Proactive Information Retrieval by Capturing Search Intent from Primary Task Context

MARKUS KOSKELA, PETRI LUUKKONEN, TUUKKA RUOTSALO, MATS SJÖBERG, and PATRIK FLORÉEN,
Helsinki Institute for Information Technology HIIT
Department of Computer Science, University of Helsinki, Finland

A significant fraction of information searches are motivated by the user's *primary task*. An ideal search engine would be able to use information captured from the primary task in order to proactively retrieve useful information. Previous work has shown that many information retrieval activities depend on the primary task in which the retrieved information is to be used, but fairly little research has been focusing on methods that automatically learn the informational intents from the primary task context. We study how the implicit primary task context can be used to model the user's search intent and to proactively retrieve relevant and useful information. Data comprising of logs from a user study, in which users are writing an essay, demonstrate that users' search intents can be captured from the task and relevant and useful information can be proactively retrieved. Data from simulations with several data sets of different complexity show that the proposed approach of using primary task context generalizes to a variety of data. Our findings have implications for the design of proactive search systems that can infer users' search intent implicitly by monitoring users' primary task activities.

CCS Concepts: •**Information systems** → *Query intent; Users and interactive retrieval;* •**Human-centered computing** → *Text input;*

Additional Key Words and Phrases: Task-based Information Retrieval; Proactive Search; User Intent Modeling

ACM Reference Format:
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.* V, N, Article A (July 2017), 27 pages.
DOI: 0000001.0000001

1. INTRODUCTION

Search engines are used to find information that helps us in our daily tasks, be they leisure or professional. An ideal search engine would support the user in identifying useful information that can be used in solving these *primary tasks* the user is performing [Vakkari 2001]. A primary task can be any cognitively complex process performed by humans that may invoke information needs – the real world task, which is the main task one has to carry out [Belkin and Croft 1992; Li and Belkin 2008]. For example, a user may be writing or reading a document and needs to use related information in order understand what is being read or to support the writing process [Melguizo et al. 2009; Rhodes and Starner 1996; Vakkari 2001]. A prime reason requiring a user to interrupt the primary task is because the available information or users knowl-

This work is supported by TEKES (project Revolution of Knowledge Work).
Contact author: Markus Koskela, email: *markus.koskela@csc.fi*, currently affiliated with CSC - IT Center for Science Ltd.
Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
© 2017 ACM. 2160-6455/2017/07-ART A $15.00
DOI: 0000001.0000001

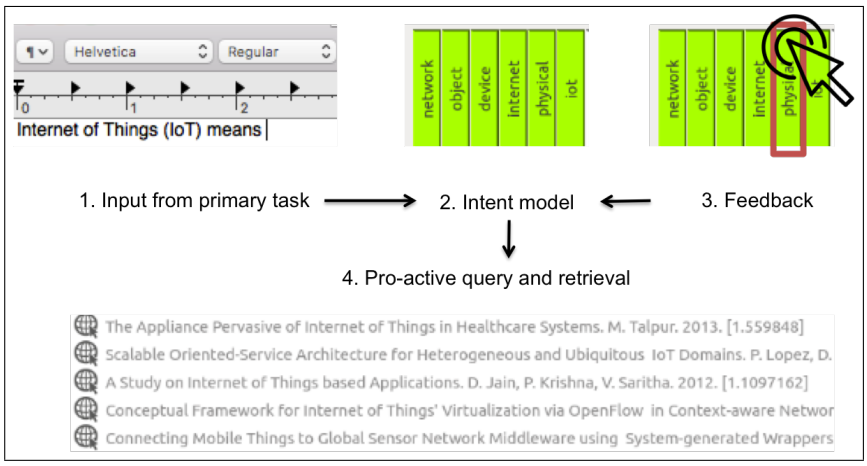

Fig. 1. An illustration of our approach. 1) The user carries out a primary task of writing a document about *Internet of Things*. 2) The text of the primary task is an input for the intent modelling method, which estimates a weighted vector of keywords representing the primary task context. 3) The user can interact with the model to improve the estimate. 4) The model is used to proactively retrieve information related to the primary task.

edge are inadequate to complete the task [Belkin et al. 1982; Ingwersen and Järvelin 2005]. This prompts the user to seek supporting information and triggers an information need. Information retrieval is based on information needs operationalized as search task actions, such as queries or following links. However, it is widely agreed that search tasks are part of broader user activity, often referred as work tasks or primary tasks [Byström and Hansen 2005; Li and Belkin 2008].

Proactive information retrieval refers to a method or a system that can retrieve information implicitly without requiring explicit attention or interactions from the user beyond the primary task [Dumais et al. 2004a]. Proactive retrieval can be implemented by modelling the user's search intent based on observing the primary task and instantly retrieve information according to the model without the user having to actively formulate explicit search queries.

Consequently, a central challenge in proactive information retrieval is to estimate user's search intent from limited and indirect primary task input. Previous research has shown that pre-search context [Kong et al. 2015] and user's tasks [Saastamoinen and Järvelin 2016] can significantly affect user's search performance. The motivation for further investigating primary task context is twofold. First, to understand whether a user's primary task, in contrast to just pre-search browsing history, can be used to predict search intentions. Second, to study if users utilize proactive search in realistic tasks and detect the trade-offs that such a system might cause.

In this article we study how the primary task context can be used to implicitly model the user's search intent and to proactively retrieve relevant and useful information. Since the primary task context triggering information needs can be noisy, we propose a predictive model that combines the primary task input with predictions drawn from a model that captures domain information. For example, in the case of primary task input "Internet of Things", the retrieval is not be based only on the phrase "internet of things", but also intentional aspects relevant to Internet of Things, such as "networks", "devices", or "physical systems". These aspects are derived from the predictive model.

Our approach is illustrated in the example in Figure 1, where a user is engaged with a primary task of writing a document and the model captures the user's intentional

aspects and retrieves information accordingly. Following this intuition, we investigate three research questions:

R1: How can the primary task context be used to model the user's intent sufficiently accurately to retrieve resources *relevant* to the primary task?
R2: Can the proactive information retrieval tool produce *useful* resources in a way that better supports the user in performing the primary task?
R3: Can the retrieval results be improved by user interaction with the proactive retrieval results?

In order to answer the research questions, we report two experiments: large-scale simulations and a user study. The simulations used document text from several standard data sets as a primary task context and measured the benefits of the intent prediction model for retrieval effectiveness. The user study compared proactive features with a conventional typed query interaction in a task where users were writing an essay on a given topic and were asked to select materials from search engine results supporting the writing task. We use writing as the primary task in the experiments in this article. However, the proposed model is not limited to writing, as the context can be observed from various primary task contexts.

The results of the simulation and the user study show that using the primary task context is effective in inferring an intent model that can be successfully used in proactive information retrieval. The users found proactive search useful; a large portion of the information selected by the users for the primary task resulted from proactively retrieved documents. Significant differences were not found in the number of explicit queries between the proposed method and a conventional interface, but two trade-offs were observed. First, a trade-off between precision and recall where proactive retrieval opts for recall over precision. The decrease in retrieval precision is a natural consequence of the proactive and explorative nature of the proposed method. Second, a trade-off in visited and selected documents where, despite the lower precision of proactive suggestions, the majority of documents the users selected or visited were in fact retrieved proactively.

Our contributions can be summarized as follows:

— We present a proactive information retrieval approach that utilizes the primary task context to capture the user's search intentions.
— We demonstrate through simulation that our method can successfully predict implicit search intents.
— We show an improved recall of proactively retrieved information with the cost of reduced precision.
— We show that proactive information retrieval can reduce manual search effort in completing the primary task.

The rest of the article is organized as follows. In the following section a discussion of related work is provided, and in Section 3 we describe the proposed intent modelling method and our proactive search system. In Section 4 we describe our simulation experiments and in Section 5 we describe our user study. We conclude and discuss future work in Section 6.

2. RELATED WORK

Information retrieval has traditionally relied on the explicit query paradigm, i.e., the retrieval session is initiated by the user making an explicit textual search query. A well-known example are web search engines, such as Google or Microsoft Bing. This approach is, however, limited, as a short query text often will not fully convey the user's search intent. For example, does the query "jaguar" refer to the brand of cars or the

animal? Several extensions have thus been proposed which try to infer the user's intent by taking into account additional context information, or by correlating with other users' search behavior [Xie 2000; Fu 2010]. For example, Cao et al. [2009] use previous search queries by the same user as context, e.g., if your search query was "BMW" that gives a strong indication of the search intent of your subsequent "jaguar" query. Xiang et al. [2010] further refined this idea by considering different kinds of relations between the subsequent queries, for example reformulation of the query, or narrowing of the search scope. Another approach is to try to identify other users with similar search intent [Agichtein et al. 2006] by analyzing the logs of search engines. A more sophisticated version based on identifying search tasks was proposed by White et al. [2013]. A similar approach is personalized search [Sontag et al. 2012], in which only the user's own search history is used, which can give more specific and personalized results. A comprehensive study can be found in [White et al. 2009], where several different types of context is studied, including historic, interaction and social context (i.e., other user's search interests). Another empirical investigation focuses on the role of multitasking, cognitive coordination, and cognitive shifts during information search [Du and Spink 2011]. They found that task-switching causes users to distribute their attentional resources and may cause interruptions in the search process.

2.1. Personalization and Recommender Systems in Interactive Information Retrieval

Search personalization leveraging contextual information about the users and their environment, such as search and interaction histories, demographics, geography, or sensor-based context can provide tailored search experiences for their users [Teevan et al. 2010]. Recently, there has been a growing focus on blending search and recommendation features to support personalization. This line of research has aimed at better supporting information search by modeling user interests and intentions by utilizing both content and social information, but also interactions that occur during search tasks. Today, recommender systems face analogous challenges, including integrating signals from users to update recommendations on-line.

The majority of the work on modeling context and search personalization has focused on constructing topical profiles of the users short- and long-term search history [Speretta and Gauch 2005; Chirita et al. 2005; Ma et al. 2007; White et al. 2010; Bennett et al. 2012; Sontag et al. 2012; Chirita et al. 2007], models of their query and result-click sequences [Cao et al. 2008; Joachims 2002], or explicit interactions with search user interfaces that provide recommendation functionality to direct the search [Ruotsalo et al. 2013; Ahn et al. 2007].

A variety of recent investigations to contextualize search have emphasized a users task-based search activity [Jones and Klinkner 2008; Kanoulas et al. 2011; Melucci 2012]. Contrary to our approach, many recent works have often focused on optimizing and evaluating session with fairly simple interactions: single queries or short sessions where a few subsequent typed queries, even though a significant fraction of interactions with search systems are associated with more complex tasks [Jones and Klinkner 2008; Liu and Belkin 2010; Raman et al. 2014], which span one or more search sessions and are related to users primary tasks rather than being isolated search activity [Ingwersen and Järvelin 2005; Vakkari 2003]. Researchers have also highlighted the importance of interaction support for more complex tasks [Vakkari 2003], whole-session relevance [Raman et al. 2013], and task performance beyond session boundaries [Liao et al. 2012].

In addition to conventional search systems, recommender systems have become increasingly important in information access [Resnick and Varian 1997; Adomavicius and Tuzhilin 2008] Recommender systems gather information from a given user and the users context [Adomavicius and Tuzhilin 2008]; create and update the users pro-

file; and without requiring explicit user queries, recommend information tailored to the users profile. Like contextual search, the next generation of recommender systems faces many of the same challenges of incorporating heterogeneous contexts into recommendations, as well as an analogous challenge of incorporating the interactive process of the users exploration in a single session to contextually update recommendations on-line in response to user interactions [Mahmood and Ricci 2007; Bostandjiev et al. 2012; Hariri et al. 2014; He et al. 2016; Orso et al. 2017].

Unlike previous research on search personalization, our approach is proactive relying only on the interactions observable from the primary user task. On the other hand, our approach relies solely on user specific data from the task. This is in contrast to reliance on similarities across users as in collaborative filtering systems.

2.2. Proactive Search

Personalized search and recommender systems to support search activity already analyze the user's context and history in order to improve explicit queries. The step from supporting explicit querying to continuously monitor and analyze the context in order to *anticipate* the upcoming information needs is fairly straightforward. This information retrieval paradigm is sometimes called *interactively predicting search intent* [Ruotsalo et al. 2014; Cheng et al. 2010], *anticipatory search* [Liebling et al. 2012] or *proactive search* [Elliott and Jose 2009]. An early proactive search setup can be found in the *Remembrance Agent* [Rhodes and Starner 1996], which indexes a user's personal data, such as emails and written notes. The system runs continuously in the background and displays a list of summaries of documents that are related to the current document being read or written. The interface is designed to be unobtrusive, allowing the user either to pursue the recommended documents or to ignore them. *Letizia* [Lieberman 1995] is another early system that provides proactive recommendations during web browsing using a set of heuristic rules. Commercially deployed examples include *Google Now* and *Microsoft Cortana*, which run on the users' smart phones and provide resources based the current context. In particular Google Now tries to model not only short-term search intents, but also long-term interests and habits based on several months of search log data [Guha et al. 2015]. Another example of using search history is proposed in Song and Guo [2016], where patterns repeated over time are extracted and used to proactively recommend resources to the user at specific times of the day. Vuong et al. propose a system where the search intentions are automatically inferred from user behavior implicitly captured via machine vision [Vuong et al. 2017b; 2017a]. Another somewhat surprising source of context for proactive search is subtitling of live TV broadcasts being viewed by the user [Henzinger et al. 2005].

Several authors have studied the correlation between web browsing behavior and consecutive searches, in order to anticipate the next query based on viewed web pages [Cheng et al. 2010; Liebling et al. 2012]. Kong et al. [2015] found several important characteristics of the web browsing context which typically cause users to perform further search queries. Perhaps the most interesting of these is the fact that news articles often trigger completely novel queries. This points to a drawback of the collaborative filtering approach, which bases the query prediction on previously seen patterns by other users. To address this issue, and also to ensure interesting non-obvious query predictions, Bordino et al. [2013] propose to use just the content of the current web page as the contextual information. They introduce a graph model to transform the web page content, and Wikipedia pages extracted from it, to an enriched set of query suggestions.

A task scenario that has often been studied in the context of proactive search or recommendation is writing. Here the context is typically the text written by the user

into the word processing software, and thus while there is textual input, there is no explicit query. This is a particularly challenging scenario as writing is cognitively demanding for the user and the user needs to focus attentional resources to the primary task. This is a problem for proactive search as an automatically displayed and continuously updated list of resources may be disruptive for the user. The writing process is commonly modelled as having three stages: planning, review, and translation, i.e., transforming the writer's mental ideas into sentences [Hayes and Flower 1980]. The latter is the most cognitively challenging stage, and thus has the least tolerance for undesired disruptions [Melguizo et al. 2009]. Thus, typically most proactive writing aids target the planning and reviewing stages, such as various bibliographic tasks involved in writing scientific or professional texts. Examples of reference recommendation during writing include *Writer's Aid* [Babaian et al. 2002], *PIRA* [Twidale et al. 2008] and *CiteSight* [Livne et al. 2014]. Systems providing more general proactive queries while writing, include *Watson* [Budzik et al. 2001], which performs contextual text and image queries based on text written using Microsoft Word. Liu et al. [2013] demonstrate a peripheral information panel showing proactive search results to aid Powerpoint presentation authoring, and Dumais et al. [2004b] an email application which shows personal documents related to the current email being authored or read. In [Luukkonen et al. 2016] proactive recommendations for writing are produced by text input prediction using a long short-term memory (LSTM) network.

Our research follows the line of research in proactive search agents. Our approach, however, mitigates the requirement of only suggesting queries or other aids for the user, and our approach estimates the intent model which is continuously updated and search is performed automatically in the background.

2.3. Visualizing User Intent

A retrieval system based on modelling the user's intent can provide a visualization of the current estimate, and show a relevance feedback interface for the user to manipulate it into the desired direction. For example, the intent estimate may have gone in the wrong direction, or the user wants to explore another area of the document space. In [Baldonado and Winograd 1997], the user intent is termed information context, and is estimated as the current collection of search results. The system supports various methods of relevance feedback, e.g., query-by-example, finding related references, refining or enlarging the current query. The *YourNews* system [Ahn et al. 2007] allowed the user to directly edit the user model for online news filtering, e.g., by adding or removing keywords. *TaskSieve* [Ahn et al. 2008] had a similar approach, but fused the query and user models into a task model that was open to direct manipulation by the user. In *IntrospectiveView* [Bakalov et al. 2010] a user interest model is visualized by placing words on a circle surface, the closer to the centre the higher the degree of interest. Different degrees are visualized as color coded rings. The user can manipulate the degree of interest by dragging the word to another ring in the circle. The circle is split into sectors according to different types of concepts, such as people, locations, and companies. In [Lehmann et al. 2010], the structure and connections of already visited articles are visualized as a graph and potentially relevant topics for further exploration are highlighted in a circle around it. A generalization of this idea can be found in [Ahn and Brusilovsky 2013] which adapts the *VIBE* system [Olsen et al. 1993] by adding a user model. The system visualizes the search target objects on a 2D display according to their similarities to a set of points-of-interest (POIs). The POIs represent important keywords or concepts. Users can adapt the view of the data by moving around the POIs. In [Ahn and Brusilovsky 2013] the POIs are generated both from the current query and the user model.

In [Athukorala et al. 2015; Glowacka et al. 2013; Hussain et al. 2010] the problem of updating the user intent model is addressed using reinforcement learning approaches. In particular, in [Glowacka et al. 2013], the user can give direct feedback by manipulating the relevance scores of keywords related to the search results by dragging them closer or further away from the centre of a circle. After the feedback, the model describing the user's search intent is updated and thus also the search results. In [Ruotsalo et al. 2013] the future possible intents, based on alternative feedback that the user might give, are furthermore visualized in an outer circle.

In the approach proposed in this article, the intent model is visualized by displaying the keywords representing the current model to the user. The intent model visualization is used for interacting with the model. A straightforward approach we adopt in this article is to use interactive query expansion, where selecting a keyword increases its relative importance in the intent estimate, thus enabling the user to direct the search.

3. PROACTIVE RETRIEVAL SYSTEM

In this section, we describe our proposed intent modelling method and proactive search system. We estimate a model of the current user intent based on input data from both the primary task context and explicit user feedback. To obtain the contextual predictions, we use a model based on an upper-confidence bound multi-armed bandit algorithm. This algorithm estimates an *intent model*; a weighted term vector representing the user's intent from the interactions in the user's primary task. We also describe how the intent model is used to create a proactive query, which can be used to retrieve documents.

The arms in the multi-armed bandit model are the terms in the document collection and the data is the occurrence of the terms in the documents. The model is rewarded from two input sources: 1) text observed when the user is producing or consuming content (writing or reading), and 2) explicit interactions observed as feedback for the terms predicted in the previous iterations; see Figure 1.

The advantage of a multi-armed bandit model is that it balances exploration with exploitation resulting in the best estimate given the input (exploitation), but at the same time trading-off with the uncertainty of the estimate given the document collection representing domain information (exploration). For example, a user writing "internet of things" should not lead only to documents explicitly about Internet of Things (exploitative estimates), which would likely be the same documents and terms that the user has already seen, but also to documents having less certain estimates (balancing exploitation and exploration), e.g., documents about ubiquitous computing, different sensors, networks, and so on.

In the remainder of the section, we will introduce our intent data model in Section 3.1, describe the used input sources for the intent model in Section 3.2 and the estimation of the intent model in Section 3.3, and describe the generation of the proactive query in Section 3.4.

3.1. Data Model

For computing the intent model, we use a training database consisting of M documents, from which N unique terms are extracted by excluding stop words. The jth document in the database is represented by a feature vector $x_j \in \mathbb{R}^N$ where x_{ij} is the *tf-idf* value of the ith word. We use the following formula for the *tf-idf* value: $x_{ij} = f_{ij} \log\left(\frac{M}{m_i}\right)$, where f_{ij} is the raw frequency of the ith word in the jth document and m_i is the number of documents that contain the ith word.

We denote by $X \in \mathbb{R}^{N \times M}$ the *tf-idf* matrix of the M documents in the data set, where each column of X corresponds to one document feature vector and each row corresponds to a distribution of the terms over the documents.

3.2. Input from Context and User Feedback

Considering writing as the primary task, the input to the intent model comes from two sources: the context, i.e., written input words, and the explicit user feedback (see Figure 1). We denote the relevance vector of observed input words by $y \in [0, \gamma]^N$, $\gamma > 1$, initialized as a zero vector.

The written input words are first converted to the closest corresponding terms of the training set using an approximate string matching algorithm. We then set $y_i = 1$ corresponding to having observed the ith word included among the n most recent written input words. To model the fact that most recently observed words are more relevant than words written some time ago, we apply a time decay on the earlier relevance values as follows: $y_i = s_i^{-1}$, where s_i is the number of words written since the last occurrence of the ith word among the n most recent input words. For computational performance reasons it is useful to truncate very small relevance values, i.e. $y_i < \tau$, to zero. In our experiments the threshold was set to $\tau = 0.1$.

The explicit feedback given for the model is boosted by setting $y_i = \gamma$ for terms that have received direct feedback.

3.3. Capturing Search Intent

The user intent model is estimated based on the input y using the LinRel algorithm [Auer 2003]. The observed values in y are assumed to be formed from the following linear model

$$y = X\hat{w} \ , \tag{1}$$

where $\hat{w} \in \mathbb{R}^M$ consists of weights for the M documents. The magnitude of the jth element of $\hat{w}$ determines the weight of the jth document.

Given y and X, the weights $\hat{w}$ can be obtained solving the Tikhonov regularized regression problem

$$\hat{w} = \underset{w \in \mathbb{R}^M}{\arg\min} \{ \|y - Xw\|^2 + \mu \|w\|^2 \} \ , \tag{2}$$

where $\mu \geq 0$ is a regularization parameter, set to $\mu = 1.0$ in all our experiments. The problem has an explicit solution of the form

$$\hat{w} = (X^T X + \mu I)^{-1} X^T y \ , \tag{3}$$

where I is an identity matrix of size $M \times M$.

Using Eq. (3) the relevance estimate $\hat{y}$ of the keywords can then be computed as

$$\hat{y} = X\hat{w} = X(X^T X + \mu I)^{-1} X^T y = Ay \tag{4}$$

and the upper bound of the standard deviation of $\hat{y}_i$ as

$$\hat{\sigma}_i = \|row_i(A)\|^2. \tag{5}$$

For each word i, the upper confidence bound for the relevance value, i.e., the intent model, is computed as $\hat{y}_i + c\hat{\sigma}_i$, or in vector form:

$$v = \hat{y} + c\hat{\sigma}, \tag{6}$$

where $c \geq 0$ is the exploration/exploitation parameter controlling the trade-off between exploring a wider area of the search space (large c) and focusing on the currently most promising region (small c).

3.4. Proactive Query

The proactive query to retrieve relevant documents is constructed from keywords corresponding to the nonzero values of both the input y (consisting of both the written input words and the direct user feedback) and the intent model v. The input y is used as is, but in order to reduce the number of terms included in the query, only the largest values of v not included in the input y are considered. In other words, we set $v_i = 0$ for both all i with $y_i > 0$ and all i corresponding to keywords not among the N_k largest values of v. Finally, we normalize the weights of suggested keywords by dividing the vector v by the maximum value $v_{\max} = \max_i(v_i)$.

The resulting weight vector $y+v$ is then used to retrieve documents using the Lucene search engine with the standard vector space model as implemented in Lucene.

4. SIMULATION EXPERIMENTS

In order to test whether our method can deduce the task context and provide relevant documents for a variety of domains, we performed a set of simulation experiments. In the simulations, the contextual input consist of the first words of a given document. Interaction with the intent model is also simulated to perform further model validation. These experiments address the research questions R1 and R3.

4.1. Data Sets

The simulations were performed using four data sets having fixed categories. Together the selected data sets cover a wide variety of domains. We picked three publicly available text article databases: 20 Newsgroups, Reuters-21578, and Ohsumed, using their standard training and test set splits over topics. In addition, a subset of the abstracts in the arXiv preprint database was used.

— The *20 Newsgroups*[1] includes 11293 documents in the training set and 7528 documents in the test set. Each document belongs to a given newsgroup, and we use the newsgroup in question as the topic for the document; thus there are 20 different topics.
— In the Reuters-21578 data set, there is a TOPICS attribute, with a total of 135 different values. As some topics and very common and most topics are rare, and there are documents classified to multiple topics, we take the R52 version of the data set[2] and remove the two most common topics (*acq* and *earn*), which together cover over two thirds of the database. Our resulting *Reuters* data set has 2096 training set documents, 789 test set documents, and the number of topics is 50.
— The *Ohsumed*[3] collection includes abstracts from medical journals in 1987–1991, classified into 23 topics. There are 10433 and 12733 abstracts in the training and test sets, respectively.
— The used arXiv[4] subset consists of the Computer Science branch downloaded on October 28, 2015. The branch contains a total of 40 subcategories, which are used as topics in our experiments. A document in the arXiv database can belong to several subcategories, i.e., several topics in our case. 10% of the abstracts in the whole data set were randomly selected into the training set, containing a total of 10310 abstracts. The rest of the data set is used as the test set. We label this data set *arXiv CS*.

[1] http://qwone.com/~jason/20Newsgroups/
[2] http://web.ist.utl.pt/acardoso/datasets/
[3] http://disi.unitn.it/moschitti/corpora.htm
[4] http://arxiv.org/

4.2. Tasks

For each data set, we simulate a setting where a user is writing a text about a given topic. The intent model's training database consists of the training data in the data set. We choose at random a document from the corresponding test set and simulate inputting words from the beginning of the text. We envision a situation where the user has to find some relevant background resources about the given topic. The aim is thus to find other documents about the same topic t as the input document. Because we need the topic information, we can use only data sets with predetermined topics for the simulations; thus we could not use, for instance, Wikipedia.

In the input documents of the 20 Newsgroups data set, all quoted text in the news articles is skipped, as it has been written by earlier participants on the news thread. Instead, the input words are picked from the actual text (subject of the article followed by the article contents) written by the author of that particular news article.

With each new input word, the context is changed and thus the sets of keywords and returned documents are updated. For the proactive queries (Section 3.4), we use the n first words in the input and $N_k = 10$. The Lucene search engine was set to return 10 documents from the test set. The exploitation/exploration parameter in Eq. (6) is set to $c = 1.0$ as recommended in the literature [Athukorala et al. 2015].

4.2.1. Exploratory Search Task. We simulate interaction with the intent model by explicitly selecting keywords as follows. We consider the 20 highest-ranking keywords returned by LinRel (Eq. (6)). For each of the 20 keywords, we count the *tf-idf* values of the keyword k in the set of target documents D_t and divide by the number of documents in D_t. We then allocate to each keyword k a probability of being selected proportional to this value. One keyword is randomly selected using this distribution. The keyword selection process is always repeated ten times in a row, and the weight assigned to selected keywords was chosen to be $\gamma = 2$.

We use all the documents in the database belonging to the same topic as the test document as the target set D_t. As a measure of performance, we use the *precision* with regard to the topic t of the input document.

4.2.2. Known-item Search Task. In the known-item search task, we study a setting where, during writing, the user needs to re-find a certain previously seen document, e.g., in order to verify a previously read piece of information. The setting is the same as for the exploratory search task, except that now we have only one target document in D_t. We take a random document from the test set, and select this as our input document as before. Then we perform a Lucene search over the rest of the test set, using the input document as the query. The highest-scoring retrieved document is now our target document. Note that our target document is thus a different document than our input document, and in the simulation we either find the target document or not.

4.3. Results

Figure 2 and Table I summarize the simulation results with the four data sets (20 Newsgroups, Reuters, Ohsumed, and arXiv CS). The relative improvements obtained with the simulated interaction are also shown in Table I.

4.3.1. Exploratory Search Task. The left column of Figure 2 shows the document retrieval precision in the exploratory search task with the four data sets, as averages over 100 simulation runs. The blue curve shows the retrieval precision over the number of input words n. For $n \in \{10, 20, 30, 40\}$, the green curve shows the effect of ten simulated subsequent keyword selections. Hence, for example for $n = 10$ this means that ten words are first entered one at a time from the input document, simulating the user typing the ten first words. Then, ten keywords suggested by the intent model

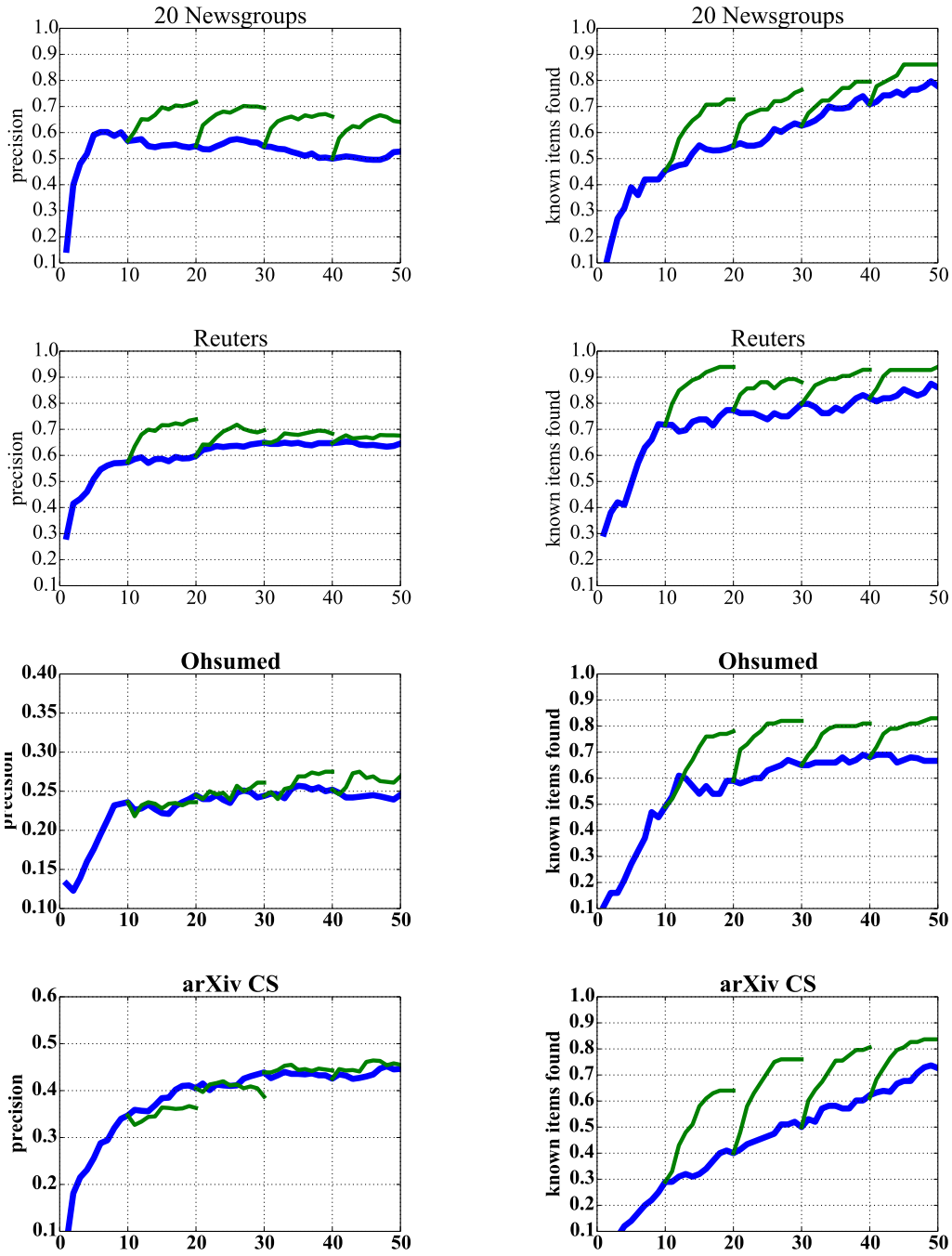

Fig. 2. Exploratory search document precision (left column) and the fraction of known items found (right column) on the 20 Newsgroups (first row), Reuters (second row), Ohsumed (third row), and arXiv CS (last row) data sets. Blue curves correspond to using the current context only; green curves show the simulated interaction experiments with $n \in \{10, 20, 30, 40\}$ and up to ten keywords added. Horizontal axes show the number of words in the current context. All values are averages over 100 simulation runs.

Table I. Exploratory search document precision and the fraction of known items with n words in the current context. The percentages in parentheses show the relative improvement of interaction with the intent model after selecting ten keywords.

Exploratory:

n	20 Newsgroups		Reuters		Ohsumed		arXiv CS	
10	0.57	**(+26%)**	0.57	**(+29%)**	0.24	(+0.0%)	0.35	(+4.6%)
20	0.55	**(+27%)**	0.60	**(+17%)**	0.25	(+6.5%)	0.41	(-4.4%)
30	0.55	**(+21%)**	0.65	(+5.3%)	0.24	(+13%)	0.44	(+0.9%)
40	0.50	**(+28%)**	0.65	(+4.5%)	0.25	(+6.7%)	0.43	(+7.0%)

Known item:

n	20 Newsgroups		Reuters		Ohsumed		arXiv CS	
10	0.45	**(+60%)**	0.72	**(+31%)**	0.49	**(+59%)**	0.29	**(+120%)**
20	0.55	**(+40%)**	0.77	**(+14%)**	0.59	**(+39%)**	0.40	**(+90%)**
30	0.63	**(+27%)**	0.80	**(+16%)**	0.65	**(+25%)**	0.50	**(+61%)**
40	0.71	**(+22%)**	0.82	**(+14%)**	0.68	**(+22%)**	0.61	**(+37%)**

are selected—one at a time with the model being updated in between—simulating the user giving positive feedback to ten keywords in the user interface.

First of all it can be seen that there are notable differences between the precision curves of the corresponding databases. In the case of 20 Newsgroups and Reuters, the retrieval precisions reach a relatively high value (about 0.6) already after 5–10 words of text and remain rather constant after that. On the contrary, the retrieval precision for the Ohsumed and arXiv CS data sets remain at much lower values (around 0.25 and 0.4, respectively) with a slight overall increase in precision as the number of input words increase. An apparent reason for the differences in precision values is the difference in document types: 20 Newsgroups and Reuters consist of news articles, whose typical structure is to have the main content in the beginning, whereas Ohsumed and arXiv CS consist of academic papers, which are longer and more discursive. Another reason for the discrepancy is how contiguous the topics in the given databases are and how much overlapping there exists. For example, if two given topics are semantically very close to each other, there is a high chance that the input test document may be similar to documents of a different topic. In any case, the number of input words has only a limited effect on retrieval precision. For 20 Newsgroups, the decrease in precision with large values of n is likely due to the format of the input documents, as the most discriminative words, especially the article subject, appear in the beginning of the text.

Second, the results show that for some of the data sets it is possible to improve the precision of the retrieved documents by interacting with the intent model. This can be clearly seen in Table I, which shows the relative improvements in precision from adding ten keywords through simulated keyword selections. The relative improvements are about 20–30% for 20 Newsgroups for all tested values of n and for Reuters with small context sizes ($n \in \{10, 20\}$). These improvements are statistically significant, as tested with paired Student t-test, $p < 0.01$. For Ohsumed, arXiv CS, and Reuters with $n \in \{30, 40\}$, the keyword suggestions did not bring statistically significant differences to the precision values. It should be noted, however, that precision does not increase when a new relevant document is found based on the simulated feedback and the new document replaces another earlier found relevant document in the lists of returned documents.

The absolute precision values for 20 Newsgroups and Reuters with the keyword suggestions reach approximately 0.7.

4.3.2. Known-item Search Task. The right column of Figure 2 shows the fraction of known items found against the number of words, as averages over 100 simulation runs, i.e., if the target document is found 13 times over the 100 simulation runs, the fraction is 0.13.

Similarly as before, the blue and green curves illustrate the performance with only the input words and with selected keywords added to the context, respectively.

First, as expected, the probability of finding the specific document increases gradually over the number of written words. There is again variation between the studied data sets, albeit on a lesser scale than in the exploratory setting. For Reuters, the fraction of known items found reaches 0.7 already at $n = 10$ and eventually exceeds 0.8. The other data sets are somewhat more difficult but also eventually reach 0.6–0.8.

Simulated interaction with the intent model can improve the results considerably in the known-item search setting, already after selecting only a few keywords. Table I shows that with short context sizes, i.e., $n = 10$, the relative improvement in the fraction of known items found can be 30–120%. This suggests that the used partly exploratory method for generating keyword suggestions ($c = 1$ was used in Eq. (6)) can produce relevant keywords already with small amounts of context information. On larger context sizes, the blue curves have higher values and the relative improvements are smaller, although still considerable; in particular for Reuters, there is not much room for improvement, and after adding a few keywords, the fraction of known items found reaches 0.9. In all cases, the improvements obtained by adding the keywords are statistically significant (McNemar's test for paired categorical variables, $p < 0.01$, as the result of a single simulation run is binary: found or not found).

4.3.3. Summary of Findings. For research question R1, the results of the simulation show that our method can deduce the task context for many different types of data sets, already when there is only a small number of words observed. Furthermore, the method was shown to work both in exploratory search with a set of relevant target documents and in known-item search.

There are however considerable differences between the used data sets due to the type of documents, e.g., news articles versus scientific abstracts. In exploratory search, the retrieval precision with 20 Newsgroups reaches a peak value with less than 10 words of context whereas with arXiv CS, the precision values continue to increase as the context size increases. In general, using a longer context than 20 words in exploratory search does not bring any substantial improvements. For the known-item search task, as was to be expected due to the nature of the experiment setting, a larger context invariably increases the probability of finding the target document.

To address research question R3, we employed simulated interaction with the intent model. In the experiments, the simulated interaction improved the retrieval results especially in the known-item search task. This also shows that the intent model is able to produce relevant keywords for the current task context. However, due to the inherent characteristics of the data, the relevant improvements arising from choosing suggested keywords differ considerably.

5. USER STUDY

A user study was carried out to study the effectiveness of proactive retrieval and the associated users' information selection behavior in a realistic simulated writing task scenario [Borlund and Schneider 2010].

We used a 2x2 between subjects design with two system variants and two tasks. The system and task ordering were counterbalanced to avoid carryout effects due to practice or fatigue. Thus, there were in total 24 writing tasks: 12 for each topic, and likewise 12 for each system variant.

5.1. Participants

We recruited 12 participants from a university to take part in the study. Two of them were females. Because the text presented in the user interface was in English, only participants with a self-reported good knowledge of English were eligible to take part in the experiment.

Participant were also controlled for their prior knowledge about the topics. Participants self-assessed their knowledge on a scale of 1–5. Participants who answered either 1 or 5 were excluded as they were considered to be either too familiar with a topic and able to write the essay without consulting the retrieved resources, or having too little information about the topic to be able to make meaningful searches. The participants selected to the study reported on average prior knowledge between 1.9 and 2.5 for topics 1 and 2, respectively, and no participants had to be excluded.

None of the participants had prior experience with the system or the data.

Recruitment was by word of mouth and participants received one movie ticket worth approximately 10 USD for their participation in the experiment.

5.2. Tasks

The participants were placed in a simulated work task to write a short essay about a given topic and collect materials and references related to the essay using the provided information retrieval system. As the participants were recruited among university students and personnel, the simulated work task was described as similar to writing an assignment report or a dissertation [Borlund and Schneider 2010]. Each participant in the study was given two writing tasks with fixed topics.

Each essay was to include an introduction to the topic, description of some common methods used in the field, and examples of common applications. Participants were also asked to use the search interface to select 5–10 relevant references to be included in the essay.

The topics of the essays, *"human activity recognition"* (topic 1) and *"neural networks for computer vision"* (topic 2), were selected due to both the presence of relevant material in the used data sets and the availability of suitable experts to provide ground truth annotations.

5.3. Data

Two data sets were used as resources in the user study: the English *Wikipedia* and a snapshot of the full *arXiv CS* database, with a total numbers of documents of about 10 million and 95 000, respectively. The intent model was trained using the same arXiv CS training set as what was used in the simulations.

5.4. User Interface

We implemented a retrieval system and designed a simple search user interface for the purpose of the experiment. In order to capture the primary task context within a writing task, we implemented an extension to the GNU Emacs text editor that automatically tracked written text and transmitted it to the search system after each keypress.

The designed user interface is shown in Figure 3 (right) together with a text editor showing a text written by the user (left). In Figure 3 the resources displayed are *arXiv* preprints and English *Wikipedia* pages. The top-5 resources from both sources are immediately visible in the interface, with further results available using the scroll bars.

Selecting any of the suggested resources by clicking with the left mouse button saves the resource in question to the *"Selected items"* list for further analysis. By clicking

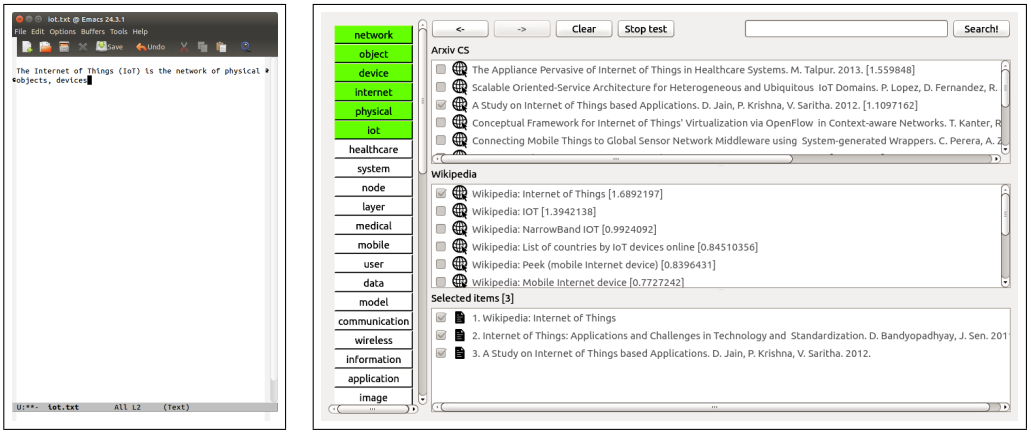

Fig. 3. User interface components used in the user study. The experimental user interface (on the right) with the text *"The Internet of Things (IoT) is the network of physical objects, devices"* as the current context (written by the user using a standard text editor; on the left). The web browser used to display the contents of selected resources is not shown.

on the resources with the right mouse button, the contents of the resource (i.e., the corresponding Wikipedia page or arXiv abstract in the setting of Figure 3) is opened in a regular web browser (not shown in Figure 3).

The list of suggested keywords appear in the user interface on the left hand side and is updated at the same time as the document suggestions. The suggested keywords serve a dual role: to visualize the current estimate of the user's intent and to enable user interaction with the intent model. For to the latter role, the keywords are shown as clickable buttons, with the active (recently written or clicked) keywords colored in green. The user can at any time interact with the intent model by clicking on a suggested keyword. This immediately updates the lists of shown suggestions and keywords.

Furthermore, the user can go back and forward in history using the corresponding buttons ("$\leftarrow$" and "$\rightarrow$"). The history consists of the updates performed by the system based on changes in the context due to either written words or clicked keywords. It is also possible to explicitly clear the context using the *"Clear"* button, which deletes all other context than the selected items.

Finally, the interface contains an explicit search box in the top-right corner. Entering an explicit search clears the current context, as with the *"Clear"* button, and the system returns results based solely on the entered query.

5.5. Baseline

The baseline system was exactly the same as the proactive search system, but it did not use the intent prediction method, and thus the only search option available is through the explicit interaction by typing queries in a search box. Other interface elements were removed from the baseline system.

5.6. Configurations

The search engine was indexed with two data sets (Wikipedia and arXiv CS) and it was set to return a maximum of 10 results in response to a request.

The proactive system had additional parameters and interface elements and it was configured as follows. The proactive user interface was set to display the top 20 keywords, and the weight assigned to the clicked keywords was $\gamma = 2$. As the estimate of

Table II. Behavioral statistics of the user study. The rows show the average values for all experiments and averages conditioned on the system versions (full and baseline). The columns correspond to number of words written; number of explicit queries performed and proactive suggestions offered; the number of documents selected after an explicit or a proactive query; and the number of documents visited with the browser after an explicit or a proactive query. The main statistically significant results are bolded.

	words written	explicit queries	proactive suggestions	selected documents		visited documents	
				explicit	proactive	explicit	proactive
all	194	5.33	27.7	4.25	2.04	6.00	2.17
full	190	4.58	55.4	**2.25**	**4.08**	**3.08**	**4.33**
baseline	198	6.08	-	**6.25**	-	**8.92**	-

the current context in the user study, the $n = 10$ latest written words and $N_k = 10$ suggested keywords were used. The exploration/exploitation coefficient was set to $c = 1$.

5.7. Procedure

Before the actual experiments, the protocol included explaining the functionality of the system, including the existence of two system variants and their differences, to the participants. The participants then practiced using both system variants on their own by writing about a test topic *"Internet of Things"* for a few minutes.

The participants were informed that they had a time limit of 20 minutes. It was also permitted to conclude the ongoing task before the full time had elapsed.

Due to the time limit in the experiments, and to simplify the experiment setting, the participants were asked not to browse further from the web pages opened. After the participants had performed the practice task and they had no further questions about the experiment, the actual experiment was started.

In the proactive search condition, the user interface was shown in a corner of the computer screen (on a desktop monitor), displaying the proactive retrieval results in an unobtrusive manner.

In the experiment, each user was writing one topic using the full system variant and the other topic using the baseline system variant.

After finishing each essay, the participants were asked to fill in a questionnaire and give open ended feedback about the user study and the tested search interfaces.

In total, the experiment took about 70 minutes for each participant, including about 20 minutes for the introduction and training phases.

Participants were told that they could ask the experimenter for clarification at any time during the experiment.

5.8. Apparatus

The user study was performed using two Linux desktop computers equipped with 27" monitors. The screen estate was divided into three interface components: the web browser was opened on the left side of the screen and the right side contained the search interface (top-right corner) and the GNU Emacs text editor (bottom-right corner). The setup was prepared for each user in advance, and GNU Emacs was set to be used in a basic configuration without any of its advanced capabilities. The system was set to update the lists of suggestions whenever a pause of three seconds occurred in the keypresses. Instead of updating the suggestions at regular intervals, we chose to wait until the user pauses the typing before updating the lists to maximize the likelihood that the extracted words are fully-written and to minimize the distraction to the user.

The participants used mouse and standard keyboard to operate the system.

5.9. Performance Measures

As the primary performance measure in the user study, we consider the *relative usefulness of the proactive suggestions* in performing the given writing task, using the

documents selected or visited during the experiments. This addresses the research question R2. We record the number of proactive suggestions and explicit queries performed during the writing of essays and observe whether the document visits (opening the document in the web browser) and document selections to the list of selected items happen as a result of a proactive suggestion or an explicit query. As secondary measures for R2, the quality of the final essays was also evaluated by the experts on the range 1–3 (with 3 as the highest value) and the precisions of the visited and selected documents were recorded.

To address research question R1, we measure the ability of the system versions to produce *relevant* results for the user, by measuring the number of documents returned and the *precision* and *recall* of the aggregated search results during the writing task. That is, we collect and pool the documents returned by the proactive and explicit queries performed, and evaluate the precision and recall of this set of documents.

To obtain the ground truth for evaluating precision and recall, the relevance of the pooled retrieval results of all participants for both essay topics (2435 and 2688 for topics 1 and 2; 5123 in total) were independently and blindly assessed by two expert researchers on the topics. The assessment was done on a binary scale (relevant or not relevant). The disparities in the independent assessments (5.8% in total) were solved with a second round of joint assessment. The number of documents deemed relevant were 258 (11%) and 406 (15%), respectively, for topics 1 and 2.

For research question R3, we include the option to provide feedback by clicking any of the suggested keywords in our user interface and measure the effect of the feedback on the above measures.

5.10. Results

Table II shows the overall behavioral statistics of the user study. The table rows show the average values for all experiments and for the two system versions. On average, the participants wrote essays containing 194 words (minimum 87, maximum 322) in 20 minutes 11 seconds, made 5.3 explicit queries, selected 6.3 documents, and visited 8.2 documents (i.e., opened them on the browser view). In the full system on average 55.4 proactive queries were performed.

There were no substantial correlation between the self-assessed familiarity of the topic and any measured attribute: length of essays, total number of queries, total number or precision of the selected documents, etc. Statistically significant differences were not found on any of the measured attributes between the two topics or the order of the two writing tasks during the study.

In the following discussion, we focus on the initially visible top-5 documents, as the users used the scroll bars to access the rest of the retrieved documents only very rarely (see Section 5.10.3).

5.10.1. Usefulness of Proactive Suggestions (R2). In our experiments (see Table II), there were no statistically significant differences between the two systems in the number of explicit queries performed (full: 4.58, baseline: 6.08, on average), the total number of documents selected, or the total number of documents visited with the browser (Wilcoxon signed-rank test, $p < 0.05$). However, in both the number of documents *selected after an explicit query* (full: 2.25; baseline: 6.25) and the number of documents *visited after an explicit query* (full: 3.08; baseline: 8.92) there are statistically significant differences between the systems. Indeed, with the full system, the majority of both document selections and document visits were resulting from proactive search instances. This shows that our system was able to produce useful documents that the users both noticed and utilized. The lack of statistical significance in the number of explicit queries may be caused by a tendency of the participants to perform explicit

Table III. Task-level retrieval results of the user study. The rows show the average values for all experiments and averages conditioned on the system versions (full and baseline). The columns correspond to the total number, the number of relevant, precision, and recall of documents returned in the top-5 result lists. The main statistically significant results are bolded.

	all	relevant	precision	recall
all	152	36.2	39.0%	11.1%
full	**270**	**54.1**	**20.6%**	**16.9%**
baseline	**35.0**	**18.3**	**57.4%**	**5.38%**

queries due to familiarity with them and just to make sure they have covered everything. As users get more comfortable with proactive tools, this behavior might change.

The precisions of the selected and the visited documents were on average 0.83 and 0.81, respectively, with no statistically significant differences between the systems. In the full system, there was also no significant difference in the precision values of documents resulting from an explicit query or a proactive suggestion; this holds both for documents visited and documents selected.

Of the top-5 retrieved documents, 55% originated from arXiv CS and 45% from Wikipedia. The distributions for the selected and visited documents were also similar: 55% of all selected and 53% of all visited documents were from arXiv CS.

The used system variant did not have any significant effect on the quality of the essays. The averages of the quality assessments for the full system and the baseline were 1.6 and 1.7, respectively.

After completing both writing tasks, the users were asked an additional question "The proactive search results were useful and I was able to complete the assigned task mostly without explicit search queries when using the proactive system setup". On a Likert scale of 1–5 (with 1 meaning "completely disagree", and 5 "completely agree"), the average was 3.42.

5.10.2. Relevance of Retrieved Documents (R1). Table III shows the task-level retrieval results of the user study. As expected, the full system produces more distinct documents in the top-5 slots of the result lists (270 versus 35.0 on average). The difference between the systems is also significant for the numbers of relevant documents (54.1 versus 18.3 on average). The corresponding values for recall are 16.9% for the full and 5.38% for baseline system. The baseline scores a higher value on retrieval precision (full: 20.6%; baseline: 57.4%). This is intuitive as the proactive retrieval was set to constantly produce results (whenever a three-second pause in the typing occurred), that is, even before the input was converged to any specific topic. Therefore, the retrieval results are essentially random in the beginning, which drastically lowers the precision value. Second, the proactive retrieval results are in general of a more explorative nature. The attributes were tested with a paired Student t-test, $p < 0.01$.

Overall, the results indicate that the proactive interface can support the user with a wider variety of relevant documents.

5.10.3. Auxiliary User Interaction (R3). Interaction with the suggested keywords in the full system (see Section 5.4) was only scarcely performed by the participants. The average number of clicks per writing task was 1.0. In the open ended feedback, several users commented that they felt the keyword suggestions were not necessary to complete the writing task, and in some cases even that they forgot about the possibility to use keywords. The control buttons (*"←"*, *"→"*, *"Clear"*) were also relatively seldom used. The average number of button presses per writing task was 2.8, of which 1.2 were uses of the *"Clear"* button. Furthermore, the scroll bars were used only 0.25 times per

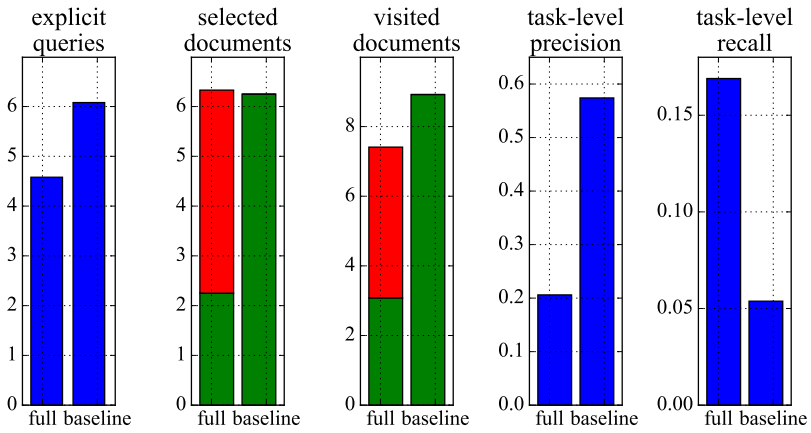

Fig. 4. Overview of the results of the user study. The green and red bars correspond to documents accessed after an explicit or a proactive query, respectively. The three leftmost plots address the research question R2, and the two plots on the right the research question R1.

writing task. These results would support the design principle that the needed interaction with a proactive retrieval system is minimized and the suggested resources are readily available [Rhodes 2000].

5.10.4. Summary of Findings. An overview of the results of the user study is shown in Figure 4. The first three plots on the left address the research question R2, and the remaining two plots the research question R1. Regarding R2, the results show that proactive search is able to provide plenty of useful resources for the writing task, as the majority of active operations on the documents (i.e., selections and visits) when using the full interface are resulting from proactive searches (shown as red bars in Figure 4). Furthermore, as there were no significant differences in the task output (the quality of the written essays and the relevance of the selected documents), this would suggest that the proposed system can reduce the required manual search effort for the kind of writing task studied in the experiments without affecting the quality of the task output. Regarding R1, the full system provided the user access to considerably more (i.e., threefold) relevant documents to support the completion of the task at hand. The overall retrieval precision with the full system is distinctly lower due to the explorative nature of proactive search and, most notably, the design choice of producing immediate retrieval results in the full system. This inevitably leads to high variance, low precision results especially in the beginning of a task before the input converges to a certain topic. For R3, the user study was inconclusive as the participants did not opt for providing explicit user feedback to a sufficient degree.

6. DISCUSSION

In this article, we propose a method for capturing the user intent using the contextual information available from a primary task. We use writing as the primary task in the experiments, but the proposed intent model is not limited to supporting writing. Rather, the context can be observed from various implicit task contexts. In order to be able to support the user in the current task, the method requires relevant documents to be available in the data set. The simulation experiments show that the proactive retrieval method and the proposed interaction with the intent model are particularly suited for re-finding a certain previously seen document. This would suggest that the

proposed method would be effective when combined with a personal data storage system collecting the user's digital footprint to a user-controlled data repository [Sjöberg et al. 2017]. Still, the topic does not need to be fully fixed in advance, as the system can also support more exploratory settings.

There were considerable differences between the used data sets in the simulations due to the type of documents, e.g., news articles versus scientific abstracts. With news articles, the retrieval precision may reach its maximum value with less than 10 words of context observed, as the title and the beginning of the article typically contain the main content of the article. On the other hand, scientific abstracts are typically more discursive and the precision values tend to continue to increase as the context size increases.

Another aspect influencing the effectiveness of proactive information retrieval is the difficulty of the primary task. In the current user study, we selected participants that self-assessed their familiarity with the topics between 2–4 on a scale of 1–5. That is, we excluded participants too familiar with the topic as they might be able to write the essay without consulting the retrieved resources, and participants having too little prior information to be able to perform the task in the given time limit. The effect of the task difficulty is, however, an interesting question that should be studied further.

In the user study, the comparison was designed to be against the current universal method of information gathering during a writing task, i.e. making explicit queries using a traditional search interface. The list of suggested keywords in the user interface was designed to implement a dual role, to visualize the current estimate of the user intent and to enable user interaction with the intent model. The keywords are volatile in an unconverged state and get more focused as the intent model converges to a certain topic. It should be emphasized that the user interface is a research prototype that was designed for the purpose of the user study. In particular, the participants utilized only infrequently the explicit user feedback mechanism offered in the interface. Similar user behavior has been observed in earlier studies [White and Marchionini 2007], and it can be due to the perceived extra effort required to judge the suggested keywords, the relative ease of finding suitable documents to support the writing task (as was commented by some participants), the keywords being viewed too obvious or too similar to each other, the unfamiliarity of the participants with the interface, or unidentified user interface issues.

To evaluate precision and recall in the user study, we use pooled retrieval results which enable us to record the overall recall during the whole writing task, which is an essential measure for the research question R1. The retrieval system was set to constantly produce results, i.e. also before converging to a topic. In the beginning, the results are thus essentially random, substantially the pooled precision value. Therefore, we consider precision to be a rather uninformative measure in this setting. However, we do consider the retrieved results an important cue for the user about the state of the intent model also in the unconverged state.

6.1. Research Questions

In this section, we collect our findings on the specified research questions.

R1: How can the primary task context be used to model the user's intent sufficiently accurately to retrieve resources relevant to the primary task? The results of the experiments show that our proactive method can successfully model the task context and provide relevant documents with a small amount of primary task input (about ten words in the simulation experiments) collected from the context. The provided documents are also more diverse than what is retrieved based on standard explicit queries, offering the user a broader selection of relevant resources to support the completion of the task at hand.

R2: Can the proactive information retrieval tool produce useful resources in a way that better supports the user in performing the primary task? The results show that the proposed method is able to provide plenty of useful resources for the studied writing task. This can be concluded from the user study where it was observed that the majority of documents the users selected or visited were retrieved proactively. This can be considered as reduced manual search effort in the primary task. However, the decrease in the number of explicit queries with the proposed method was not statistically significant in the user study. This result calls for further investigation. The quality of task output (essay quality and relevance of the selected documents) was not affected in either direction by introducing the proposed proactive system. Overall, this suggests that the user was able to perform the task with less effort and equal results as compared to the baseline.

R3: Can the retrieval results be improved by user interaction with the proactive retrieval results? It was observed that the retrieval results can be improved using simulated user interaction, showing that the intent model is able to suggest both relevant and useful keywords for the current task context. The improvement was most notable when re-finding a certain previously seen document, i.e., in the simulated known-item search task. The user study was inconclusive for this research question as the participants did not make enough use of the explicit user feedback mechanism (clicking on the suggested keywords) in the experiments.

6.2. Future Work

The variability of the intent model estimate is controlled by a specific exploration/exploitation parameter c. When the parameter is set to pure exploitation mode, the suggested resources and keywords correspond directly to the current context data. When in exploration mode, more diverse documents and keywords are suggested. In the experiments reported in this article, we used the value $c = 1$ recommended in the literature; this value worked well, especially in the known-item search task. A further study on the effect of varying c is a natural continuation of this work.

Initial additional experiments indicate that the *tf-idf* formula used directly influences the keywords provided. We will study further how different parameter choices correspond to different settings (exploratory search, known item search) and different types of data.

Another possible future development would be to use key phrases instead of keywords for interactive query expansion, as key phrases can convey more elaborate meanings than single words and can thus serve as a more effective source of feedback. Furthermore, corpus-based query suggestion [Bhatia et al. 2011] could be applied to better cover the various possible choices to improve the estimate of the user's intent. Another possible direction would be to investigate better methods to visualize the entire proactive query. In the proposed system, showing the actual query issued might be confusing to the user as it is a collection of weighted keywords and not e.g. a natural language sentence. Here our goal was to have an unobtrusive interface that would not require the user to have a deep understanding of the underlying mechanisms. Still, it would be an interesting topic for further work to be able to display the entire query in a more understandable way.

Further development of our system also calls for a larger user study. Interesting resources that can be included for retrieval are the emails of the user and document files saved on the user's computer. The difficulty of the primary task is also a pertinent issue whose effect needs to be addressed in more detail.

Finally, in contrast to many of the existing methods for producing proactive recommendations, the method proposed in this article generalizes the context gathering in the sense that the context data can be extracted from several sources, such as a word

processing software, PDF reader, or web browser. The only requirement for the context data is that it has to be in textual form.

7. CONCLUSIONS

We have described a method for capturing the user intent using the information available from a primary task and a system implementation to proactively provide sources of information to a user. Simulation experiments and a user study provide evidence that our system is able to proactively produce relevant and useful resources.

The user study demonstrated that the information that the users found useful, i.e., the documents that they selected or visited, could be accessed with fewer explicit queries (Figure 4). This indicates that the proposed method can reduce users' manual search effort by proactively offering useful information. The total number of explicit queries performed decreased when the proactive search was being utilized. The decrease was not however significant. As the explicit queries with the proposed system mostly did not result in additional useful resources for the given primary task, our hypothesis is that they were mostly unnecessary queries due to users' familiarity with manual searches, and unwillingness to fully trust the proactive system to cover everything. Perhaps this will change as users become more comfortable with proactive search. In addition, a higher task-level recall can be achieved with the proactive functionality. Using the conventional search interface, the participants were on average exposed to only about 5% of the relevant documents. With the proposed method, the recall increased threefold.

In the proposed system, the user could inspect and interact with the intent model by clicking on the estimated relevant keywords. In the simulated experiments, keyword feedback was found to be useful, in particular in the known-item search task. In the exploratory search task, keyword feedback was useful only with half of the data sets. The differences in performance between the known-item and exploratory search tasks can partly be explained by the evaluation setup, as in the known-item search task, a clear target document was required to be found and the feedback can direct the search towards the target, whereas the precision in exploratory search might not increase, even when new relevant documents are found based on the feedback, if they happen to replace other relevant documents found earlier.

In the user study, the suggested keywords were only occasionally used by the participants. The reluctance of users to provide feedback is a well-known issue, which can be caused by several issues. A larger sample size could reveal differences in user behavior and provide more insights about the usefulness of the feedback suggestions. It can be said that proactive retrieval turns the traditional information retrieval paradigm on its head: the user engages with the search after the retrieval to improve the results, instead of performing a search query to begin the retrieval session.

REFERENCES

Gediminas Adomavicius and Alexander Tuzhilin. 2008. Context-aware recommender systems. In *Proceedings of the 2008 ACM Conference on Recommender Systems (RecSys '08)*. ACM, New York, NY, USA, 335–336. DOI:http://dx.doi.org/10.1145/1454008.1454068

Eugene Agichtein, Eric Brill, and Susan Dumais. 2006. Improving web search ranking by incorporating user behavior information. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '06)*. ACM, New York, NY, USA, 19–26. DOI:http://dx.doi.org/10.1145/1148170.1148177

Jae-wook Ahn and Peter Brusilovsky. 2013. Adaptive visualization for exploratory information retrieval. *Information Processing & Management* 49, 5 (2013), 1139 – 1164. DOI:http://dx.doi.org/10.1016/j.ipm.2013.01.007

Jae-wook Ahn, Peter Brusilovsky, Jonathan Grady, Daqing He, and Sue Yeon Syn. 2007. Open user profiles for adaptive news systems: Help or harm?. In *Proceedings of the 16th International Conference on World Wide Web (WWW '07)*. ACM, New York, NY, USA, 11–20. DOI: http://dx.doi.org/10.1145/1242572.1242575

Jae-wook Ahn, Peter Brusilovsky, Daqing He, Jonathan Grady, and Qi Li. 2008. Personalized web exploration with task models. In *Proceedings of the 17th International Conference on World Wide Web (WWW '08)*. ACM, New York, NY, USA, 1–10. DOI: http://dx.doi.org/10.1145/1367497.1367499

Kumaripaba Athukorala, Alan Medlar, Kalle Ilves, and Dorota Glowacka. 2015. Balancing exploration and exploitation: Empirical parameterization of exploratory search systems. In *Proc. of the 24th ACM International on Conference on Information and Knowledge Management (CIKM '15)*. ACM, New York, NY, USA, 1703–1706. DOI: http://dx.doi.org/10.1145/2806416.2806609

Peter Auer. 2003. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research* 3 (March 2003), 397–422. http://dl.acm.org/citation.cfm?id=944919.944941

Tamara Babaian, Barbara J. Grosz, and Stuart M. Shieber. 2002. A writer's collaborative assistant. In *Proceedings of the 7th International Conference on Intelligent User Interfaces (IUI '02)*. ACM, New York, NY, USA, 7–14. DOI: http://dx.doi.org/10.1145/502716.502722

Fedor Bakalov, Birgitta König-Ries, Andreas Nauerz, and Martin Welsch. 2010. Introspectiveviews: An interface for scrutinizing semantic user models. In *User Modeling, Adaptation, and Personalization: 18th International Conference, UMAP 2010, Big Island, HI, USA, June 20-24, 2010. Proceedings*, Paul De Bra, Alfred Kobsa, and David Chin (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 219–230. DOI: http://dx.doi.org/10.1007/978-3-642-13470-8_21

Michelle Q. Wang Baldonado and Terry Winograd. 1997. Sensemaker: An information-exploration interface supporting the contextual evolution of a user's interests. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems (CHI '97)*. ACM, New York, NY, USA, 11–18. DOI: http://dx.doi.org/10.1145/258549.258563

Nicholas J. Belkin and W. Bruce Croft. 1992. Information filtering and information retrieval: Two sides of the same coin? *Commun. ACM* 35, 12 (Dec. 1992), 29–38. DOI: http://dx.doi.org/10.1145/138859.138861

Nicholas. J. Belkin, Robert N. Oddy, and H. M. Brooks. 1982. Ask for information retrieval: Part i.: Background and theory. *Journal of Documentation* 38, 2 (1982), 61–71.

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 (SIGIR '12)*. ACM, New York, NY, USA, 185–194. DOI: http://dx.doi.org/10.1145/2348283.2348312

Sumit Bhatia, Debapriyo Majumdar, and Prasenjit Mitra. 2011. Query suggestions in the absence of query logs. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. ACM, 795–804.

Ilaria Bordino, Gianmarco De Francisci Morales, Ingmar Weber, and Francesco Bonchi. 2013. From Machu_Picchu to "rafting the urubamba river": Anticipating information needs via the entity-query graph. In *Proceedings of the sixth ACM international conference on Web search and data mining*. ACM, 275–284.

Pia Borlund and Jesper W. Schneider. 2010. Reconsideration of the simulated work task situation: A context instrument for evaluation of information retrieval interaction. In *Proceedings of the Third Symposium on Information Interaction in Context (IIiX '10)*. ACM, New York, NY, USA, 155–164. DOI: http://dx.doi.org/10.1145/1840784.1840808

Svetlin Bostandjiev, John O'Donovan, and Tobias Höllerer. 2012. Tasteweights: A visual interactive hybrid recommender system. In *Proceedings of the Sixth ACM Conference on Recommender Systems (RecSys '12)*. ACM, New York, NY, USA, 35–42. DOI: http://dx.doi.org/10.1145/2365952.2365964

Jay Budzik, Kristian J. Hammond, and Larry Birnbaum. 2001. Information access in context. *Knowledge-Based Systems* 14, 1–2 (2001), 37–53. DOI: http://dx.doi.org/10.1016/S0950-7051(00)00105-2

Katriina Byström and Preben Hansen. 2005. Conceptual framework for tasks in information studies. *Journal of the American Society for Information Science and Technology* 56, 10 (2005), 1050–1061. DOI: http://dx.doi.org/10.1002/asi.20197

Huanhuan Cao, Derek Hao Hu, Dou Shen, Daxin Jiang, Jian-Tao Sun, Enhong Chen, and Qiang Yang. 2009. Context-aware query classification. In *Proceedings of the 32Nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '09)*. ACM, New York, NY, USA, 3–10. DOI: http://dx.doi.org/10.1145/1571941.1571945

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 (KDD '08)*. ACM, New York, NY, USA, 875–883. DOI: http://dx.doi.org/10.1145/1401890.1401995

Zhicong Cheng, Bin Gao, and Tie-Yan Liu. 2010. Actively predicting diverse search intent from user browsing behaviors. In *Proceedings of the 19th international conference on World wide web*. ACM, 221–230.

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 (SIGIR '07)*. ACM, New York, NY, USA, 7–14. DOI:http://dx.doi.org/10.1145/1277741.1277746

Paul Alexandru Chirita, Wolfgang Nejdl, Raluca Paiu, and Christian Kohlschütter. 2005. Using odp metadata to personalize search. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '05)*. ACM, New York, NY, USA, 178–185. DOI:http://dx.doi.org/10.1145/1076034.1076067

Jia Tina Du and Amanda Spink. 2011. Toward a web search model: Integrating multitasking, cognitive coordination, and cognitive shifts. *Journal of the American Society for Information Science and Technology* 62, 8 (2011), 1446–1472. DOI:http://dx.doi.org/10.1002/asi.21551

Susan Dumais, Edward Cutrell, Raman Sarin, and Eric Horvitz. 2004a. Implicit queries (iq) for contextualized search. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '04)*. ACM, New York, NY, USA, 594–594. DOI:http://dx.doi.org/10.1145/1008992.1009137

Susan Dumais, Edward Cutrell, Raman Sarin, and Eric Horvitz. 2004b. Implicit queries (IQ) for contextualized search. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 594–594.

Desmond Elliott and Joemon M. Jose. 2009. A proactive personalised retrieval system. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management (CIKM '09)*. ACM, New York, NY, USA, 1935–1938. DOI:http://dx.doi.org/10.1145/1645953.1646269

Xin Fu. 2010. Towards a model of implicit feedback for web search. *Journal of the American Society for Information Science and Technology* 61, 1 (2010), 30–49. DOI:http://dx.doi.org/10.1002/asi.21198

Dorota Glowacka, Tuukka Ruotsalo, Ksenia Konuyshkova, Samuel Kaski, Giulio Jacucci, and others. 2013. Directing exploratory search: Reinforcement learning from user interactions with keywords. In *Proceedings of the 2013 international conference on Intelligent user interfaces*. ACM, 117–128.

Ramanathan Guha, Vineet Gupta, Vivek Raghunathan, and Ramakrishnan Srikant. 2015. User modeling for a personal assistant. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining (WSDM '15)*. ACM, New York, NY, USA, 275–284. DOI:http://dx.doi.org/10.1145/2684822.2685309

Negar Hariri, Bamshad Mobasher, and Robin Burke. 2014. Context adaptation in interactive recommender systems. In *Proceedings of the 8th ACM Conference on Recommender Systems (RecSys '14)*. ACM, New York, NY, USA, 41–48. DOI:http://dx.doi.org/10.1145/2645710.2645753

John R Hayes and LS Flower. 1980. Identifying the organization of writing processes. i lw gregg & er steinberg (eds.), cognitive processes in writing (pp. 3-30). (1980).

Chen He, Denis Parra, and Katrien Verbert. 2016. Interactive recommender systems: A survey of the state of the art and future research challenges and opportunities. *Expert Systems with Applications* 56, Supplement C (2016), 9 – 27. DOI:http://dx.doi.org/https://doi.org/10.1016/j.eswa.2016.02.013

Monika Henzinger, Bay-Wei Chang, Brian Milch, and Sergey Brin. 2005. Query-free news search. *World Wide Web* 8, 2 (2005), 101–126. DOI:http://dx.doi.org/10.1007/s11280-004-4870-6

Zakria Hussain, Alex P Leung, Kitsuchart Pasupa, David R Hardoon, Peter Auer, and John Shawe-Taylor. 2010. Exploration-exploitation of eye movement enriched multiple feature spaces for content-based image retrieval. In *Machine Learning and Knowledge Discovery in Databases*. Springer, 554–569.

Peter Ingwersen and Kalervo Järvelin. 2005. *The Turn: Integration of Information Seeking and Retrieval in Context (The Information Retrieval Series)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

Thorsten Joachims. 2002. Optimizing search engines using clickthrough data. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '02)*. ACM, New York, NY, USA, 133–142. DOI:http://dx.doi.org/10.1145/775047.775067

Rosie Jones and Kristina Lisa Klinkner. 2008. Beyond the session timeout: Automatic hierarchical segmentation of search topics in query logs. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM '08)*. ACM, New York, NY, USA, 699–708. DOI:http://dx.doi.org/10.1145/1458082.1458176

Evangelos Kanoulas, Ben Carterette, Paul D. Clough, and Mark Sanderson. 2011. Evaluating multi-query sessions. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '11)*. ACM, New York, NY, USA, 1053–1062. DOI:http://dx.doi.org/10.1145/2009916.2010056

Weize Kong, Rui Li, Jie Luo, Aston Zhang, Yi Chang, and James Allan. 2015. Predicting search intent based on pre-search context. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 503–512.

Simon Lehmann, Ulrich Schwanecke, and Ralf Drner. 2010. Interactive visualization for opportunistic exploration of large document collections. *Information Systems* 35, 2 (2010), 260 – 269. DOI:http://dx.doi.org/10.1016/j.is.2009.10.004 Special Section: Context-Oriented Information Integration.

Yuelin Li and Nicholas J. Belkin. 2008. A faceted approach to conceptualizing tasks in information seeking. *Information Processing & Management* 44, 6 (2008), 1822 – 1837. DOI:http://dx.doi.org/https://doi.org/10.1016/j.ipm.2008.07.005 Adaptive Information Retrieval.

Zhen Liao, Yang Song, Li-wei He, and Yalou Huang. 2012. Evaluating the effectiveness of search task trails. In *Proceedings of the 21st International Conference on World Wide Web (WWW '12)*. ACM, New York, NY, USA, 489–498. DOI:http://dx.doi.org/10.1145/2187836.2187903

Henry Lieberman. 1995. Letizia: An agent that assists web browsing. In *Proc. IJCAI (1)*. 924–929.

Daniel J. Liebling, Paul N. Bennett, and Ryen W. White. 2012. Anticipatory search: Using context to initiate search. In *Proc. SIGIR*. ACM, 1035–1036.

Jingjing Liu and Nicholas J. Belkin. 2010. Personalizing information retrieval for multi-session tasks: The roles of task stage and task type. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '10)*. ACM, New York, NY, USA, 26–33. DOI:http://dx.doi.org/10.1145/1835449.1835457

Yefeng Liu, Darren Edge, and Koji Yatani. 2013. Sidepoint: a peripheral knowledge panel for presentation slide authoring. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 681–684.

Avishay Livne, Vivek Gokuladas, Jaime Teevan, Susan T Dumais, and Eytan Adar. 2014. Citesight: supporting contextual citation recommendation using differential search. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*. ACM, 807–816.

Petri Luukkonen, Markus Koskela, and Patrik Floréen. 2016. LSTM-based predictions for proactive information retrieval. In *Proceedings of Neu-IR: The SIGIR 2016 Workshop on Neural Information Retrieval*. http://arxiv.org/abs/1606.06137

Zhongming Ma, Gautam Pant, and Olivia R. Liu Sheng. 2007. Interest-based personalized search. *ACM Trans. Inf. Syst.* 25, 1, Article 5 (Feb. 2007). DOI:http://dx.doi.org/10.1145/1198296.1198301

Tariq Mahmood and Francesco Ricci. 2007. Learning and adaptivity in interactive recommender systems. In *Proceedings of the Ninth International Conference on Electronic Commerce (ICEC '07)*. ACM, New York, NY, USA, 75–84. DOI:http://dx.doi.org/10.1145/1282100.1282114

Mari Carmen Puerta Melguizo, Lou Boves, and Olga Muoz Ramos. 2009. A proactive recommendation system for writing: Helping without disrupting. *International Journal of Industrial Ergonomics* 39, 3 (2009), 516–523. DOI:http://dx.doi.org/10.1016/j.ergon.2008.10.004

Massimo Melucci. 2012. *Contextual Search*. Now Publishers Inc., Hanover, MA, USA.

Kai A. Olsen, Robert R. Korfhage, Kenneth M. Sochats, Michael B. Spring, and James G. Williams. 1993. Visualization of a document collection: The vibe system. *Information Processing & Management* 29, 1 (1993), 69 – 81. DOI:http://dx.doi.org/10.1016/0306-4573(93)90024-8

Valeria Orso, Tuukka Ruotsalo, Jukka Leino, Luciano Gamberini, and Giulio Jacucci. 2017. Overlaying social information: The effects on users search and information-selection behavior. *Information Processing & Management* 53, 6 (2017), 1269 – 1286. DOI:http://dx.doi.org/https://doi.org/10.1016/j.ipm.2017.06.001

Karthik Raman, Paul N. Bennett, and Kevyn Collins-Thompson. 2013. Toward whole-session relevance: Exploring intrinsic diversity in web search. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '13)*. ACM, New York, NY, USA, 463–472. DOI:http://dx.doi.org/10.1145/2484028.2484089

Karthik Raman, Paul N. Bennett, and Kevyn Collins-Thompson. 2014. Understanding intrinsic diversity in web search: Improving whole-session relevance. *ACM Trans. Inf. Syst.* 32, 4, Article 20 (Oct. 2014), 45 pages. DOI:http://dx.doi.org/10.1145/2629553

Paul Resnick and Hal R. Varian. 1997. Recommender systems. *Commun. ACM* 40, 3 (March 1997), 56–58. DOI:http://dx.doi.org/10.1145/245108.245121

Bradley Rhodes and Thad Starner. 1996. Remembrance Agent: A continuously running automated information retrieval system. In *The Proceedings of The First International Conference on The Practical Application Of Intelligent Agents and Multi Agent Technology*. 487–495.

Bradley J. Rhodes. 2000. Margin Notes: Building a contextually aware associative memory. In *Proceedings of the 5th International Conference on Intelligent User Interfaces (IUI '00)*. ACM, New York, NY, USA, 219–224. DOI:http://dx.doi.org/10.1145/325737.325850

Tuukka Ruotsalo, Giulio Jacucci, Petri Myllymäki, and Samuel Kaski. 2014. Interactive intent modeling: Information discovery beyond search. *Commun. ACM* 58, 1 (Dec. 2014), 86–92. DOI:http://dx.doi.org/10.1145/2656334

Tuukka Ruotsalo, Jaakko Peltonen, Manuel Eugster, Dorota Glowacka, Ksenia Konyushkova, Kumaripaba Athukorala, Ilkka Kosunen, Aki Reijonen, Petri Myllymäki, Giulio Jacucci, and Samuel Kaski. 2013. Directing exploratory search with interactive intent modeling. In *Proceedings of the 22nd ACM international conference on Conference on information and knowledge management (CIKM '13)*. ACM, New York, NY, USA, 1759–1764. DOI:http://dx.doi.org/10.1145/2505515.2505644

Miamaria Saastamoinen and Kalervo Järvelin. 2016. Search task features in work tasks of varying types and complexity. *Journal of the Association for Information Science and Technology* (2016). DOI:http://dx.doi.org/10.1002/asi.23766

Mats Sjöberg, Hung-Han Chen, Patrik Floréen, Markus Koskela, Kai Kuikkaniemi, Tuukka Lehtiniemi, and Jaakko Peltonen. 2017. *Digital Me: Controlling and Making Sense of My Digital Footprint*. Springer International Publishing, Cham, 155–167.

Yang Song and Qi Guo. 2016. Query-less: Predicting task repetition for nextgen proactive search and recommendation engines. In *Proceedings of the 25th International Conference on World Wide Web (WWW '16)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland, 543–553. DOI:http://dx.doi.org/10.1145/2872427.2883020

David Sontag, Kevyn Collins-Thompson, Paul N. Bennett, Ryen W. White, Susan Dumais, and Bodo Billerbeck. 2012. Probabilistic models for personalizing web search. In *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining (WSDM '12)*. ACM, New York, NY, USA, 433–442. DOI:http://dx.doi.org/10.1145/2124295.2124348

Micro Speretta and Susan Gauch. 2005. Personalized search based on user search histories. In *Proceedings of the 2005 IEEE/WIC/ACM International Conference on Web Intelligence (WI '05)*. IEEE Computer Society, Washington, DC, USA, 622–628. DOI:http://dx.doi.org/10.1109/WI.2005.114

Jaime Teevan, Susan T. Dumais, and Eric Horvitz. 2010. Potential for personalization. *ACM Trans. Comput.-Hum. Interact.* 17, 1, Article 4 (April 2010), 31 pages. DOI:http://dx.doi.org/10.1145/1721831.1721835

Michael B. Twidale, Anatoliy A. Gruzd, and David M. Nichols. 2008. Writing in the library: Exploring tighter integration of digital library use with the writing process. *Information Processing & Management* 44, 2 (2008), 558–580. DOI:http://dx.doi.org/10.1016/j.ipm.2007.05.010

Pertti Vakkari. 2001. A theory of the task-based information retrieval process: a summary and generalization of a longitudinal study. *Journal of Documentation* 57, 1 (2001), 44–60.

Pertti Vakkari. 2003. Task-based information searching. *Annual Review of Information Science and Technology* 37, 1 (2003), 413–464. DOI:http://dx.doi.org/10.1002/aris.1440370110

Tung Vuong, Giulio Jacucci, and Tuukka Ruotsalo. 2017a. Proactive information retrieval via screen surveillance. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '17)*. ACM, New York, NY, USA, 1313–1316. DOI:http://dx.doi.org/10.1145/3077136.3084151

Tung Vuong, Giulio Jacucci, and Tuukka Ruotsalo. 2017b. 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. DOI:http://dx.doi.org/10.1145/3130974

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 (SIGIR '09)*. ACM, New York, NY, USA, 363–370. DOI:http://dx.doi.org/10.1145/1571941.1572005

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 (CIKM '10)*. ACM, New York, NY, USA, 1009–1018. DOI:http://dx.doi.org/10.1145/1871437.1871565

Ryen W. White, Wei Chu, Ahmed Hassan, Xiaodong He, Yang Song, and Hongning Wang. 2013. Enhancing personalized search by mining and modeling task behavior. In *Proceedings of the 22Nd International Conference on World Wide Web (WWW '13)*. ACM, New York, NY, USA, 1411–1420. DOI:http://dx.doi.org/10.1145/2488388.2488511

Ryen W White and Gary Marchionini. 2007. Examining the effectiveness of real-time query expansion. *Information Processing & Management* 43, 3 (2007), 685–704.

Biao Xiang, Daxin Jiang, Jian Pei, Xiaohui Sun, Enhong Chen, and Hang Li. 2010. Context-aware ranking in web search. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*. ACM, 451–458.

Hong (Iris) Xie. 2000. Shifts of interactive intentions and information-seeking strategies in interactive information retrieval. *Journal of the American Society for Information Science* 51, 9 (2000), 841–857. DOI: http://dx.doi.org/10.1002/(SICI)1097-4571(2000)51:9⟨841::AID-ASI70⟩3.0.CO;2-0

Received July 2017; revised ?; accepted ?