Information Processing and Management 56 (2019) 394–407

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/infoproman

Characterizing and predicting downloads in academic search

Xinyi Li*, Maarten de Rijke

Informatics Institute, University of Amsterdam, Science Park 904, Amsterdam XH 1098, The Netherlands

ARTICLE INFO

Keywords:
Academic search
Download behavior
Download prediction
User segmentation

ABSTRACT

Numerous studies have been conducted on the information interaction behavior of search engine users. Few studies have considered information interactions in the domain of academic search. We focus on conversion behavior in this domain. Conversions have been widely studied in the e-commerce domain, e.g., for online shopping and hotel booking, but little is known about conversions in academic search. We start with a description of a unique dataset of a particular type of conversion in academic search, viz. users' downloads of scientific papers. Then we move to an observational analysis of users' download actions. We first characterize user actions and show their statistics in sessions. Then we focus on behavioral and topical aspects of downloads, revealing behavioral correlations across download sessions. We discover unique properties that differ from other conversion settings such as online shopping. Using insights gained from these observations, we consider the task of predicting the next download. In particular, we focus on predicting the time until the next download session, and on predicting the number of downloads. We cast these as time series prediction problems and model them using LSTMs. We develop a specialized model built on user segmentations that achieves significant improvements over the state-of-the art.

1. Introduction

Conversions are critical to websites as they are directly related to revenue, and can indicate the performance of the platform. Research into conversions has received considerable attention in areas such as online shopping and hotel booking (see, e.g., Ghose, Ipeirotis, & Li, 2014; Kooti et al., 2016; Lee, Ha, Han, Rha, & Kwon, 2015), where a conversion is a purchase or booking action. These studies reveal the purchase and booking behavior characteristics of users and provide insights on predicting conversions in those domains.

Academic search concerns the retrieval of information objects in the domain of academic research (papers, journals, authors, etc). Research on conversions has received little attention in the domain of academic search. Conversion in this setting refers to the download action of a paper, which happens when the user finds relevant information and wants to save it for later use. Similar to purchases, downloads of papers generate revenue for the academic search platform, often through a subscription service or pay-per-download. Therefore, it is valuable to study the download behavior of users and understand users' behavioral patterns. E.g., what actions do users perform that lead to a download? Is there a temporal pattern in downloads? Are there behavioral differences among users with different topical interests?

To start, we conduct an observational study to characterize user download behavior in academic search. To the best of our knowledge, this is the first characterization of its kind in the area of academic search. While some of the findings may coincide with

* Corresponding author.
E-mail addresses: x.li@uva.nl (X. Li), derijke@uva.nl (M. de Rijke).

https://doi.org/10.1016/j.ipm.2018.10.019
Received 22 April 2018; Received in revised form 21 October 2018; Accepted 26 October 2018

the personal experiences of readers of this paper, many findings provide insights that can only be found from a large user base.

Using the insights obtained in this manner, we try to generate predictions of users' download behavior. We are motivated by the information overload problem in online recommendations. Numerous studies have shown that information overload has a negative impact on user reactions (Aljukhadar, Senecal, & Daoust, 2010; Goswami, 2015; Li, 2016). It has been observed in large-scale experiments that showing an excessive number of paper recommendations may not bring benefits to the click through rate, but instead bring harm (Beierle, Aizawa, & Beel, 2017). In our setting of predicting download behavior, one task is to predict how many downloads the user is going to have next.[1] An application scenario for this prediction task is when the system sends out recommendations in the form of news letters, such as the recommender system on Mendeley.[2] If the system is able to predict the magnitude of the user information need, it can better tailor the length of the list of recommendations, hence avoiding information overload and leading to a better user experience. Moreover, time is an important source of information for understanding user satisfaction (Borisov, Markov, de Rijke, & Serdyukov, 2016): finding the right timing for a recommendation may also improve the performance of recommender systems (Dali Betzalel, Shapira, & Rokach, 2015). Correspondingly, we address the task of predicting the time gap until the next download, aiming to make the system more preemptive and send recommendations when users are in need.

In this paper, we provide answers to the following research questions:

1. What are the user actions that lead to a download in academic search?
2. What are behavioral patterns and topical aspects of user download behavior across sessions?
3. How do we predict user download behavior?

Our main contributions are:

1. We introduce a new dataset for downloads in academic search and characterize user interactions with academic search engines.
2. We study the users' actions across sessions, revealing correlations among various behavioral signals and explaining the topical aspects of user downloads.
3. We build a specialized model for download prediction that utilizes user session history and that is based on user segmentation, which leads to significant improvements over a state-of-the-art baseline.

2. Related work

Related work comes in several kinds: academic search, academic paper recommendation, and online shopping prediction.

2.1. Academic search

Academic search concerns the task of indexing and retrieval of entities (papers, journals, ...) in the domain of academic research. Academic search services are commonly provided by academic search engines, such as Google Scholar, Microsoft Academic Search (Sinha et al., 2015), AMiner (Tang, 2016), and CiteSeerX (Li, Councill, Lee, & Giles, 2006). Several studies have indicated that academic search engines are an essential portal for obtaining research information (Hemminger, Lu, Vaughan, & Adams, 2007; Niu & Hemminger, 2012; Niu et al., 2010). Mitra and Awekar (2017) study the search results of several academic search engines and find that they have low overlap. Other research concerning user behavior in academic search occurs mostly via surveys (Pontis & Blandford, 2015; Pontis, Blandford, Greifeneder, Attalla, & Neal, 2015) or small-scale log analyses on high-level statistics (page views, access frequencies etc.) (Ke, Kwakkelaar, Tai, & Chen, 2002). They are either restricted to a small group of participants, or to users from a single discipline, which renders the findings less generalizable. Nicholas et al. (2008) study the full-text viewing behavior in journal libraries and find that the viewing habits vary greatly among scholars. Recently, Xiong, Power, and Callan (2017) have proposed to use entity embeddings to improve relevance ranking of papers in academic search, which leads to better performance on a test set of 100 queries from their transaction log.

There are very few studies on academic search that are based on a large-scale transaction log. In recent work, Li, Schijvenaars, and de Rijke (2017) and Li and de Rijke (2017) study the phenomena of null queries and topic shifts in academic search, respectively, based on large-scale log analyses. Khabsa, Wu, and Giles (2016) study the distribution of academic search queries on Microsoft Academic Search and build a classifier for different query types. To the best of our knowledge, no large-scale study has been conducted on download behavior in academic search.

2.2. Academic paper recommendation

Paper recommender systems provide users with relevant paper suggestions, preferably personalized to their own interests.

[1] Note that we do not consider the "zero download" scenario in our setting, which would be formulated as a different problem: churn prediction. Our focus is on users who regularly use the academic search service.

[2] Mendeley (https://www.mendeley.com/) provides personalized paper recommendations through news letters, based on users' interactions with the system.

Gori and Pucci (2006) use random walks on citation graphs to make paper recommendations. Li, Yang, and Zhang (2013) propose to recommend papers using matrix factorization combined with topic modeling. They find that topic representations for users can help distinguish users with different interests, and surface better suggestions. Nishioka and Scherp (2016) use social media streams to profile users, and recommend papers based on the profiles. Sun et al. (2018) study research networking sites and leverage social network connections for paper recommendations. Beierle et al. (2017) demonstrate how recommendation overload affects click through rate. Through 3.4 million delivered recommendations, they find lower click-through rates for higher numbers of recommendations; users can feel "overloaded" rather quickly. Ollagnier, Fournier, and Bellot (2018) propose to recommend papers with a new bibliometric measure based on the papers that users are reading.

2.3. Online shopping prediction

We introduce related work on online shoppers on e-commerce sites because online purchases share important commonalities with academic downloads: (1) they both represent a conversion after user interactions with the system; (2) both scenarios come with a "budget." Money is the budget factor in online shopping, while in academic search it could be money (subscription service or pay-per-download) and time (assuming that users are aware of the finite amount of time they have to read the papers). Lee et al. (2015) examine the purchase behavior and trajectory of users, and use behavioral features for purchase predictions of items. Kooti et al. (2016) extract user purchase histories from emails to analyze their purchase behavior. They find that previous purchase history information helps to predict time and price of the next purchase. Kooti, Grbovic, Aiello, Bax, and Lerman (2017) study online shopping behavior in app stores. They discover that 1% of the users account for 59% of the total spending in app purchases, and that they behave very differently from a random user in shopping. For these 1% users, Kooti et al. (2017) propose a supervised model to generate shopping predictions. Yeo, Kim, Koh, Hwang, and Lipka (2017) study purchase prediction for retargeting, by using purchase features extracted from users' browsing history.

In summary, this paper differs from previous work in academic search because it studies download behavior as opposed to other user behavior. Compared to other high-level log analyses, this paper provides insights into user actions within sessions and across sessions. It is also based on a large transaction log of a popular academic search engine rather than small-scale user studies or surveys, hence bringing findings that are more generalizable. This work is directly related to paper recommender systems. Whatever the implementations of a paper recommender system may be, they all need to consider information overload and recommendation timing. Therefore, they can benefit both from our characterization of download behavior, and predictions of the download number and time gap.

3. User download patterns

In this section we present observations of user download actions in academic search. The definition of a download here is the act of requesting a PDF file for a paper.[3] We study search sessions that include at least one download action. The search sessions are characterized by entering a query as the first interaction, and they end with a cutoff time of 30 min inactivity that is commonly used in web session analyses (Catledge & Pitkow, 1995; Srivastava, Cooley, Deshpande, & Tan, 2000).

We first introduce the dataset and the various actions that users perform within a session. We analyze the action statistics as well as the action trajectories that lead to a download action. Then we uncover download patterns across sessions.

3.1. Dataset and user action definitions

3.1.1. Dataset

To study users' download behavior, we use a transaction log provided by ScienceDirect,[4] which offers academic search services and primarily covers the domains of health science, life science, physical science and social science. Collected between September 28, 2014 and March 5, 2015, the log contains more than 39 million queries via institution-authorized access as well as personal access. The former access type refers to users in a certain IP range (e.g., from a research institution), who are referred to as *IP-users*. The latter refers to users who log in or access the search engine from outside the institution, i.e., so-called *non-IP users*. Two thirds of the query traffic comes from IP-users.

For the purpose of studying and predicting user downloads, we filter the logs based on two rules: (1) users are uniquely identifiable, so that we can distinguish them from each other, and (2) users are active in terms of issuing queries and requesting downloads, in order to guarantee enough observations for our study. IP-users from the same institution may end up having the same user ID or session ID. Therefore, we look at non-IP users only to ensure that each user ID maps to a unique user. The majority of these non-IP users have access via subscription, and the rest through pay-per-download. We then select active users that have a minimum of 30 queries in a timespan of 30 days, and a total of at least 20 download sessions in the period covered by the log. To prevent the inclusion of bots/crawlers, we also remove overly frequent users that have more than 1000 queries recorded in the log or with more than 100 clicks/downloads on average per day, which account for fewer than 0.1% of the users. We end up with 1089 users and

[3] The dataset used in our study also includes download actions of less importance, such as downloading references, that we include in our study without focusing on.

[4] http://www.sciencedirect.com/

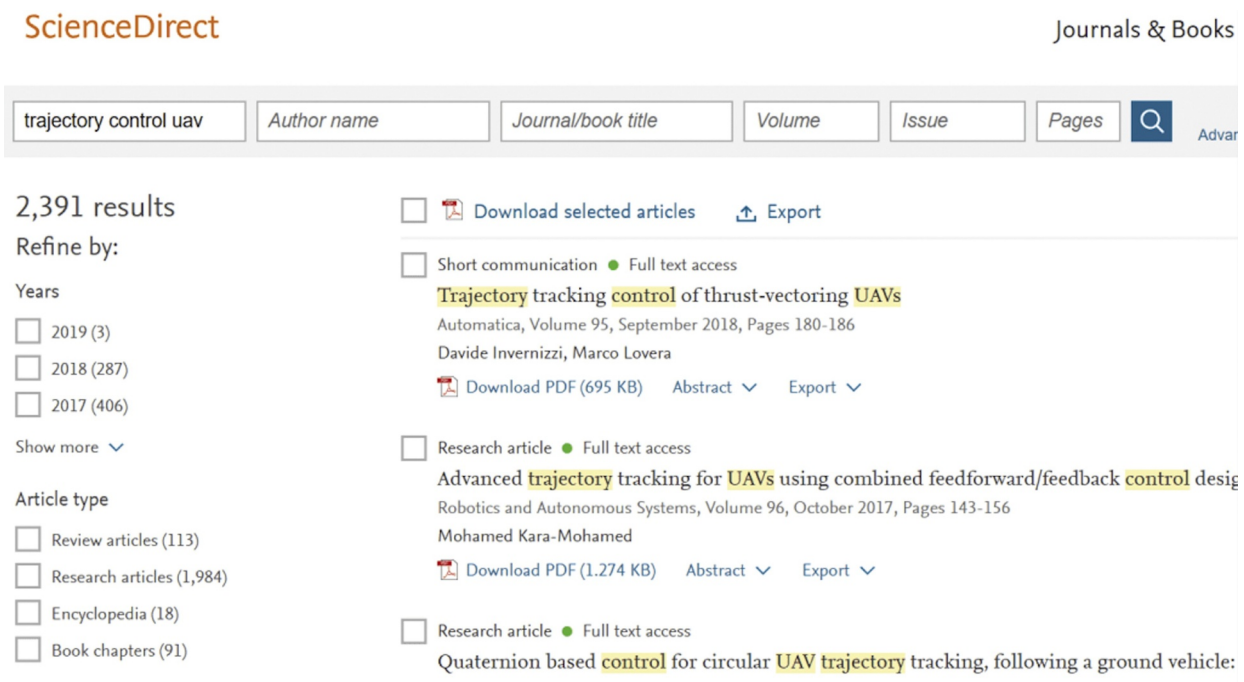


Fig. 1. ScienceDirect search user interface.

30,988 sessions that include at least one download action, referred to as *download sessions*. There are a total of 206,830 download actions, i.e., an average of 190 downloads per user. The above data selection process provides us with enough observations per user to study their download behavior.

3.1.2. User actions

After a user issues a query, papers are shown on the search engine result page (SERP) in a list style. Then the user can take several subsequent actions: (1) click on a paper title for detailed information, which opens up a new window, (2) directly click to download a paper by clicking the PDF button next to a paper, or click on the abstract button next to a paper to see paper abstract in an expanded panel (while still staying in the SERP). We show a snippet of the user interface in Fig. 1 and summarize the actions of interest in Table 1.

"Abrf click" (5) is an action similar to "HTML click" (4) that leads to a paper page without full text but with a scanned image of the paper content. "HTML click" (4) and "Abrf click" (5) are triggered when users click on a paper's title on the SERP. Users will notice the difference after the click but not beforehand. "Change query source" (3) should not be confused with query reformulation which refers to typing a new query.

While academic search engines have different user interfaces, most of them provide similar high-level functionalities as the ones we list in Table 1: the user actions available on ScienceDirect resemble competing services. Therefore, despite some small differences between functionalities of popular academic search engines, we believe that the insights learned through this study have generalizable implications for academic search engines as a whole.

3.2. Download behavior within a session

What actions do users like to perform in a session?

Table 1
Possible user actions.

	Action	Explanation
1.	Query	user issues a query
2.	Download PDF	user requests a PDF version of a paper
3.	Change query source	user changes the source of a previous query, i.e., selecting different journals or subjects for the query.
4.	HTML click	user clicks on a paper on the SERP (search engine result page) for detailed information, which opens a new window
5.	Abrf click	similar to "HTML click" except that the clicked result does not contain full text
6.	Abstract click	user click to see the abstract of a paper on the SERP
7.	Reference download	user downloads the reference of a result

Table 2
Statistics of user actions in a session.

		Mean		Median
1.	Query	2	.82	2
2.	Download PDF	6	.45	3
3.	Change query source	0	.46	0
4.	HTML click	1	.49	0
5.	Abrf click	0	.30	0
6.	Abstract click	0	.52	0
7.	Reference download	0	.08	0
Query dwell time (s)		567		323
Click dwell time (s)		734		397
Session duration (s)		1336		754

We find that the most frequent user action is “download,” followed by “query,” shown in Table 2. It should be noted that users tend to have multiple downloads within a single session, with the median number being 3 per session. This can be explained by the richness of informational queries in academic search (Li et al., 2017), which users issue to search for relevant information around a certain topic. Interestingly, clicks have lower occurrences than queries, and are often absent in sessions. This suggests that clicking results to view detailed information may not be necessary for users to make a download decision, while partial information (title, authors) already provides enough cues of relevance. For sessions that contain a click, users tend to spend relatively much time inspecting detailed information (e.g., glancing over the full text), with the median click dwell time being over 6 min.

Table 3 lists the frequent action trajectories toward a download in a session. The most frequent trajectory is a single query (1) leading to a download (2), making up 30.3% of all trajectories. Ranking second is a query (1) and a query reformulation (1) leading to a download (2). The trajectories involving clicks are far less frequent. These observations indicate that queries are acting as a more common signal toward downloads than clicks.

Below, we give an example of a download session sampled from the log to illustrate the process:

28Nov2014:16:22:13	Query (1)	dynamic friendship network
28Nov2014:16:23:40	Query (1)	dynamic friendship network model
28Nov2014:16:24:34	Abrf click (5)	shorturl = /scie...pii/0378873394002467
28Nov2014:16:25:47	Download PDF (2)	shorturl = /scie...pii/0378873394002467/pdf

In this session, the user starts with the query “dynamic friendship network” and proceeds with the query reformulation “dynamic friendship network model.” Then, the user clicks on a result by performing an “Abrf click,” and after examining the result for a while chooses to download the PDF file.

3.3. Download behavior across sessions

Next, we go beyond individual patterns and look for temporal patterns and correlations across sessions.

3.3.1. Temporal patterns

Looking at download numbers on different days in a week, we observe a steady trend during weekdays, while the number declines in the weekends; see Fig. 2. This trend is similar to the e-commerce setting where more purchases happen during weekdays than weekends (Kooti et al., 2016).

Table 3
Top 10 most frequent action trajectories toward the first download in a session. Actions are numbered as in Table 1: 1. Query; 2. Download PDF; 3. Change query source; 4. HTML click; 5. Abrf click; 6. Abstract click; 7. Reference download.

Trajectory	Frequency
1 → 2	30.3%
1 → 1 → 2	8.7%
1 → 4 → 2	4.4%
1 → 1 → 1 → 2	3.7%
1 → 3 → 2	2.1%
1 → 1 → 1 → 1 → 2	1.8%
1 → 1 → 4 → 2	1.3%
1 → 5 → 2	1.3%
1 → 1 → 1 → 1 → 1 → 2	0.9%
1 → 4 → 4 → 2	0.9%

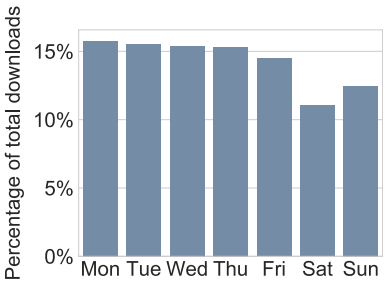


Fig. 2. Download distribution over the week.

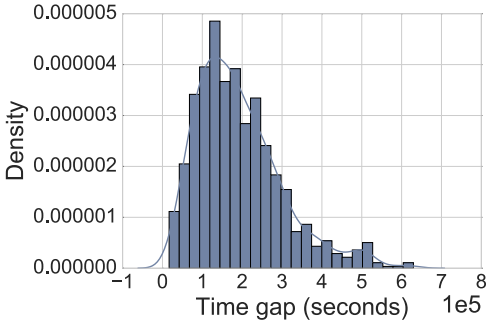


Fig. 3. Distribution of time gap between consecutive download sessions averaged per user.

Table 4
Description of factors in Fig. 4.

Name	Description
numQuery	Number of queries
numClick	Number of all clicks
numDownload	Number of PDF downloads
numChangeSrc	Number of change query source actions
numAbstClick	Number of abstract clicks
numAbrfClick	Number of Abrf clicks
numHtmlClick	Number of Html clicks
numAbstDownload	Number of abstract downloads
dwellTimeQ	Averaged dwell time on queries
dwellTimeC	Averaged dwell time on clicks
wLength	Average query length in number of words
cLength	Average query length in number of chars
wholeTime	Session duration
timeTillNext	Time gap until the next download session
nextDownloadNum	Number of PDF downloads in the next download session

However, we find that academic searchers take longer to perform the next conversion action than online shoppers. This is evident from the time gap between download sessions. Fig. 3 shows the distribution of the averaged time gap[5] for each user. It has a median of 172,915 s and a mean of 192,532 s (2.00 and 2.23 days respectively), while in online purchases the median time gap is 1 day (Kooti et al., 2016).

3.3.2. *Correlations between sessions*

We are interested to find out connections among sessions, that is, how one session impacts another. We aim to answer questions such as: if a user has performed many actions (e.g., downloads) in the current session, will the activity intensity sustain in the next session? And will the next download happen in a shorter time gap or a longer one?

We examine two types of correlation: (1) the correlation between the current session and time until the next download session; and (2) the correlation between the current session and the number of downloads in the next download session. We consider several factors in the current session, including user action statistics (Table 2) and query statistics (average word/character length of queries). Table 4 gives a description of the factors. The correlations are shown in Fig. 4.

[5] We average the time gap for each user to avoid the bias toward active users that have many sessions.

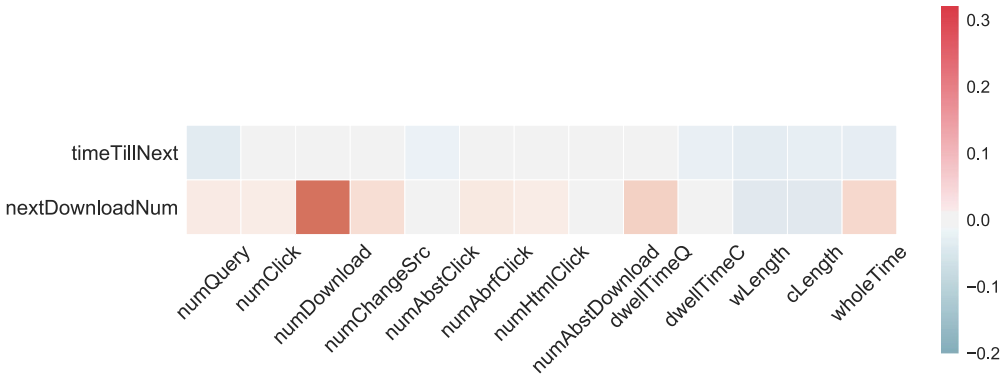
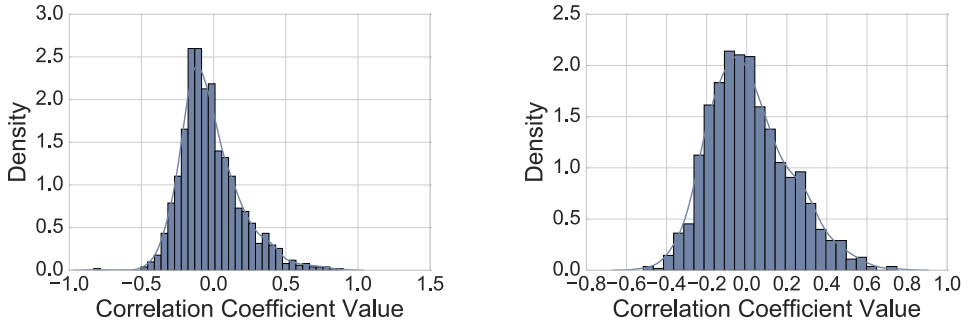


Fig. 4. Correlations between statistics of the current download session (horizontal) and of the next download session (vertical).



(a) Correlations between time until next download and the number of queries in the current session.

(b) Correlations between the number of downloads in the next session and that in the current session.

Fig. 5. Distribution of correlation coefficients for individual users.

In Fig. 4, factors in the current session are all negatively correlated with the time until the next download (Landis & Koch, 1977). Out of all the factors, the number of queries is the most negatively correlated ($p < 0.0001$, two-tailed t-test). This suggests that the more queries the user submitted in the current session, the sooner her next download session might occur.

Conversely, the number of downloads in the next session are positively correlated with most of the current session factors. The prominent factor is the current session's number of downloads, which has a medium positive correlation with that of the next session ($p < .0001$, two-tailed t-test). This indicates a certain degree of consistency between the number of downloads across sessions.

The above correlations are calculated from sessions of all users. Therefore they represent the overall trend from all observations. Next, we examine correlations at the individual level. For the two correlations that we calculate, i.e., time and the number of downloads, we examine the most negative factor "number of queries" for time until next download and the most positive factor "number of downloads" for the next number of downloads, respectively. We examine each user's sessions and obtain the two correlation coefficients. We show the distributions of the correlation values in Fig. 5. Both correlation values are nearly normally distributed but the means differ. More than half of the users show a negative correlation between time and query in Fig. 5a, which is in line with the overall trend in Fig. 4. However, the distribution in Fig. 5b indicates that nearly half of the users tend to be consistent in the number of downloads between consecutive sessions, and half do not. Bias explains why the overall correlation is positive in Fig. 4 while the individual correlation distribution disagrees: users with positive correlations have more sessions in the log, thus affecting the overall correlation.

3.4. How topics impact downloads

In this section we discover the topical characteristics of the user behavior. Compared to clicks on a result page, we believe that downloads are less noisy, and are stronger signals to reflect users' topic interests. Therefore, we represent the topics using downloads. To identify the topics of user downloads, we resort to the Scopus classification of subject areas.[6] Specifically, we look at the journal

[6] https://www.elsevier.com/solutions/scopus/content

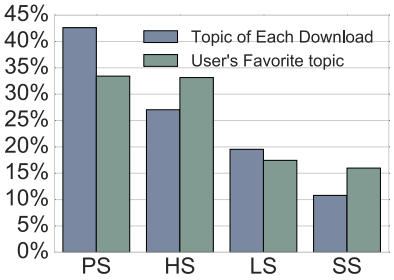


Fig. 6. Topic distribution of each download and each user's favorite topic respectively. PS: physical sciences, HS: health sciences, LS: life sciences, SS: social sciences.

Table 5
Statistics of sessions grouped by users with different topical interests. PS: physical sciences, HS: health sciences, LS: life sciences, SS: social sciences.

	Mean								Median			
	PS		HS		LS		SS		PS	HS	LS	SS
Query	2	.82	2	.68	2	.95	2	.94	2	2	2	2
Click	2	.08	2	.27	2	.06	3	.09	0	0	0	1
Download	8	.04	5	.65	6	.31	4	.86	3	3	3	2
Duration (s)	1330		1282		1384		1408		745	718	806	819
Time between download sessions (s)	176,655		180,134		166,943		178,190		60,009	64,984	53,764	56,102

where a paper is published and use the subject area of the journal to represent the topic. The subject information of journals is manually annotated and publicly available through API access.[7] The subjects fall into 4 broad topics (health sciences, life sciences, physical sciences and social sciences) and a total of 333 specific categories. We use the subject information to represent topics because it is manually annotated and easily interpretable, and is more accurate than topics inferred by topic models such as LDA (Blei, Ng, & Jordan, 2003).

Are certain topics more popular than others? Fig. 6 shows the distribution of the topic of each download record, and the distribution of each user's most popular topic (determined by most frequent download type). While both distributions are heavily imbalanced, physical sciences is the most popular subject and social sciences is the least. This finding is in line with the focus of ScienceDirect on natural science journals. Below, we look at users with different topical interests and examine their behavior and topical differences.

3.4.1. Behavioral differences

We hypothesize that distinct topical interests may come with different download patterns and examine the behavioral differences among users with different topical interests. To examine behavior, signals such as clicks, downloads, session duration and time until the next download session are investigated, as shown in Table 5.

Users interested in the social sciences stand out from the others: they do not perform as many downloads compared to people interested in other subjects; however, they have more clicks and spend more time in sessions. As to the time between download sessions, the health sciences have the longest time gap, while the life sciences have the shortest. And there is a 21% difference between the median values, which is 3 h. In summary, download behavior indeed varies among user groups interested in different subjects.

3.4.2. Topical profiles of users

First, we examine the interdisciplinary nature of users: how many unique topics out of the 4 general topics do users cover through their downloads? We show the distribution in Fig. 7. Most users cover more than one topic in their downloads. This is not surprising, as recent findings (Porter & Rafols, 2009) suggest that "science is indeed becoming more interdisciplinary." Therefore, users may have information needs for multiple topics when conducting increasingly interdisciplinary research. Besides, some of the users might be research policy designers who need to study several topics.

Furthermore, we try to find out the differences between users that are interested in each of the 4 topics. We first look at the topical diversity of users, i.e., how many unique subtopics users cover through their downloads. Each subtopic corresponds to a specific subject area. Note that users may cover the subtopics of multiple domains. We find that users in the life sciences domain explore the largest number of subtopics with a mean of 24 and a median of 22, while users in the health sciences and in the social sciences explore the fewest, with mean and median values being roughly on par with each other, shown in Table 6. This shows that users's topical diversity differs across their disciplines.

[7] https://dev.elsevier.com/sc_apis.html

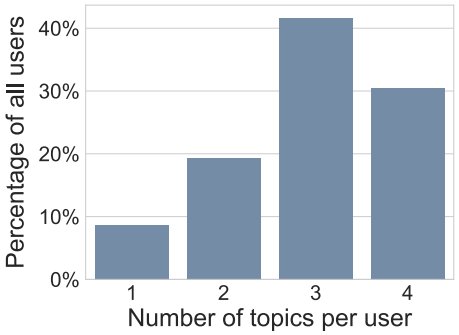


Fig. 7. Distribution of the number of topics per user.

Table 6
Statistics of unique subtopics covered per user.

Category	Min	Max	Mean	Median
Physical sciences users	2	208	24	19
Health sciences users	2	76	21	17
Life sciences users	2	101	24	22
Social sciences users	1	83	20	18

Next, we consider topical coherency in downloads. Consider two users who cover the same number of subtopics: one may have downloaded only a few papers but switches topic whenever possible, while the other may have downloaded many but does not switch subtopics as often. Since they cover the same number of subtopics, we would consider the second user to be topically more coherent due to a smaller number of topical switches. We design a topic coherency metric that measures the likelihood of staying in the same subtopic(s) in downloads as shown below. The higher the score, the more likely the user stays in the subtopic(s).

$$\text{Topical Coherency} = 1 - \#unique\_subtopics/\#downloads$$

Here, #unique_subtopics refers to the unique number of subtopics under one of the 4 general topics. Topic coherency is computed per individual user.

Table 7 indicates that users in the physical sciences have the highest topical coherency with a mean score of 0.832 and a median score of 0.846. This might seem surprising because these users exhibit a high level of diversity of downloads (in Table 6). However, they also commit the largest number of downloads (in Table 5), and they do not switch topics as often as others, and hence they obtain the highest coherency score. On the other hand, users in the life sciences have the lowest topical coherency scores while also being the most diverse in download subtopics (Table 6).

3.5. Upshot

We have introduced user actions during a download session and investigated their frequencies, trajectories, cross-session correlations, and topical impacts on paper downloads. We have found that certain signals are indicative of downloads, and users with different topical interests tend to behave differently. Next, we use those observations, especially behavioral and topical features, for predicting paper downloads.

4. Download prediction models

In this section, we describe our models for paper download prediction. As a baseline, we adopt a state-of-the-art model for predicting online shoppers' behavior (Kooti et al., 2016), since this task bears resemblance to our download prediction task (as explained in Section 2).

Then we propose an LSTM-based model to effectively leverage users' historical interactions, as well as a specialized model based

Table 7
Topical coherency per user.

Category	Mean	Median
Physical sciences users	0.832	0.846
Health sciences users	0.826	0.833
Life sciences users	0.807	0.815
Social sciences users	0.817	0.833

on user segmentation.

We consider two prediction tasks: given a user's previous download sessions, (1) predict the time until the next download session, and (2) predict the number of downloads in the next download session. More formally, for each user u, the training sessions ordered by occurrence are denoted as $u_d = \{s_1, s_2, ..., s_{n-2}\}$, where n is the total number of sessions the user has. In testing, for each user u and given a session s_{n-1} as input, our models need to predict the number of downloads in s_n and the time gap between the end of s_{n-1} and the start of s_n.

4.1. Baseline model

The baseline model (Kooti et al., 2016) considers shopping prediction as a multi-class classification task. It uses a Bayesian network classifier and a set of features derived from the online shopping setting. The features used in Kooti et al. (2016) include demographics of online shoppers, purchase price history, purchase time history etc. Although it is not possible to directly apply those features in the academic search setting, in some cases it is possible to identify natural counterparts. Specifically, purchase time features can be mapped to download time features, and purchase price features can be mapped to features of the number of downloads. Other features such as queries and other actions (download references etc.) are exclusive to our setting. The baseline Bayesian network model does not allow arbitrary number of inputs,[8] while our LSTM-based models do and are able to take input from the full session history. To make the comparison fairer, we also include aggregated features from the full session history for our baseline. In the end, the following features are used for the baseline Bayesian network predictive model:

1. Current session action features: the number of occurrences of queries, clicks, downloads, change query source, abstract click, HTML click, Abrf click, abstract downloads.
2. Current session time features: dwell time on queries, dwell time on clicks, session duration.
3. Query features: average word and character length of queries.
4. Historical session features: time gap from the last download session, average/median/standard deviation of time gap between consecutive download sessions; number of downloads in the last download session, average/median/standard deviation of the number of downloads in historical download sessions.

Formally, each session s_i is represented as a feature vector and a label: $s_i = \langle f_i, l_i \rangle$, where s_i is the ith session of the user, f_i is the feature vector for the session, and label l_i corresponds to the label for prediction, i.e., the time gap until the next session s_{i+1} or the number of downloads in s_{i+1}. In testing, the model is given f_{n-1} to predict l_{n-1}.

4.2. LSTM model

The second model we consider is an LSTM (long short-term memory), a recurrent neural network model proposed by Hochreiter and Schmidhuber (1997). Through its memory cells and gate architecture (input, forget and output gates), an LSTM is able to alleviate the vanishing and exploding gradients problem that exists in simple recurrent neural networks. LSTMs are known to perform well for tasks that deal with long sequences. Motivated by the correlations of behavioral statistics across sessions (explained in Section 3), we use an LSTM to model the paper download prediction problem in order to utilize the full session history for prediction. Specifically, sessions of users are modeled as sequences, each consisting of a feature vector and a label (either time or number of downloads). The features are the same as those in the baseline model. The LSTM model takes the sessions of each user as input and learns to predict the label.

Formally, the training and testing cases are defined similar to the baseline model's setting, except for test sessions. In testing, the model is given $\{s_1, s_2, \cdots, s_{n-2}, f_{n-1}\}$, where f_{n-1} is the feature vector for session s_{n-1}, as input in order to predict label l_{n-1}. In training, we optimize for multi-class cross entropy. We choose Stochastic Gradient Descent with Nesterov momentum as the learning algorithm, and use mini-batches. We initialize the network parameters via Xavier initialization (Glorot & Bengio, 2010), and hyper-parameters such as learning rate are tuned via grid search.

4.3. Specialized model based on user segmentation

In mobile shopping prediction, user segmentation has been considered (Kooti et al., 2017) as some users may behave differently than others, and specialized models are built for them. In Fig. 5 we noticed that individual users may have different download patterns, reflected by the varying correlations of download behavior across sessions. E.g., after a session with many downloads, some users tend to have fewer downloads in the next session, but some may not. In our setting, the LSTM model should in theory learn to distinguish between these different patterns of users. However, it may not work as well on time series data of unequal lengths. E.g., it is known that in terms of classification tasks for unequal-length time series, DTW (dynamic time warping) (Berndt & Clifford, 1994) outperforms LSTMs on some occasions (Fiterau et al., 2017; Lei, Yi, Vaculin, Wu, & Dhillon, 2017) due to its ability to consider warping in time series. To improve our prediction performance, we segment users and build specialized models for them. Specifically, we segment our users into clusters by behavioral similarity measured via DTW, and then train specialized LSTM models on the user

[8] Users may have different numbers of sessions.

clusters. In this way we are giving special treatments to users that are similar, who share behavioral patterns.

We use DTW because it is able to effectively handle time series of different lengths, which allows for stretching or compressing sequences while comparing similarity. Specifically, DTW is set to find the minimum warping distance between two series P of length n and Q of length m:

$$\begin{aligned} \mathrm{P} &= p_1, p_2, \cdots, p_n \\ \mathrm{Q} &= q_1, q_2, \cdots, q_m. \end{aligned}$$

An n-by-m matrix is constructed where each element (i, j) corresponds to the squared distance between p_i and q_j. The goal is to find a path W through the matrix that minimizes the accumulated distances

$$\mathrm{DTW}(P, Q) = \min\left\{\sum_{i=1}^{K} w_k\right\},$$

where w_k is the kth element on the warping path W. Then, the warping path can be solved recursively:

$$\gamma(i, j) = d(p_i, q_j) + \min\{\gamma(i-1, j-1), \gamma(i-1, j), \gamma(i, j-1)\},$$

where $d(p_i, q_j)$ is the distance between p_i and q_j, and $\gamma(i, j)$ is the cumulative distance.

To measure distances between users, we model each user's download behavior as a time series: the number of downloads and time between download sessions. Notice that in our setting users may have different numbers of sessions. We use 1-nearest neighbor DTW to obtain the distance between any pair of users, which ensures good warping accuracy. Then we apply average linkage hierarchical clustering on users based on the distances. The LSTMs are subsequently trained on clusters that represent users with similar download behavior. Each cluster contains a minimum of 10% of the total number of users.

5. Experiments and results

In this section we present the experiments and results of download prediction on two tasks: (1) predicting the time until the next paper download session and (2) predicting the number of paper downloads in the next download session.

5.1. Experimental setup

Similar to Kooti et al. (2016), we cast the download prediction task as a multi-class classification task instead of a regression problem, because it is difficult to predict the exact time. For time prediction we divide the time gaps into 5 classes. The time gaps and their distribution in the dataset are described as follows: very short (within 2 h, 20.6%), short (2 h to 1 day, 30.4%), median (1 day to 3 days, 18.4%), long (3 days to 7 days, 14.2%) and very long (over 7 days, 16.4%). For predicting the number of downloads, we segment the number of downloads into three classes: 1 download (30.0% of all sessions), 2–4 downloads (36.8%), and ≥ 5 downloads (33.2%).

We use the 1089 users and 30,988 sessions described in Section 3.1. We segment it into training and testing data following the description in the prediction model section.[9] We test the statistical significance of observed differences in predictions using a paired Wilcoxon signed-rank test. We denote significant differences between the baseline and other methods using * for $\alpha = .05$ and ** for significance at $\alpha = .01$. We denote differences between the LSTM with user segmentation and all other methods using $^+$ for $\alpha = .05$ and $^{++}$ for significance at $\alpha = .01$.

5.2. Experimental results

5.2.1. Baseline

For predicting the time until the next download session, the baseline model yields a prediction accuracy of 0.347; see Table 8, first row. This score is comparable to that achieved by the baseline model in the online shopping time prediction task (5 class classification, 0.311 accuracy, (Kooti et al., 2016)).

For predicting the number of paper downloads in the next session, the baseline achieves an accuracy of 0.441; see Table 9, first row.

5.2.2. Time series based models

Next, we present the results of the LSTM models. To determine how historical session information impacts prediction, we control the number of session inputs during testing. We hypothesize that in testing, feeding the network with the full session history will lead to better predictions than feeding only partial session information.

The results of predicting the time until the next download are shown in Table 8, rows 2–5. All LSTM models perform significantly better than the baseline, even when only using information from the current session as test input. But the performance gap between

[9] Due to the dependency in the time series, we split the data by time so that the models learn from historical information and predict the future download.

Table 8
Predicting the time until the next paper download with the baseline and LSTMs.

Model	Accuracy
Baseline	0.347
LSTM current session	0.354**
LSTM current session + 1 previous session	0.354**
LSTM current session + 2 previous sessions	0.354**
LSTM full session	0.357**
LSTM full session + user segmentation	0.371**++

Table 9
Predicting the number of paper downloads in the next download session with the baseline and LSTMs.

Model	Accuracy
Baseline	0.441
LSTM current session	0.453
LSTM current session + 1 previous session	0.462*
LSTM current session + 2 previous sessions	0.463*
LSTM full session	0.464**
LSTM full session + user segmentation	0.481**++

using different numbers of historical session inputs in testing is small. One explanation is that the LSTM model is already capable of "memorizing" the dependencies across sessions in training. Therefore, it can obtain a good prediction performance even without using the full session history in testing. Compared to the baseline, the LSTM model with full session history gains improvements in predicting very short and short time gaps (time gaps defined in Section 5.1), with an increase in accuracy of 18.2% and 20% for the 2 classes respectively, while performing worse in other classes.

The results of predicting the number of downloads (Table 9, rows 2–5) show a similar pattern as those for the time prediction tasks. All LSTM models perform better than the baseline, with an increase coming from predicting single download sessions (+23.8%) and 2–4 downloads sessions (+2.6%). However, here historical session information leads to significant improvements over the LSTM models without them. Using full session history significantly improves the performance over models using only part of the session history.

5.2.3. Specialized model based on user segmentation

Both for predicting the time until the next download session (Table 8, row 6) and predicting the number of downloads in the next download sessions (Table 9, row 6), the LSTM model with user segmentation performs significantly better than the baseline and other LSTM models. This should not come as a surprise as we notice the differences of user behavior separated by clusters, for instance, the median of user download time gaps varies significantly across clusters, ranging from 53,353 to 67,147 s. It would be better for the prediction models to train on users that are similar in behavior, rather than on a mixture of different users. This explains the performance increase by applying user segmentation.

5.2.4. Additional topical feature

In Section 3.4.1 we have seen that there are behavioral differences among users with different topical interests. We hypothesize that using the topical interests of users would help download prediction. Next, we examine whether topical features improve performance on our download prediction tasks. We augment the models considered so far with an additional categorical feature indicating which of the 4 general topics the user is most interested in. The results for predicting the time until the next download session are shown in Table 10.

The addition of a topical feature leads to significant improvements, both with and without user segmentation. A similar conclusion can be drawn when predicting the number of downloads in the next download session, as shown in Table 11.

Table 10
Predicting time until the next download with an additional topical feature. We denote significant differences after using the topic features with ** for significance at $\alpha = .01$.

Model	Accuracy
LSTM full session	0.357
LSTM full session + topic feature	0.360**
LSTM full session + user segmentation	0.371
LSTM full session + user segmentation + topic feature	0.376**

Table 11
Predicting the number of downloads in the next download session with additional topical feature. We denote significant differences after using the topic features with ** for significance at $\alpha = .01$.

Model	Accuracy
LSTM full session	0.464
LSTM full session + topic feature	0.466**
LSTM full session + user segmentation	0.481
LSTM full session + user segmentation + topic feature	0.485**

As we discussed in Section 3.4, the topical feature in an academic search setting can be a useful indicator of behavioral patterns. Here, the performance boosts show its utility for the two download prediction tasks that we consider. In both tasks, three additions gave us cumulative boosts in performance: (1) switching to LSTMs; (2) employing user segmentation; (3) adding a topical feature, where the most significant improvement comes from the LSTM model with user segmentation.

6. Conclusion

We have studied the download behavior of users of an academic search engine, a type of conversion behavior that has not yet been well examined in the literature. We first conducted a thorough observational study. We introduced a new dataset for user download behavior, defined user actions during a session, and showed action trajectories toward a download. Then we examined cross session download behavior, which was our main focus. We identified temporal patterns in users' download behavior. We also discovered multiple correlations of user behavior across sessions. Certain behavioral factors such as the number of downloads are correlated across sessions. The time gap until the next download session is negatively correlated with the number of queries.

We also examined topical aspects of downloads and their impacts on download behavior. We used annotated topical information of journals to classify the topics of users' downloads. We have found a bias in the distribution of topics of downloads, where the natural sciences (physical, life and health sciences) outnumber the social sciences. Furthermore, we identified behavioral variances in terms of download diversity and coherence between users who are interested in different topics. Not only do users download papers across subtopics, but they also download across disciplines, which confirms recent findings that academic research is becoming increasingly interdisciplinary.

Building on the insights gained from our observations, we moved on to two download prediction tasks: predicting the time until the next paper download session and predicting the number of downloads in the next download session. These two tasks help alleviate the information overload problem in academic recommender systems, as well as making the predictions pre-emptive in terms of their timing. We proposed a model based on LSTMs to utilize users' full session history for prediction, which gave rise to significant improvements over a state-of-the-art baseline method developed for a similar problem. Motivated by the observed differences in individual behavioral pattern, and the ability of dynamic time warping (DTW) to measure the similarity between time series, we built specialized models based on user segmentation with DTW. The specialized models showed significant improvements, indicating that user segmentation with DTW is beneficial. Last but not least, we established the usefulness of topic features in download prediction.

As to future work, we would like to pursue three main themes. First, our predictions of the number of paper downloads and time gap can help paper recommender systems handle information overload and recommendation timing; we plan to include such predictions as signals in an online academic paper recommendation system. Second, so far we have considered only a "general" topical interests rather than fine grained subtopics in predicting downloads. This is due to sparsity of paper downloads per subtopic in our dataset, e.g., we have no download records for some subtopics. The subtopics, or certain types of journals may have their unique cycle of publishing papers. They can potentially impact user download patterns. We hope to collect data that will allow us to fully explore the impact of subtopics on download behavior. Third, while we have closely studied users' interaction behavior, we have not captured it with click models (Chuklin, Markov, & de Rijke, 2015): downloads can be seen as a special type of click; can we model different actions on an academic search interface to understand the different types of bias or to understand what makes the rich result presentation per item that is customary in academic search (consisting of title, authors, abstract snippet and possibly more) effective?

Code

To facilitate reproducibility of the results in this paper, we are sharing the code to run our experiments. See https://github.com/lxymichael/academic-paper-download-prediction.

Acknowledgments

This research was partially supported by Ahold Delhaize, the Bloomberg Research Grant program, the China Scholarship Council, the Criteo Faculty Research Award program, Elsevier, the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement nr 312827 (VOX-Pol), the Google Faculty Research Awards program, the Microsoft Research Ph.D. program, the Netherlands Institute for Sound and Vision, the Netherlands Organisation for Scientific Research (NWO) under project nrs CI-14-

25, 652.002.001, 612.001.551, 652.001.003, and Yandex. All content represents the opinion of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.

References

Aljukhadar, M., Senecal, S., & Daoust, C.-E. (2010). *Information overload and usage of recommendations. Proceedings of the ACM RecSys 2010 workshop on user-centric evaluation of recommender systems and their interfaces*. ACM.

Beierle, F., Aizawa, A., & Beel, J. (2017). Exploring choice overload in related-article recommendations in digital libraries. arXiv:1704.00393.

Berndt, D. J., & Clifford, J. (1994). *Using dynamic time warping to find patterns in time series. Proceedings of the 3rd international conference on knowledge discovery and data mining*. AAAI359–370.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research, 3*, 993–1022.

Borisov, A., Markov, I., de Rijke, M., & Serdyukov, P. (2016). *A context-aware time model for web search. Proceedings 39th international ACM SIGIR conference on research and development in information retrieval*. ACM205–214.

Catledge, L. D., & Pitkow, J. E. (1995). Characterizing browsing strategies in the world-wide web. *Computer Networks and ISDN systems, 27*(6), 1065–1073.

Chuklin, A., Markov, I., & de Rijke, M. (2015). *Click models for web search*. Morgan & Claypool Publishers.

Dali Betzalel, N., Shapira, B., & Rokach, L. (2015). *Please, not now!: A model for timing recommendations. Proceedings of the 9th ACM international conference on recommender systems*. ACM297–300.

Fiterau, M., Bhooshan, S., Fries, J., Bournhonesque, C., Hicks, J., Halilaj, E., Ré, C., & Delp, S. (2017). Shortfuse: Biomedical time series representations in the presence of structured information. arXiv:1705.04790.

Ghose, A., Ipeirotis, P. G., & Li, B. (2014). Examining the impact of ranking on consumer behavior and search engine revenue. *Management Science, 60*(7), 1632–1654.

Glorot, X., & Bengio, Y. (2010). *Understanding the difficulty of training deep feedforward neural networks. Proceedings of the 13th international conference on artificial intelligence and statistics*. JMLR249–256.

Gori, M., & Pucci, A. (2006). *Research paper recommender systems: A random-walk based approach. Proceedings of the 2006 IEEE/WIC/ACM international conference on web intelligence*. IEEE778–781.

Goswami, S. (2015). Analysing effects of information overload on decision quality in an online environment. *SAMVAD International Journal of Management, 9*, 65–69.

Hemminger, B. M., Lu, D., Vaughan, K., & Adams, S. J. (2007). Information seeking behavior of academic scientists. *Journal of the American Society for Information Science and Technology, 58*(14), 2205–2225.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9*(8), 1735–1780.

Ke, H.-R., Kwakkelaar, R., Tai, Y.-M., & Chen, L.-C. (2002). Exploring behavior of e-journal users in science and technology: Transaction log analysis of Elsevier's ScienceDirect OnSite in Taiwan. *Library & Information Science Research, 24*(3), 265–291.

Khabsa, M., Wu, Z., & Giles, C. L. (2016). *Towards better understanding of academic search. Proceedings of the 16th ACM/IEEE-CS on joint conference on digital libraries*. ACM111–114.

Kooti, F., Grbovic, M., Aiello, L. M., Bax, E., & Lerman, K. (2017). *iPhone's digital marketplace: Characterizing the big spenders. Proceedings of the tenth ACM international conference on web search and data mining*. ACM13–21.

Kooti, F., Lerman, K., Aiello, L. M., Grbovic, M., Djuric, N., & Radosavljevic, V. (2016). *Portrait of an online shopper: Understanding and predicting consumer behavior. Proceedings of the ninth ACM international conference on web search and data mining*. ACM205–214.

Landis, J. R., & Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics, 33*(1), 159–174.

Lee, M., Ha, T., Han, J., Rha, J.-Y., & Kwon, T. T. (2015). *Online footsteps to purchase: Exploring consumer behaviors on online shopping sites. WebSci '15 proceedings of the ACM web science conference*. ACM Article 15

Lei, Q., Yi, J., Vaculin, R., Wu, L., & Dhillon, I. S. (2017). Similarity preserving representation learning for time series analysis. arXiv:1702.03584.

Li, C.-Y. (2016). Why do online consumers experience information overload? an extension of communication theory. *Journal of Information Science, 43*(6), 835–851.

Li, H., Councill, I., Lee, W.-C., & Giles, C. L. (2006). *Citeseerx: an architecture and web service design for an academic document search engine. Proceedings of the 15th international conference on world wide web*. ACM883–884.

Li, X., & de Rijke, M. (2017). *Do topic shift and query reformulation patterns correlate in academic search? Proceedings of the 39th European conference on IR research*. Springer146–159.

Li, X., Schijvenaars, R., & de Rijke, M. (2017). *Investigating queries and search failures in academic search. Information processing & management53. Information processing & management* Elsevier666–683.

Li, Y., Yang, M., & Zhang, Z. M. (2013). *Scientific articles recommendation. Proceedings of the 22nd ACM international conference on information and knowledge management*. ACM1147–1156.

Mitra, A., & Awekar, A. (2017). *On low overlap among search results of academic search engines. Proceedings of the 26th international conference on world wide web companion*. ACM823–824.

Nicholas, D., Huntington, P., Jamali, H. R., Rowlands, I., Dobrowolski, T., & Tenopir, C. (2008). *Viewing and reading behaviour in a virtual environment: The full-text download and what can be read into it. Aslib proceedings*. Emerald Group Publishing Limited185–198 60

Nishioka, C., & Scherp, A. (2016). *Profiling vs. time vs. content: What does matter for top-k publication recommendation based on twitter profiles? Proceedings of the 16th ACM/IEEE-CS on joint conference on digital libraries*. IEEE171–180.

Niu, X., & Hemminger, B. M. (2012). A study of factors that affect the information-seeking behavior of academic scientists. *Journal of the American Society for Information Science and Technology, 63*(2), 336–353.

Niu, X., Hemminger, B. M., Lown, C., Adams, S., Brown, C., Level, A., et al. (2010). National study of information seeking behavior of academic researchers in the United States. *Journal of the American Society for Information Science and Technology, 61*(5), 869–890.

Ollagnier, A., Fournier, S., & Bellot, P. (2018). *BIBLME RecSys: Harnessing bibliometric measures for a scholarly paper recommender system. BIR 2018 workshop on bibliometric-enhanced information retrieval*.

Pontis, S., & Blandford, A. (2015). Understanding "influence:" An exploratory study of academics' processes of knowledge construction through iterative and interactive information seeking. *Journal of the Association for Information Science and Technology, 66*(8), 1576–1593.

Pontis, S., Blandford, A., Greifeneder, E., Attalla, H., & Neal, D. (2015). Keeping up to date: An academic researcher's information journey. *Journal of the American Society for Information Science and Technology, 68*(1), 22–35.

Porter, A., & Rafols, I. (2009). Is science becoming more interdisciplinary? measuring and mapping six research fields over time. *Scientometrics, 81*(3), 719–745.

Sinha, A., Shen, Z., Song, Y., Ma, H., Eide, D., Hsu, B.-j. P., & Wang, K. (2015). *An overview of Microsoft Academic Service (MAS) and applications. Proceedings of the 24th international conference on world wide web*. ACM243–246.

Srivastava, J., Cooley, R., Deshpande, M., & Tan, P.-N. (2000). Web usage mining: Discovery and applications of usage patterns from web data. *ACM SIGKDD Explorations Newsletter, 1*(2), 12–23.

Sun, J., Jiang, Y., Cheng, X., Du, W., Liu, Y., & Ma, J. (2018). A hybrid approach for article recommendation in research social networks. *Journal of Information Science, 44*(5), 696–711.

Tang, J. (2016). *AMiner: Toward understanding big scholar data. Proceedings of the ninth ACM international conference on web search and data mining*. ACM , 467–467.

Xiong, C., Power, R., & Callan, J. (2017). *Explicit semantic ranking for academic search via knowledge graph embedding. Proceedings of the 26th international conference on world wide web*. ACM1271–1279.

Yeo, J., Kim, S., Koh, E., Hwang, S.-w., & Lipka, N. (2017). *Predicting online purchase conversion for retargeting. Proceedings of the Tenth ACM international conference on web search and data mining*. ACM591–600.