistent with the inferred user intent. The systems may need to vary the sources depending on the modeling task, e.g., short-term models should use recently opened documents, emails, and instance messages; whereas long-term models should use information from learning activities and historical online transactions.

10.1.3 Designing privacy preservation strategies. Our research has implications for the implementation of data minimization and retention requirements of some regulations for data protection and privacy. In particular, our study shows that, in most cases, only short-term behavior from a limited set of applications is sufficient for effective query augmentation. For queries with navigational intent, more specifically, using contextual information does not always yield an improvement. For other types of queries, short-term behaviors from a wider range of applications may yield benefits. Longer-term behaviors from many applications, however, may not be needed and may even negatively affect the performance. As previous research has demonstrated [68], users are often reluctant to share information with a search system and they do not understand why certain auto-complete queries have been suggested or where they came from. Future work on search systems may consider providing users with various privacy thresholds and their explanations for suggestions, and allow them to freely explore the relationship between privacy preservation, contextual sources, and search quality. Furthermore, past work on privacy-aware web search [71, 74] largely focused on complex mechanisms that bundled the entire user profile and placed the user in a critical role for profiling (by allowing the users to choose the degree of detail of their profile information to be exposed to the search engine). Our study indicates that a much simpler approach that leaves out some aspects of the user activity for profiling could offer similar acceptable search quality.

10.2 Limitations & Future Work

This study presents some limitations which may have some impact on the generalizability of the results. This study was carried out as part of a field study, and because of this, followed an unusual experimental procedure. Two of the more important aspects of this procedure that might have impacted our results are that some of the user behaviors may have been restrained on purpose, due to the fact that users were fully aware that their digital activities were being tracked, and the two-week monitoring period may not reflect the entire blueprint of their behaviors, including activity changes due to seasons/holidays or monthly routines. Furthermore, while the 14-day digital activity monitoring of 13 participants resulted in large data, consisting of nearly 250,000 screen frames, it was a fairly small sample compared to what could be sourced from longer term instrumentation with a larger population. Although the potential impact of the experimental procedure and recruitment on the validity and reliability of our results cannot be ignored, we feel that our results are important and make an essential contribution to the research on query augmentation and user modeling.

The observed differences in this study may be related to the nature of the sources that were selected. For example, it may have been better to also include user behaviors on smartphones and spoken conversations between people in the analysis. However, given that this study was computer-based, and that we had extracted all available texts in all applications, the definitions of context we adopted here seem reasonable.

Another limitation is the use of the non-personalized Bing Search API as a control condition. There could be other machine learning models that perform better than the one Bing service employs. Our aim was not to propose a new model which would need comparison with other state-of-the-art approaches, rather it was to study the effect of the more contextual information and application sources in improving query augmentation. Here, we are more interested in the effects of the different data inputs instead of a particular model's performance. Therefore, every other variable (e.g., model choices or the impact of long-term user behavior prior to the experiments) was kept the same in all conditions. Comparing different machine learning models is a subject for future work.

Future work is also necessary to determine how best to combine sources or use them separately, including using unsupervised machine learning to automatically determine source weights. Other types of context, such as short- and long-term spoken conversations between people [62], or user profiles (gender, age, ethnicity, locality), can be studied in personal and collaborative search scenarios [2]. The importance of other situational contexts, such as user device or location can also be explored in the future. Another direction is to deepen our findings. For this, we suggest a larger data set collected to statistically analyze the explanatory power of different variables.

11 ACKNOWLEDGMENTS

This research was funded by the project COADAPT (Human and Work Station Adaptation Support to aging citizens, grant agreement No. 826266) and the project PON AIM (id: AIM1875400-1, CUP: B74I18000210006), and was partially supported by the Academy of Finland (Flagship programme: Finnish Center for Artificial Intelligence FCAI and decision numbers: 322653, 328875, 336085). We would like to thank Kenneth Quek for his kind proofreading of the manuscript.

REFERENCES

[1] Mohannad ALMasri, Catherine Berrut, and Jean-Pierre Chevallet. 2016. A Comparison of Deep Learning Based Query Expansion with Pseudo-Relevance Feedback and Mutual Information. In *Advances in Information Retrieval*, Nicola Ferro, Fabio Crestani, Marie-Francine Moens, Josiane Mothe, Fabrizio Silvestri, Giorgio Maria Di Nunzio, Claudia Hauff, and Gianmaria Silvello (Eds.). Springer International Publishing, Cham, 709–715.

[2] Salvatore Andolina, Valeria Orso, Hendrik Schneider, Khalil Klouche, Tuukka Ruotsalo, Luciano Gamberini, and Giulio Jacucci. 2018. Investigating Proactive Search Support in Conversations. In *Proceedings of the 2018 Designing Interactive Systems Conference* (Hong Kong, China) *(DIS '18)*. Association for Computing Machinery, New York, NY, USA, 1295–1307. https://doi.org/10.1145/3196709.3196734

[3] Ioannis Antonellis, Hector Garcia Molina, and Chi Chao Chang. 2008. Simrank++: Query Rewriting through Link Analysis of the Click Graph. *Proc. VLDB Endow.* 1, 1 (Aug. 2008), 408–421. https://doi.org/10.14778/1453856.1453903

[4] Charles E. Antoniak. 1974. Mixtures of Dirichlet Processes with Applications to Bayesian Nonparametric Problems. *Ann. Statist.* 2, 6 (11 1974), 1152–1174. https://doi.org/10.1214/aos/1176342871

[5] Doug Beeferman and Adam Berger. 2000. Agglomerative Clustering of a Search Engine Query Log. In *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Boston, Massachusetts, USA) *(KDD '00)*. ACM, New York, NY, USA, 407–416. https://doi.org/10.1145/347090.347176

[6] Nicholas J. Belkin, C. Cool, W. Bruce Croft, and James P. Callan. 1993. The Effect Multiple Query Representations on Information Retrieval System Performance. In *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Pittsburgh, Pennsylvania, USA) *(SIGIR '93)*. Association for Computing Machinery, New York, NY, USA, 339–346. https://doi.org/10.1145/160688.160760

[7] Paul N. Bennett, Ryen W. White, Wei Chu, Susan T. Dumais, Peter Bailey, Fedor Borisyuk, and Xiaoyuan Cui. 2012. Modeling the Impact of Short- and Long-Term Behavior on Search Personalization. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Portland, Oregon, USA) *(SIGIR '12)*. Association for Computing Machinery, New York, NY, USA, 185–194. https://doi.org/10.1145/2348283.2348312

[8] Bodo Billerbeck and Justin Zobel. 2004. Techniques for Efficient Query Expansion. In *String Processing and Information Retrieval*, Alberto Apostolico and Massimo Melucci (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 30–42.

[9] Andrei Broder. 2002. A Taxonomy of Web Search. *SIGIR Forum* 36, 2 (Sept. 2002), 3–10. https://doi.org/10.1145/792550.792552

[10] Andrei Broder, Peter Ciccolo, Evgeniy Gabrilovich, Vanja Josifovski, Donald Metzler, Lance Riedel, and Jeffrey Yuan. 2009. Online Expansion of Rare Queries for Sponsored Search. In *Proceedings of the 18th International Conference on World Wide Web* (Madrid, Spain) *(WWW '09)*. Association for Computing Machinery, New York, NY, USA, 511–520. https://doi.org/10.1145/1526709.1526778

[11] Georg Buscher, Andreas Dengel, and Ludger van Elst. 2008. Query Expansion Using Gaze-Based Feedback on the Subdocument Level. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Singapore, Singapore) *(SIGIR '08)*. Association for Computing Machinery, New York, NY, USA, 387–394. https://doi.org/10.1145/1390334.1390401

[12] Guihong Cao, Jian-Yun Nie, Jianfeng Gao, and Stephen Robertson. 2008. Selecting Good Expansion Terms for Pseudo-Relevance Feedback. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Singapore, Singapore) *(SIGIR '08)*. Association for Computing Machinery, New York, NY, USA, 243–250. https://doi.org/10.1145/1390334.1390377

[13] Huanhuan Cao, Daxin Jiang, Jian Pei, Qi He, Zhen Liao, Enhong Chen, and Hang Li. 2008. Context-Aware Query Suggestion by Mining Click-through and Session Data. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Las Vegas, Nevada, USA) *(KDD '08)*. Association for Computing Machinery, New York, NY, USA, 875–883. https://doi.org/10.1145/1401890.1401995

[14] Claudio Carpineto and Giovanni Romano. 2012. A Survey of Automatic Query Expansion in Information Retrieval. *ACM Comput. Surv.* 44, 1, Article 1 (Jan. 2012), 50 pages. https://doi.org/10.1145/2071389.2071390

[15] Yongqiang Chen, Peng Zhang, Dawei Song, and Benyou Wang. 2015. A Real-Time Eye Tracking Based Query Expansion Approach via Latent Topic Modeling. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management* (Melbourne, Australia) *(CIKM '15)*. Association for Computing Machinery, New York, NY, USA, 1719–1722. https://doi.org/10.1145/2806416.2806602

[16] Paul Alexandru Chirita, Claudiu S. Firan, and Wolfgang Nejdl. 2007. Personalized Query Expansion for the Web. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Amsterdam, The Netherlands) *(SIGIR '07)*. ACM, New York, NY, USA, 7–14. https://doi.org/10.1145/1277741.1277746

[17] Hang Cui, Ji-Rong Wen, Jian-Yun Nie, and Wei-Ying Ma. 2002. Probabilistic Query Expansion Using Query Logs. In *Proceedings of the 11th International Conference on World Wide Web* (Honolulu, Hawaii, USA) *(WWW '02)*. ACM, New York, NY, USA, 325–332. https://doi.org/10.1145/511446.511489

[18] Fernando Diaz, Bhaskar Mitra, and Nick Craswell. 2016. Query Expansion with Locally-Trained Word Embeddings. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Berlin, Germany, 367–377. https://doi.org/10.18653/v1/P16-1035

[19] Nan Du, Mehrdad Farajtabar, Amr Ahmed, Alexander J. Smola, and Le Song. 2015. Dirichlet-Hawkes Processes with Applications to Clustering Continuous-Time Document Streams. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Sydney, NSW, Australia) *(KDD '15)*. ACM, New York, NY, USA,

219–228. https://doi.org/10.1145/2783258.2783411

[20] Miles Efron and Megan Winget. 2010. Questions are content: A taxonomy of questions in a microblogging environment. *Proceedings of the American Society for Information Science and Technology* 47, 1 (2010), 1–10. https://doi.org/10.1002/meet.14504701208 arXiv:https://asistdl.onlinelibrary.wiley.com/doi/pdf/10.1002/meet.14504701208

[21] Carsten Eickhoff, Sebastian Dungs, and Vu Tran. 2015. An Eye-Tracking Study of Query Reformulation. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Santiago, Chile) *(SIGIR '15)*. ACM, New York, NY, USA, 13–22. https://doi.org/10.1145/2766462.2767703

[22] Karim Filali, Anish Nair, and Chris Leggetter. 2010. Transitive History-Based Query Disambiguation for Query Reformulation. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Geneva, Switzerland) *(SIGIR '10)*. Association for Computing Machinery, New York, NY, USA, 849–850. https://doi.org/10.1145/1835449.1835647

[23] Bruno M. Fonseca, Paulo Golgher, Bruno Pôssas, Berthier Ribeiro-Neto, and Nivio Ziviani. 2005. Concept-Based Interactive Query Expansion. In *Proceedings of the 14th ACM International Conference on Information and Knowledge Management* (Bremen, Germany) *(CIKM '05)*. Association for Computing Machinery, New York, NY, USA, 696–703. https://doi.org/10.1145/1099554.1099726

[24] Vreixo Formoso, Diego FernáNdez, Fidel Cacheda, and Victor Carneiro. 2013. Using Profile Expansion Techniques to Alleviate the New User Problem. *Inf. Process. Manage.* 49, 3 (May 2013), 659–672. https://doi.org/10.1016/j.ipm.2012.07.005

[25] Norbert Fuhr. 2018. Some Common Mistakes In IR Evaluation, And How They Can Be Avoided. *SIGIR Forum* 51, 3 (Feb. 2018), 32–41. https://doi.org/10.1145/3190580.3190586

[26] Jianfeng Gao and Jian-Yun Nie. 2012. Towards Concept-Based Translation Models Using Search Logs for Query Expansion. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management* (Maui, Hawaii, USA) *(CIKM '12)*. Association for Computing Machinery, New York, NY, USA, Article 1, 10 pages. https://doi.org/10.1145/2396761.2530275

[27] Mihajlo Grbovic, Nemanja Djuric, Vladan Radosavljevic, Fabrizio Silvestri, and Narayan Bhamidipati. 2015. Context- and Content-Aware Embeddings for Query Rewriting in Sponsored Search. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Santiago, Chile) *(SIGIR '15)*. Association for Computing Machinery, New York, NY, USA, 383–392. https://doi.org/10.1145/2766462.2767709

[28] Yangyang Guo, Zhiyong Cheng, Liqiang Nie, Yinglong Wang, Jun Ma, and Mohan Kankanhalli. 2019. Attentive Long Short-Term Preference Modeling for Personalized Product Search. *ACM Trans. Inf. Syst.* 37, 2, Article 19 (Jan. 2019), 27 pages. https://doi.org/10.1145/3295822

[29] Alan G. Hawkes. 1971. Spectra of Some Self-Exciting and Mutually Exciting Point Processes. *Biometrika* 58, 1 (1971), 83–90. http://www.jstor.org/stable/2334319

[30] Yunlong He, Jiliang Tang, Hua Ouyang, Changsung Kang, Dawei Yin, and Yi Chang. 2016. Learning to Rewrite Queries. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management* (Indianapolis, Indiana, USA) *(CIKM '16)*. Association for Computing Machinery, New York, NY, USA, 1443–1452. https://doi.org/10.1145/2983323.2983835

[31] Giulio Jacucci, Pedram Daee, Tung Vuong, Salvatore Andolina, Khalil Klouche, Mats Sjöberg, Tuukka Ruotsalo, and Samuel Kaski. 2021. Entity Recommendation for Everyday Digital Tasks. *ACM Trans. Comput.-Hum. Interact.* 28, 5, Article 29 (Oct. 2021), 41 pages. https://doi.org/10.1145/3458919

[32] Rosie Jones, Benjamin Rey, Omid Madani, and Wiley Greiner. 2006. Generating Query Substitutions. In *Proceedings of the 15th International Conference on World Wide Web* (Edinburgh, Scotland) *(WWW '06)*. Association for Computing Machinery, New York, NY, USA, 387–396. https://doi.org/10.1145/1135777.1135835

[33] Diane Kelly, Vijay Deepak Dollu, and Xin Fu. 2005. The Loquacious User: A Document-Independent Source of Terms for Query Expansion. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Salvador, Brazil) *(SIGIR '05)*. Association for Computing Machinery, New York, NY, USA, 457–464. https://doi.org/10.1145/1076034.1076112

[34] Youngho Kim and W. Bruce Croft. 2014. Diversifying Query Suggestions Based on Query Documents. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval* (Gold Coast, Queensland, Australia) *(SIGIR '14)*. Association for Computing Machinery, New York, NY, USA, 891–894. https://doi.org/10.1145/2600428.2609467

[35] Markus Koskela, Petri Luukkonen, Tuukka Ruotsalo, Mats SjÖberg, and Patrik Floréen. 2018. Proactive Information Retrieval by Capturing Search Intent from Primary Task Context. *ACM Trans. Interact. Intell. Syst.* 8, 3, Article 20 (July 2018), 25 pages. https://doi.org/10.1145/3150975

[36] Reiner Kraft, Chi Chao Chang, Farzin Maghoul, and Ravi Kumar. 2006. Searching with Context. In *Proceedings of the 15th International Conference on World Wide Web* (Edinburgh, Scotland) *(WWW '06)*. Association for Computing Machinery, New York, NY, USA, 477–486. https://doi.org/10.1145/1135777.1135847

[37] Reiner Kraft and Jason Zien. 2004. Mining Anchor Text for Query Refinement. In *Proceedings of the 13th International Conference on World Wide Web* (New York, NY, USA) *(WWW '04)*. ACM, New York, NY, USA, 666–674. https://doi.org/10.1145/988672.988763

[38] Saar Kuzi, David Carmel, Alex Libov, and Ariel Raviv. 2017. Query Expansion for Email Search. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Shinjuku, Tokyo, Japan) *(SIGIR '17)*. Association for Computing Machinery, New York, NY, USA, 849–852. https://doi.org/10.1145/3077136.3080660

[39] Saar Kuzi, Anna Shtok, and Oren Kurland. 2016. Query Expansion Using Word Embeddings. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management* (Indianapolis, Indiana, USA) *(CIKM '16)*. Association for Computing Machinery, New York, NY, USA, 1929–1932. https://doi.org/10.1145/2983323.2983876

[40] Steve Lawrence. 2000. Context in Web Search. *IEEE Data Engineering Bulletin* 23 (2000), 25–32.

[41] Cheng Li, Mingyang Zhang, Michael Bendersky, Hongbo Deng, Donald Metzler, and Marc Najork. 2019. Multi-View Embedding-Based Synonyms for Email Search. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Paris, France) *(SIGIR'19)*. Association for Computing Machinery, New York, NY, USA, 575–584. https://doi.org/10.1145/3331184.3331250

[42] Yuan Lin, Hongfei Lin, Song Jin, and Zheng Ye. 2011. Social Annotation in Query Expansion: A Machine Learning Approach. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Beijing, China) *(SIGIR '11)*. Association for Computing Machinery, New York, NY, USA, 405–414. https://doi.org/10.1145/2009916.2009972

[43] Claudio Lucchese, Franco Maria Nardini, Raffaele Perego, Roberto Trani, and Rossano Venturini. 2018. Efficient and Effective Query Expansion for Web Search. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management* (Torino, Italy) *(CIKM '18)*. Association for Computing Machinery, New York, NY, USA, 1551–1554. https://doi.org/10.1145/3269206.3269305

[44] Parvaz Mahdabi, Shima Gerani, Jimmy Xiangji Huang, and Fabio Crestani. 2013. Leveraging Conceptual Lexicon: Query Disambiguation Using Proximity Information for Patent Retrieval. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Dublin, Ireland) *(SIGIR '13)*. Association for Computing Machinery, New York, NY, USA, 113–122. https://doi.org/10.1145/2484028.2484056

[45] Subhadeep Maji, Rohan Kumar, Manish Bansal, Kalyani Roy, Mohit Kumar, and Pawan Goyal. 2019. Addressing Vocabulary Gap in E-Commerce Search. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Paris, France) *(SIGIR'19)*. Association for Computing Machinery, New York, NY, USA, 1073–1076. https://doi.org/10.1145/3331184.3331323

[46] Kamran Massoudi, Manos Tsagkias, Maarten de Rijke, and Wouter Weerkamp. 2011. Incorporating Query Expansion and Quality Indicators in Searching Microblog Posts. In *Proceedings of the 33rd European Conference on Advances in Information Retrieval* (Dublin, Ireland) *(ECIR'11)*. Springer-Verlag, Berlin, Heidelberg, 362–367. http://dl.acm.org/citation.cfm?id=1996889.1996936

[47] Bhaskar Mitra. 2015. Exploring Session Context Using Distributed Representations of Queries and Reformulations. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Santiago, Chile) *(SIGIR '15)*. Association for Computing Machinery, New York, NY, USA, 3–12. https://doi.org/10.1145/2766462.2767702

[48] Apostol (Paul) Natsev, Alexander Haubold, Jelena Tešić, Lexing Xie, and Rong Yan. 2007. Semantic Concept-based Query Expansion and Re-ranking for Multimedia Retrieval. In *Proceedings of the 15th ACM International Conference on Multimedia* (Augsburg, Germany) *(MM '07)*. ACM, New York, NY, USA, 991–1000. https://doi.org/10.1145/1291233.1291448

[49] Fuchun Peng, Nawaaz Ahmed, Xin Li, and Yumao Lu. 2007. Context Sensitive Stemming for Web Search. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Amsterdam, The Netherlands) *(SIGIR '07)*. Association for Computing Machinery, New York, NY, USA, 639–646. https://doi.org/10.1145/1277741.1277851

[50] Matthew E. Peters and Dan Lecocq. 2013. Content Extraction Using Diverse Feature Sets. In *Proceedings of the 22nd International Conference on World Wide Web* (Rio de Janeiro, Brazil) *(WWW '13 Companion)*. Association for Computing Machinery, New York, NY, USA, 89–90. https://doi.org/10.1145/2487788.2487828

[51] Benjamin Piwowarski, Georges Dupret, and Rosie Jones. 2009. Mining User Web Search Activity with Layered Bayesian Networks or How to Capture a Click in Its Context. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining* (Barcelona, Spain) *(WSDM '09)*. Association for Computing Machinery, New York, NY, USA, 162–171. https://doi.org/10.1145/1498759.1498823

[52] Filip Radlinski and Thorsten Joachims. 2005. Query Chains: Learning to Rank from Implicit Feedback. In *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining* (Chicago, Illinois, USA) *(KDD '05)*. ACM, New York, NY, USA, 239–248. https://doi.org/10.1145/1081870.1081899

[53] Miamaria Saastamoinen and Kalervo Järvelin. 2016. Queries in authentic work tasks: the effects of task type and complexity. *Journal of Documentation* 72 (2016), 1114–1133.

[54] Xuehua Shen, Bin Tan, and ChengXiang Zhai. 2005. Context-Sensitive Information Retrieval Using Implicit Feedback. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Salvador, Brazil) *(SIGIR '05)*. Association for Computing Machinery, New York, NY, USA, 43–50. https://doi.org/10.1145/1076034.1076045

[55] Xuehua Shen, Bin Tan, and ChengXiang Zhai. 2005. Implicit User Modeling for Personalized Search. In *Proceedings of the 14th ACM International Conference on Information and Knowledge Management* (Bremen, Germany) *(CIKM '05)*. Association for Computing Machinery, New York, NY, USA, 824–831. https://doi.org/10.1145/1099554.1099747

[56] Gyanit Singh, Nish Parikh, and Neel Sundaresan. 2012. Rewriting Null E-Commerce Queries to Recommend Products. In *Proceedings of the 21st International Conference on World Wide Web* (Lyon, France) *(WWW '12 Companion)*. Association for Computing Machinery, New York, NY, USA, 73–82. https://doi.org/10.1145/2187980.2187989

[57] Sanasam Ranbir Singh, Hema A. Murthy, and Timothy A. Gonsalves. 2013. Inference Based Query Expansion Using User's Real Time Implicit Feedback. In *Knowledge Discovery, Knowledge Engineering and Knowledge Management*, Ana Fred, Jan L. G. Dietz, Kecheng Liu, and Joaquim Filipe (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 158–172.

[58] Julia Snell. 2011. Interrogating video data: systematic quantitative analysis versus micro-ethnographic analysis. *International Journal of Social Research Methodology* 14, 3 (2011), 253–258. https://doi.org/10.1080/13645579.2011.563624 arXiv:https://doi.org/10.1080/13645579.2011.563624

[59] Jiwei Tan, Xiaojun Wan, Hui Liu, and Jianguo Xiao. 2018. QuoteRec: Toward Quote Recommendation for Writing. *ACM Trans. Inf. Syst.* 36, 3, Article 34 (March 2018), 36 pages. https://doi.org/10.1145/3183370

[60] Jaime Teevan, Eytan Adar, Rosie Jones, and Michael A. S. Potts. 2007. Information Re-Retrieval: Repeat Queries in Yahoo's Logs. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Amsterdam, The Netherlands) *(SIGIR '07)*. Association for Computing Machinery, New York, NY, USA, 151–158. https://doi.org/10.1145/1277741.1277770

[61] Jaime Teevan, Susan T. Dumais, and Eric Horvitz. 2005. Personalizing Search via Automated Analysis of Interests and Activities. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Salvador, Brazil) *(SIGIR '05)*. Association for Computing Machinery, New York, NY, USA, 449–456. https://doi.org/10.1145/1076034.1076111

[62] Tung Vuong, Salvatore Andolina, Giulio Jacucci, and Tuukka Ruotsalo. 2021. Spoken Conversational Context Improves Query Auto-Completion in Web Search. *ACM Trans. Inf. Syst.* 39, 3, Article 31 (May 2021), 32 pages. https://doi.org/10.1145/3447875

[63] Tung Vuong, Giulio Jacucci, and Tuukka Ruotsalo. 2017. Watching inside the Screen: Digital Activity Monitoring for Task Recognition and Proactive Information Retrieval. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1, 3, Article 109 (Sept. 2017), 23 pages. https://doi.org/10.1145/3130974

[64] 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* 0, 0 (2019). https://doi.org/10.1002/asi.24201 arXiv:https://asistdl.onlinelibrary.wiley.com/doi/pdf/10.1002/asi.24201

[65] Ryen W. White, Peter Bailey, and Liwei Chen. 2009. Predicting User Interests from Contextual Information. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Boston, MA, USA) *(SIGIR '09)*. Association for Computing Machinery, New York, NY, USA, 363–370. https://doi.org/10.1145/1571941.1572005

[66] Ryen W. White, Paul N. Bennett, and Susan T. Dumais. 2010. Predicting Short-Term Interests Using Activity-Based Search Context. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management* (Toronto, ON, Canada) *(CIKM '10)*. Association for Computing Machinery, New York, NY, USA, 1009–1018. https://doi.org/10.1145/1871437.1871565

[67] Hao Wu and Hui Fang. 2013. An Incremental Approach to Efficient Pseudo-Relevance Feedback. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Dublin, Ireland) *(SIGIR '13)*. Association for Computing Machinery, New York, NY, USA, 553–562. https://doi.org/10.1145/2484028.2484051

[68] Yuxi Wu, Panya Gupta, Miranda Wei, Yasemin Acar, Sascha Fahl, and Blase Ur. 2018. Your Secrets Are Safe: How Browsers' Explanations Impact Misconceptions About Private Browsing Mode. In *Proceedings of the 2018 World Wide Web Conference* (Lyon, France) *(WWW '18)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 217–226. https://doi.org/10.1145/3178876.3186088

[69] Jinxi Xu and W. Bruce Croft. 1996. Query Expansion Using Local and Global Document Analysis. In *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Zurich, Switzerland) *(SIGIR '96)*. Association for Computing Machinery, New York, NY, USA, 4–11. https://doi.org/10.1145/243199.243202

[70] Jinxi Xu and W. Bruce Croft. 2000. Improving the Effectiveness of Information Retrieval with Local Context Analysis. *ACM Trans. Inf. Syst.* 18, 1 (Jan. 2000), 79–112. https://doi.org/10.1145/333135.333138

[71] Yabo Xu, Ke Wang, Benyu Zhang, and Zheng Chen. 2007. Privacy-Enhancing Personalized Web Search. In *Proceedings of the 16th International Conference on World Wide Web* (Banff, Alberta, Canada) *(WWW '07)*. Association for Computing Machinery, New York, NY, USA, 591–600. https://doi.org/10.1145/1242572.1242652

[72] Wei Vivian Zhang and Rosie Jones. 2007. Comparing Click Logs and Editorial Labels for Training Query Rewriting. In *WWW 2007*.

[73] Zhiwei Zhang, Qifan Wang, Luo Si, and Jianfeng Gao. 2016. Learning for Efficient Supervised Query Expansion via Two-Stage Feature Selection. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Pisa, Italy) *(SIGIR '16)*. Association for Computing Machinery, New York, NY, USA, 265–274. https://doi.org/10.1145/2911451.2911539

[74] Yun Zhu, Li Xiong, and Christopher Verdery. 2010. Anonymizing User Profiles for Personalized Web Search. In *Proceedings of the 19th International Conference on World Wide Web* (Raleigh, North Carolina, USA) *(WWW '10)*. Association for Computing Machinery, New York, NY, USA, 1225–1226. https://doi.org/10.1145/1772690.1772886

A APPENDIX

Table 12. Retrieval performance in terms of *MAP* and *MFR*. Statistically significant differences with the initial ranking (control), the language model, and search history model, non-search history model; and 10-minute, 1-hour, and 1-day context sizes are marked with "*i*", "*a*", "*s*", "*n*", "*m*", "*h*", "*d*" respectively.

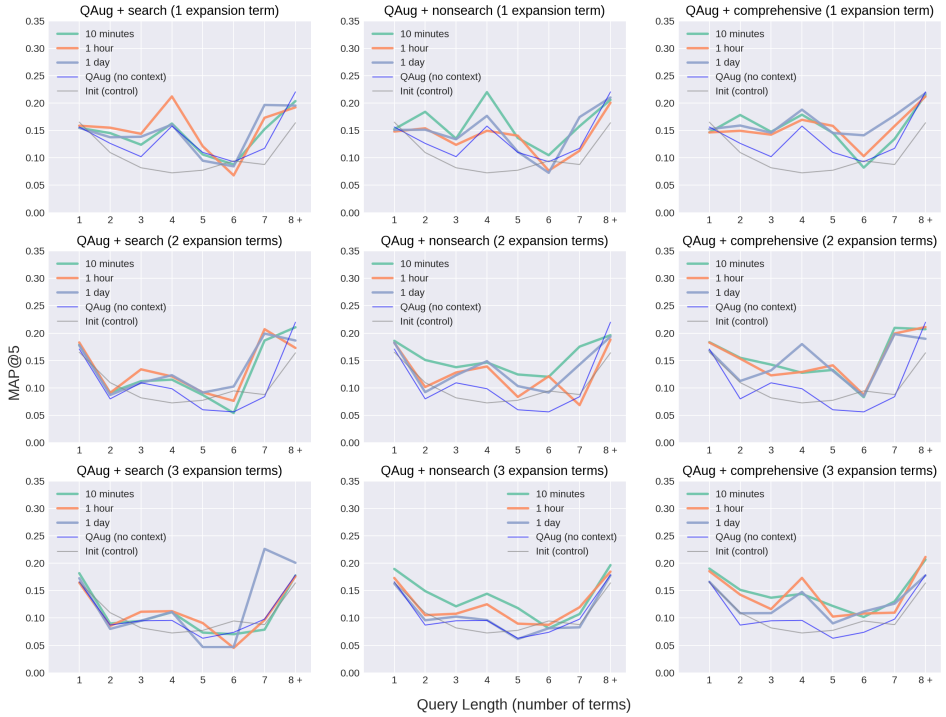
expansion terms	*1 term*			*2 terms*			*3 terms*		
context window	*10m*	*1h*	*1d*	*10m*	*1h*	*1d*	*10m*	*1h*	*1d*
Init (control)	.121								
QAug (a)	.142			.125			.115		
Search (s)	$\mathbf{.155}^{i}$	$\mathbf{.168}^{i}$	$\mathbf{.161}^{i}$	.139	.142	.141	.120	.130	.130
Nonsearch	$\mathbf{.180}_{s}^{i,a}$	$\mathbf{.165}^{i}$	$\mathbf{.163}^{i}$	$\mathbf{.163}^{i,a}$	.145	.140	$\mathbf{.149}^{a,d}$	.129	.115
Comprehensive	$\mathbf{.179}_{s}^{i,a}$	$\mathbf{.178}^{i,a}$	$\mathbf{.173}^{i,a}$	$\mathbf{.167}_{s}^{i,a}$	$\mathbf{.159}^{i,a}$	$\mathbf{.151}^{i}$	$\mathbf{.157}_{s,d}^{i,a}$	$\mathbf{.149}^{a}$	.133

(a) *MAP*

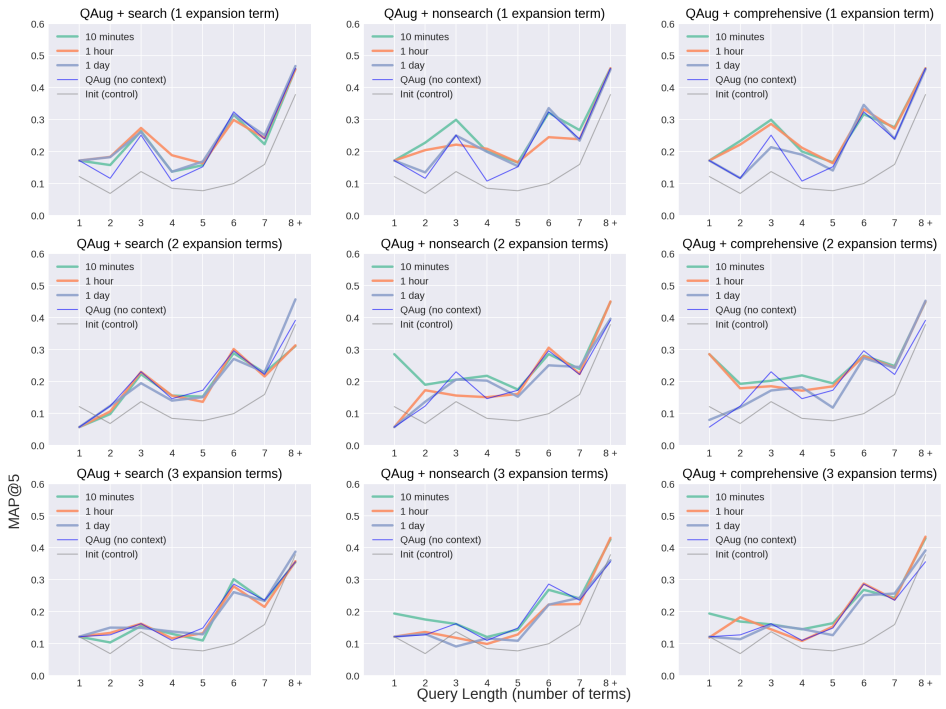
expansion terms	*1 term*			*2 terms*			*3 terms*		
context window	*10m*	*1h*	*1d*	*10m*	*1h*	*1d*	*10m*	*1h*	*1d*
Init (control)	58.2								
QAug (a)	50.7			59.3			63.1		
Search	$\mathbf{45.5}^{i}$	$\mathbf{44.4}^{i}$	$\mathbf{45.1}^{i}$	52.3	51.1	51.5	57.3	$\mathbf{53.5}^{a}$	$\mathbf{54.1}^{a}$
Nonsearch	$\mathbf{41.2}^{i,a}$	$\mathbf{43.8}^{i}$	$\mathbf{42.4}^{i}$	$\mathbf{48.2}^{a}$	$\mathbf{50.3}^{a}$	50.7	$\mathbf{49.5}^{l}$	$\mathbf{54.7}^{a}$	55.1
Comprehensive	$\mathbf{39.8}^{i,a}$	$\mathbf{40.1}^{i,a}$	$\mathbf{42.1}^{i,a}$	$\mathbf{45.7}_{s}^{i,a}$	$\mathbf{46.7}^{a}$	$\mathbf{46.1}^{i,a}$	$\mathbf{47.1}^{i,a}$	$\mathbf{47.4}^{i,a}$	$\mathbf{52.5}^{a}$

(b) MFR

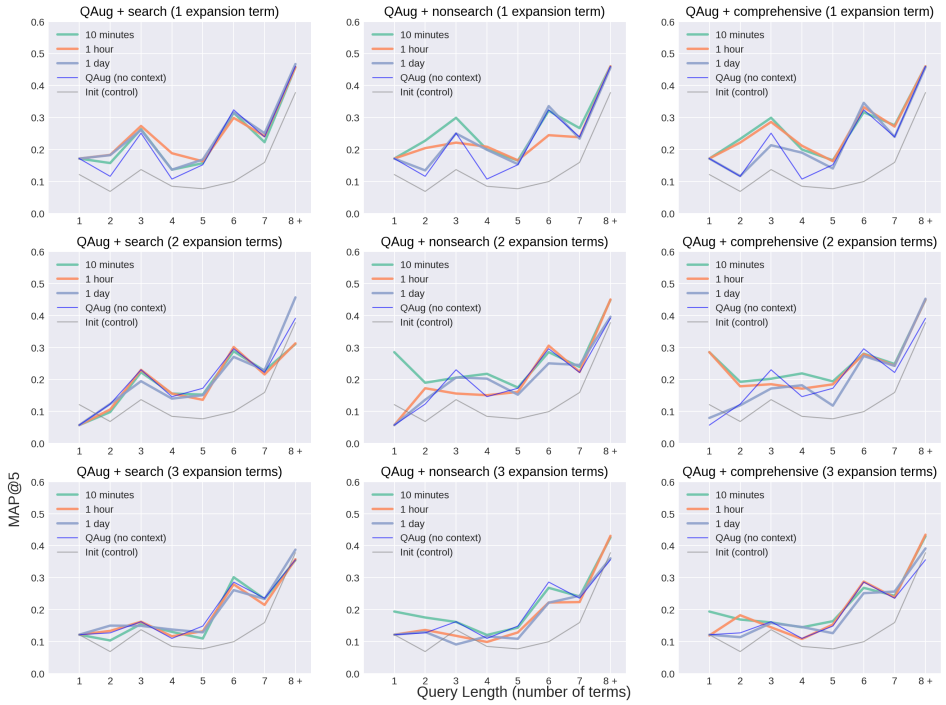
Fig. 10. Performance in terms of *MAP* by query length



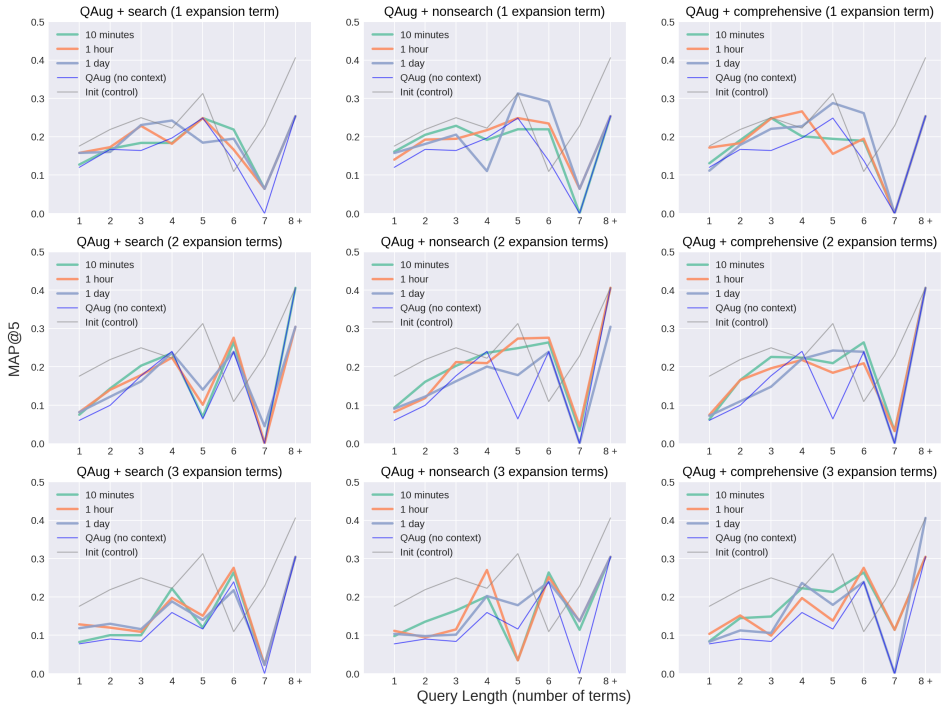
(a) Overall Performance



(b) Informational Queries



(c) Transactional Queries



(d) Navigational Queries