Continuous Evaluation of Large-scale Information Access Systems: A Case for Living Labs

Frank Hopfgartner, Krisztian Balog, Andreas Lommatzsch, Liadh Kelly, Benjamin Kille, Anne Schuth, and Martha Larson

Abstract A/B testing is currently being increasingly adopted for the evaluation of commercial information access systems with a large user base since it provides the advantage of observing the efficiency and effectiveness of information access systems under real conditions. Unfortunately, unless university-based researchers closely collaborate with industry or develop their own infrastructure or user base, they cannot validate their ideas in live settings with real users. Without online testing opportunities open to the research communities, academic researchers are unable to employ online evaluation on a larger scale. This means that they do not get feedback for their ideas and cannot advance their research further. Businesses, on the other hand, miss the opportunity to have higher customer satisfaction due to improved systems. In addition, users miss the chance to benefit from an improved information access system. In this chapter, we introduce two evaluation initiatives at CLEF, NewsREEL and Living Labs for IR (LL4IR), that aim to address this growing "eval-

Frank Hopfgartner
University of Sheffield, Sheffield, UK, e-mail: f.hopfgartner@sheffield.ac.uk

Krisztian Balog
University of Stavanger, Stavanger, Norway, e-mail: krisztian.balog@uis.no

Andreas Lommatzsch
Technische Universität Berlin, Berlin, Germany, e-mail: andreas.lommatzsch@dai-labor.de

Liadh Kelly
Maynooth University, Maynooth, Ireland, e-mail: liadh.kelly@mu.ie

Benjamin Kille
Technische Universität Berlin, Berlin, Germany, e-mail: benjamin.kille@dai-labor.de

Anne Schuth
University of Amsterdam, The Netherlands, e-mail: anne.schuth@gmail.com
Now at De Persgroep AI, The Netherlands

Martha Larson
Radboud University, Nijmegen, The Netherlands, e-mail: m.larson@cs.ru.nl

uation gap" between academia and industry. We explain the challenges and discuss the experiences organizing theses living labs.

1 Introduction

As evident from the other chapters of this book, significant efforts have been invested in establishing metrics, frameworks, and datasets to guarantee a thorough and transparent evaluation of novel approaches to retrieve or recommend documents and items. For many years, campaigns such as CLEF, TREC, NTCIR, and FIRE have played leading roles in promoting research in the field of information retrieval. The release of datasets, standardized evaluation metrics and evaluation procedures, following the established Cranfield evaluation paradigm, has contributed to innovative retrieval approach development in domains such as newswire articles, blogs, microblogs, and biomedical documents to name but a few. In the field of recommender systems research, a similar coordinated evaluation procedure with standardized datasets and evaluation criteria has been established thanks to the release of the Netflix dataset and the associated challenge, as well as the release of the MovieLens datasets. In both cases, it is safe to claim that the release of test collections was of great benefit for the research community since it spared researchers not only from the tedious task of creating their own datasets, but also allowed them to easily compare their results with state-of-the-art algorithms. However, as Voorhees and Harman (2005) point out, the use of standardized datasets also comes with certain drawbacks. In many research papers, datasets are used to fine-tune computational models or algorithms, resulting in improved performance, e.g., measured based on precision, recall, or using other popular metrics. This is a direct consequence of the ability to compare performance against state-of-the-art approaches and the desire to beat those baselines.

This limitation is well understood by commercial providers of information access systems who rely increasingly on user-centric evaluation of their systems to achieve optimal performance (Kohavi, 2015). The large number of users of their systems implicitly allows for evaluation of the efficiency and effectiveness of algorithms under real conditions as they engage with the systems. This has resulted in this user-centric evaluation paradigm evolving into the de-facto evaluation standard employed in commercial settings. Evaluation of this nature is referred to as *online evaluation* since it is employed using instances of online information access systems, or as *A/B testing* since it allows for the comparison of different variants of the system. Unfortunately, non-commercial, especially university-based, researchers are now struggling to evaluate their own approaches using this resource-demanding evaluation standard. This was also pointed out by Hawking (2015) who compared the affiliation of authors' of research papers presented at SIGIR'98 and SIGIR'15, respectively. He argued that the observed increase from 15% of industrial research papers published in 1998 compared to 41% published in 2015 is a direct consequence of

the increased need to evaluate research methods using large-scale datasets or user studies.

Addressing the lack of access to data, Hanbury et al (2015) argue for the implementation of evaluation services that store data on a central server and allow researchers access to both data and information technology infrastructure. They refer to this method as Evaluation-as-a-Service (EaaS). While this approach has the potential to alleviate the growing evaluation gap to some extent, it does not address the issue of having limited access to real users who can be test subjects for researchers' algorithms and ideas. To address this, the application of a living lab that grants researchers access to real users who follow their own information seeking tasks in a natural and thus realistic contextual setting has been proposed (Kamps et al, 2009; Kelly et al, 2009). For user-centric research on information access systems, realistic context is essential since it is a requirement for a fair and unbiased evaluation. In this chapter, we present the two living labs initiatives that have been introduced within the domains of recommender systems and information retrieval (IR).

The CLEF NewsREEL challenge is a campaign-style evaluation lab allowing participants to evaluate and optimize news recommender algorithms. The goal is to create an algorithm that is able to generate news items that users would click on, respecting a strict time constraint. The lab challenged participants to compete in either a living lab or perform an evaluation that replays recorded streams. By participating in this living lab, participants are given the opportunity to develop news recommendation algorithms and have them tested by potentially millions of users of a live system over a longer period of time.

The Living Labs for Information Retrieval (LL4IR) CLEF lab is a benchmarking platform for researchers to evaluate their retrieval systems in a live setting. The lab acts as a proxy between commercial organizations (live environments) and lab participants (experimental systems), facilitates data exchange, and makes comparison between the participating systems possible. The LL4IR lab focused on two use cases: product search (on an e-commerce site) and web search (through a commercial web search engine).

After surveying state-of-the-art in the area of online evaluation in Section 2, we present the NewsREEL (Section 3) and LL4IR (Section 4) use cases as leading examples of living labs evaluation. In Section 5 we highlight similarities and differences between the two approaches, and conclude with a discussion on the opportunities and challenges that such online evaluation campaigns offer.

2 Related Work

Information access systems have been evaluated in four major ways (Gunawardana and Shani, 2009): offline with static test collections, with small-scale user studies or user simulations, and in online evaluation environments. Tradition has favored offline evaluation to ensure reproducibility. At the same time, such evaluation may not accurately reflect user satisfaction (Teevan et al, 2007; Turpin and Scholar, 2006;

Wilkins et al, 2008). Moreover, it leaves one of the most important factors of any information retrieval or recommender system out of the loop: the user. It is the user's information need that needs to be satisfied and it is the user's personal interests that need to be considered when providing personalized access to information. This is one of the major reasons for performing *online* evaluation: evaluation with users in the loop.

The need for more realistic evaluation, involving real users, has been reiterated at several workshops (Kamps et al, 2009; Allan et al, 2012; Balog et al, 2014a). To address this, living labs have emerged as a way for researchers to be able to perform *in situ* evaluation. The main idea behind living labs is that an existing information access service serves as the experimentation platform. By replacing components of this information access platform, researchers have the opportunity to perform evaluation using interactions with real, unsuspecting users of this information access system. Major information access online evaluations and A/B testing are instances of living labs. However, this type of evaluation has only recently become available to the broader research community.

2.1 Living Labs Shared Challenges

The notion of using living labs for shared challenges in the information access space has been proposed in recent years (Azzopardi and Balog, 2011; Kelly et al, 2012). In particular, Azzopardi and Balog (2011) present details on an approach to move from a traditional IR evaluation setting to a living labs setting. The first implementation of a living lab was the NewsREEL challenge that was first organized as part of a workshop co-located with ACM RecSys (Tavakolifard et al, 2013). Later, it was operated as part of CLEF. NewsREEL allowed participants to evaluate and optimize news recommendation algorithms. The goal was to create an algorithm for news recommendation that is able to generate news items that users would click on, respecting a strict time constraint for generating and serving those recommendations. By participating in NewsREEL, researchers who develop stream-based recommendation algorithms could have these benchmarked by actual users of a live system over a longer period of time (Hopfgartner et al, 2015a). In the context of information retrieval, Balog et al (2014b) proposed a practical way of operationalizing the living lab idea by limiting evaluation to head queries, a setup that was subsequently adopted by the CLEF LL4IR lab (cf. Sect. 4.1). The same idea was also employed at the TREC 2016 and 2017 OpenSearch track, where the use case is scientific literature search (Jagerman et al, 2018). Kelly et al (2012) presented an alternative living labs setting as a solution to the evaluation of personal search.

2.2 Online Testing

A/B Testing compares two systems by showing system A to one group of users and system B to a disjoint group (Kohavi, 2015). The difference between the systems is inferred from observed user behavior. This includes, among other things, click-through rate (CTR) (Joachims et al, 2007), dwell time (Yilmaz et al, 2014), satisfied clicks (Kim et al, 2014), abandonment (Li et al, 2009), query reformulation (Hassan et al, 2013), and mouse movement (Wang et al, 2010; Diaz et al, 2013). NewsREEL, for example, employed the click-through rate as its primary evaluation criterion.

An alternative to A/B testing is to perform interleaved comparisons, which are shown to be more sensitive (Schuth et al, 2015c; Chapelle et al, 2012). This means that far fewer query impressions are required to make informed decisions on which ranker is better. Many interleaving approaches have been proposed over the past few years, see, e.g., (Joachims, 2003; Radlinski et al, 2008; Hofmann et al, 2011; Radlinski and Craswell, 2013; Schuth et al, 2014, 2015b). By far the most frequently used interleaving algorithm to date is Team Draft Interleaving (TDI) (Radlinski et al, 2008) which is also what is used in the CLEF LL4IR lab. Given a user query q, TDI produces an interleaved result list as follows. The algorithm takes as input two rankings. One ranking from the participant $r' = (a_1, a_2, \ldots)$ and one from the production system $r = (b_1, b_2, \ldots)$. The goal is to produce a combined, interleaved ranking $L = (a_1, b_2, \ldots)$. This is done similarly to how sports teams may be constructed in a friendly sports match. The two team captains take turns picking players. They can pick available documents (players) from the top of the rankings r' and r, these top ranked documents are deemed to be the best documents. Documents can only be picked once (even if they are listed in both r and r'). And the order in which the documents are picked determines ranking L. In each round, the team captains flip a coin to determine who goes first. The algorithm remembers which team each document belongs to. If a document receives a click from a user, credit is assigned to the team the document belongs to. The team (participant or production system) with most credit wins the interleaved comparison. This process is repeated for each query. For more details see the original paper describing TDI by Radlinski et al (2008) and a large-scale comparison of interleaving methods by Chapelle et al (2012).

3 News Recommendation Evaluation Lab (NewsREEL)

The first information access living lab that is introduced in this chapter focuses on the domain of news recommendation. Recommender systems pro-actively suggest information to users based on their preferences. The first recommender systems entered the realm of online content distribution in the 2000s. Unfortunately though, after a decade of research, a gap emerged between academia and industry. Academia focused on experimenting with fixed datasets often neglecting practical aspects of recommender systems. Industry, on the other hand, implemented A/B testing procedures. As discussed in Section 2.2, this procedure partitions users into groups,

exposes them to variations of the system, and monitors differences in performance. While academia achieved repeatability of experiments, industry observes the actual reactions of users. NewsREEL, short for News Recommendation Evaluation Lab, was a campaign style evaluation task designed to bridge this gap.[1] It was first organized in conjunction with the ACM RecSys 2013 Workshop on News Recommender Systems (Tavakolifard et al, 2013) and then joined CLEF as campaign-style lab between 2014 and 2017. The four CLEF editions observed a total of 230 registrations. NewsREEL afforded participants the opportunity to engage in both offline and online evaluations. On the one hand, participants had access to a large-scale stream of recorded events, which could be used for offline comparison of different algorithms. On the other hand, participants gained access to a commercial news recommender system which delivered suggestions for a set of publishers in real-time. This provided participants with access to authentic live recommender system conditions. Developing recommender services in this environment represents a challenging task. Challenges included overcoming issues of availability, responsiveness, and scalability beside algorithmic design and optimization. In particular, the environment is subject to change. Publishers push new articles as events happen. Readers' interests shift over time. Hence, models have to be updated.

In the remainder of this section, we first describe the news recommendation problem addressed by NewsREEL and introduce the online and offline tasks, NewsREEL Live and NewsREEL Replay (Section 3.1). While the online task requires participants to provide recommendations to real users in real-time, the offline task can be run on standalone hardware without online access and the necessity to fulfill specific time constraints. In addition, the offline task simplifies the debugging and the simulation of streams. Algorithms shown to be working offline can then be evaluated in the NewsREEL Live task without any changes. Section 3.2 describes the NewsREEL evaluation architecture. We discuss participation in the online challenge in Section 3.3 and the offline challenge in Section 3.4. Section 3.5 provides a discussion on NewsREEL.

3.1 NewsREEL Use Case

As previously mentioned, CLEF NewsREEL implemented a shared challenge in the news recommendation space. It consisted of two tasks that were based on the use case of providing a list of news articles relevant to a given new article that a reader might be interested in. As depicted in Figure 1, these news article recommendations are often displayed at the bottom or the side of the article. Determining what articles to suggest to readers is challenging from a technical point of view. First of all, recommendations have to be displayed to readers in real-time. Moreover, publishers have relatively limited information about readers and their interests. Supply and demand of information are continuously subject to change. Besides, publishers

[1] See `http://newsreelchallenge.org/` for details.

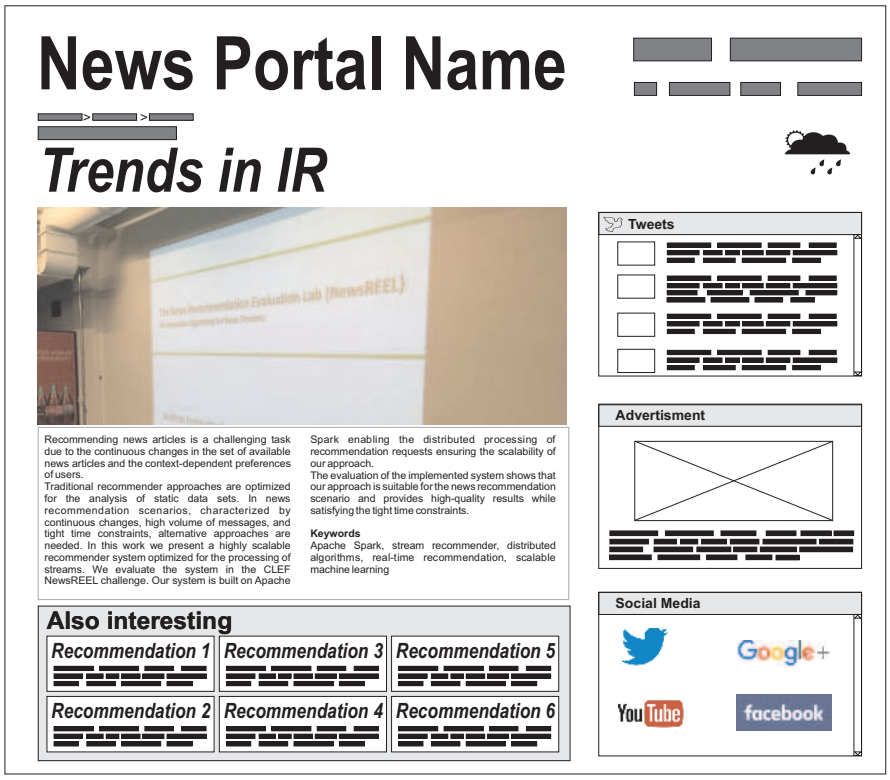

Fig. 1 Exemplary illustration of the way news recommendations are displayed to readers in the NewsREEL scenario.

constantly add new articles and readers may loose interest in events or move on to different topics. News recommender systems have to adapt to these dynamics. The two tasks are outlined in detail in the remainder of this section. For a more detailed description of the NewsREEL use case, we refer the reader to (Hopfgartner et al, 2015a).

Online Evaluation of News Recommendation Algorithms

The first NewsREEL task implemented a living lab style shared challenge. This living labs evaluation challenge is described in detail in (Hopfgartner et al, 2014). Researchers gained access to resources of the online information service provider plista[2] such that they could conduct A/B testing for a selection of recommendation techniques. Plista offers recommendation services and targeted advertisements for online publishers. As users request articles from publishers' web portals, plista provides a list of additional suggested articles. Plista forwards a random subset of these

[2] http://plista.com/

request to NewsREEL's participants via the Open Recommendation Platform (ORP) (see Brodt and Hopfgartner (2014)). In addition, participants received information about the overall activity on the publishers' platform in the form of reads, clicks on suggestions, as well as new or updated articles. Participants needed to respond to requests within 100ms.

Offline Evaluation of News Recommendation Algorithms

The second task addressed the academic perspective of focusing on reproducibility of results. Tools to replay the event stream allowed participants to compare algorithms and parameter configurations in identical conditions. In addition, participants could determine time and space complexity of their algorithms. Kille et al (2015) describe the offline task in greater detail.

We have released multiple large datasets comprising interactions between users and articles on various publishers sites. The datasets' characteristics are described in detail in (Kille et al, 2013). The news portals publish mostly German articles. Consequently 80 per cent of readers reside in the German-speaking area of Central Europe (Germany, Austria, and Switzerland). Figure 2 illustrates the geographical spread of user activity. Moreover, we have released a toolkit called idomaar (Scriminaci et al, 2016) that allowed participants to "replay" the dataset.

3.2 NewsREEL Architecture

NewsREEL has been designed with reusability in mind. Both tasks assessed the quality of recommendation strategies for news. In the online living labs task implicit feedback was received from users of the live publishers sites. The offline task estimated relative quality on a recorded stream of event messages. The tasks shared a common interface for recommendation algorithms. Thus, participants could deploy their algorithms in both tasks without additional costs. In the online task, the ORP handled communication and monitoring of feedback. In the offline task, a replaying service took the recorded streams as input, issued requests to the algorithms being evaluated, and kept track of the results. Figure 3 depicts the NewsREEL architecture. In both settings, requests emerged, were forwarded to a recommender, suggestions were delivered, and their performance was assessed. In the offline task, the contest server delivered a summary of the response times. This lets participants judge whether the algorithm is suited for online deployment. In the online task, ORP ignored recommendations arriving outside the defined response time limit. Thus, the more algorithms exceeded this threshold, the more the click-through rate decreased. In both settings, communication was based on HTTP. Data are exchanged in JSON format. Interfacing with publishers and providing large-scale data collections, NewsREEL represented a unique opportunity for academic researchers to experience a setting close to the industrial reality.

3.3 NewsREEL Recommender Algorithms

In the NewsREEL challenge, participants evaluated a wide spectrum of recommender approaches. In this section, we briefly summarize trialled methods and discuss their relation to the living labs environment. A more detailed overview of the strengths and limitations of these methods is currently under preparation.

The Algorithms Evaluation in the Online Task

Big Data Frameworks: Rapidly changing user preferences and strict requirements with respect to scalability and response time represented a major challenge for NewsREEL's participants. Several authors used big data frameworks to fulfill these requirements. Verbitskiy et al (2015) developed a most-popular recommender using the AKKA framework benefiting from concurrent message passing. They registered a high click-through rate while simultaneously ensuring fast responses.

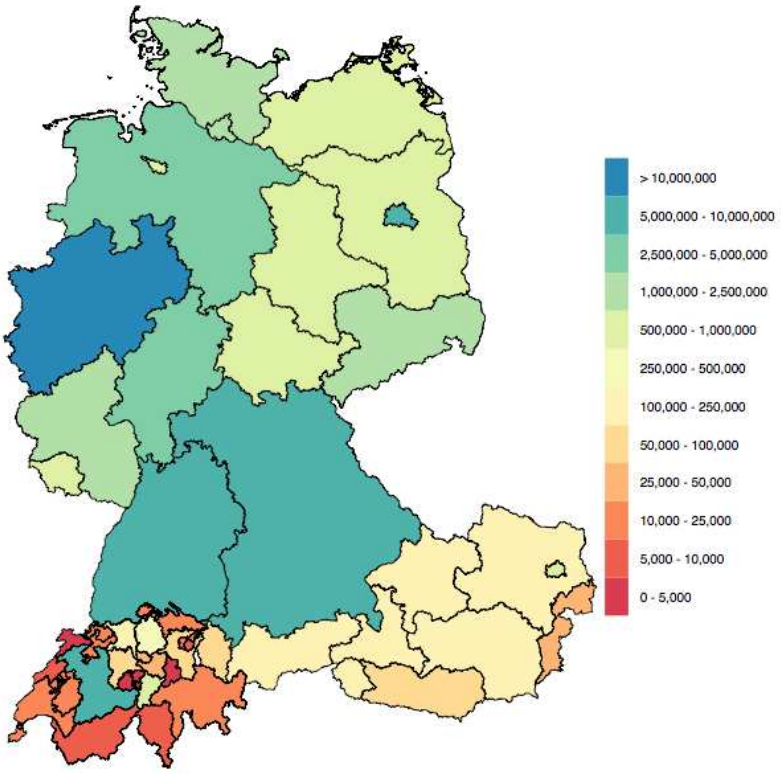

Fig. 2 Areas in Germany, Austria, and Switzerland from where requests for articles were triggered. The scale indicates the number of requests during one month.

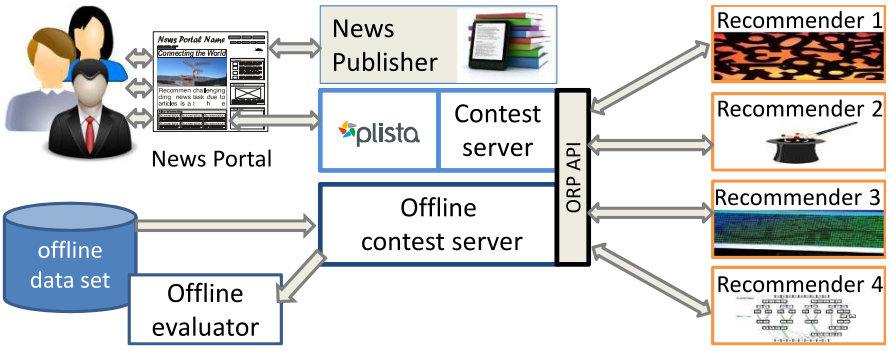

Fig. 3 The NewsREEL challenge architecture.

Ciobanu and Lommatzsch (2016) developed a stream-based news recommender using APACHE FLINK. They performed well even though the systems suffered from breaking streams in the long-term evaluation.

Several authors (Lommatzsch et al (2016); Domann et al (2016); Beck et al (2017)) have used APACHE SPARK and APACHE MAHOUT. The combinations facilitate periodically building new micro-batches to update the models. All these approaches outperformed the baseline while ensuring high scalability.

Graph- and Rule-based approaches: Bons et al (2017) developed a graph-based recommender algorithm. The graph consisted of nodes representing the items and directed edges describing the frequency and sequence in which the two connected news items were read. Recommendation requests were answered by computing the strongest item sequence containing the itemID given in the recommendation request. The graph was managed in a Neo4j graph database. Recommendations were computed based on a database query. If the itemID in the recommendation request did not exist in the graph or the node was not yet connected with the graph, the most recently created news items were returned. The evaluation of the strategy showed that the implemented graph-based recommender reached a high click-through rate in the Living Labs scenario. The implementation worked efficiently, ensuring that the time-constraints with respect to response time were reliably fulfilled.

Golian and Kuchar (2017) analyzed click patterns in time series from NewsREEL 2016. They showed that a limited set of news items attract a majority of clicks, and that they continue to dominate for longer times than expected. They conducted a series of experiments in the context of online news recommender system evaluation. The authors report that content-based methods achieve considerably lesser click-through rates than popularity-based methods.

Ludmann (2017) focused on managing streams. His system relied on *Odysseus*, a data stream management system. He defined a set of queries which took parts of the data stream and determined the most popular articles. The selection entailed the length of the data stream segment as essential parameter. They presented observations on NewsREEL Live with a variety of parameter configurations. Results

suggest that considering successful recommendations improves the click-through rates.

Recommender Ensembles: The continuous changes in the data stream motivated several participants to implement an ensemble recommender. Beck et al (2017) used an ensemble of a user-based collaborative (CF) and a most popular ("unpersonalized") recommender. The CF-based recommender provided personalized recommendation for users with session-profiles. The most popular recommender provided recommendations for new users (overcoming the cold-start problem). More complex ensembles combining different content-based and CF-based recommender algorithms are presented in (Lommatzsch and Albayrak, 2015). The developed system estimated the performance of the different recommender algorithms in different contexts (defined by on time and type of recommendation requests). The system learned which algorithms performed best for each context—new requests were delegated to the most promising algorithm. The ensemble approach outperformed all teams using only a single algorithm.

Gebremeskel and de Vries (2015) explored the utility of geographic information. They hypothesized that visitors have special interest in news stories about their local community. They implemented a recommender which leveraged geographic data when matching visitors and news articles.

Corsini and Larson (2016) discussed how images affect users' response to recommendations. They argued that selecting promising images increases the likelihood of clicks. They introduced an image processing pipeline. The pipeline detects faces and image salience. A binary classifier subsequently decided whether an image is interesting or not. The authors evaluated the approach offline and online. They report improvements in the offline case. Further work is necessary to achieve reliable online evaluation results.

Liang et al (2017) discussed how contextual bandits can be used to compute recommendations. The authors defined a list of recommendation models considering recency, categories, and reading sequences among other factors. Their contextual bandit approach seeks to determine a strategy mapping models to contexts in order to maximize the expected rewards. They applied their contextual bandit both in NewsREEL Live and NewsREEL Replay. They report that performances vary depending on the domain under consideration.

The Algorithms in the Offline Task

The offline evaluation task has attracted several teams. The teams mainly focused on testing more sophisticated recommendation approaches (e.g. deep neural networks (Kumar et al, 2017)), studied efficient optimization of parameter configuration (e.g. finding similarity metrics for Collaborative Filtering (Beck et al, 2017)), and explored the technical complexity of algorithms. One advantage of the offline task is that it does not require a permanent Internet connection and does not put additional burden on the participants to produce recommendations within a pre-defined tight time window. This ensured a low barrier to participate in the offline task and al-

lowed participants to test new ideas and algorithms. In the remainder of this section, we discuss these, and other advantages, further.

Ease of Use: Applying innovative ideas in a recommendation scenario typically requires extended testing and debugging. Before deploying algorithms, they are checked for their suitability to the scenario. The offline evaluation provides a well-suited environment for testing, debugging, and optimizing recommenders. Participants could simulate the stream on local hardware and study the strengths and weaknesses of new algorithms. The offline tests can control the load (by defining the number of concurrent messages sent by the offline simulation environment) and debug the functionality of the implemented solution. Participants typically tested algorithms first offline before moving to the online task. Innovative recommender approaches, for instance, based on Contextual Bandits or Deep Neural Networks have been evaluated offline.

Parameter Optimization: Finding the optimal parameter configurations complements testing new approaches in offline evaluation. Optimization requires sufficiently large data streams to obtain robust results. Parallelization can be used to speed up optimization. The offline task supports parallelization. Participants can simulate the stream on multiple machines to arrive more quickly at the optimal configuration. In addition, the simulated stream can be replayed faster in order to accelerate the optimization process. The offline stream simulation ensures reproducible evaluation results as well as the comparability of the results obtained in different evaluation runs. This aspect of the offline task has been extensively used by several teams (e.g. by Beck et al (2017)).

Technical Aspects: Tight time constraints, continuous changes of readers and articles, and the varying frequency with which messages emerge are difficult to simulate offline. The contest server allows participants to vary the number of concurrently sent messages. This facilitates finding bottlenecks which would cause errors in the online evaluation. Participants look at the distribution of response times to avoid such errors. This is particularly important for ensemble-based methods integration of multiple individual algorithms with varying complexities.

3.4 NewsREEL Evaluation

In order to evaluate the performance of different algorithms, NewsREEL followed the EaaS paradigm discussed by Hopfgartner et al (2018).

In the four iterations of NewsREEL, most approaches achieved results superior to the baseline and still hold the potential for further optimization. The offline evaluation facilitates fine-grained analysis and parameter optimization for new algorithms. Thereby, it enables participants to verify their ideas before deploying them online. The majority of participants used this opportunity. Figure 4 shows the distribution of click-through rate and standard deviation of all teams participating in NewsREEL 2017. In addition, the legend indicates for how many hours the corresponding algorithm had been active. A multitude of facets give rise to different perspectives on

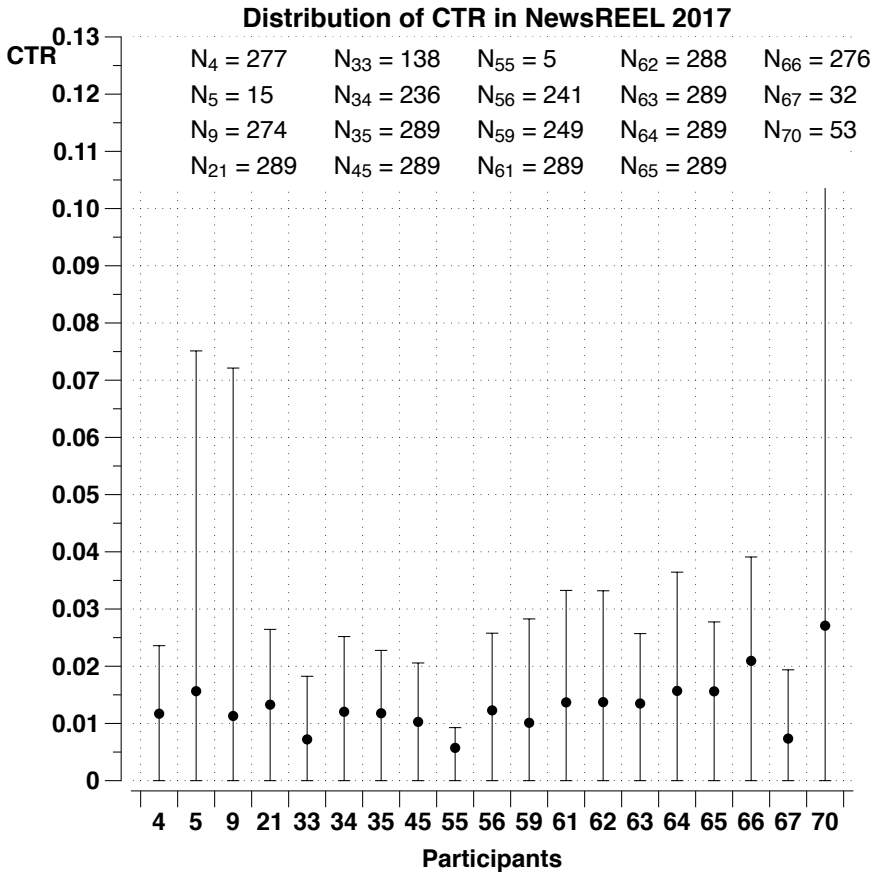

Fig. 4 Distribution of click-through rate in NewsREEL 2017

the quality desired of recommendations. First, we may ask who is to benefit from a recommender system? Readers, or users in general, avoid looking for information themselves. Publishers, on the other hand, retain readers and increase the chance for future visits. Second, we consider how to quantify utility. Recommendation has been modeled in various forms, including preference estimation, binary classification, and ranking problem. The click-through rate has been established as primary utility estimate in the online task. It represents the proportion of suggestions that readers clicked. Publishers would prefer to estimate their utility more directly, for instance, in terms of dwell time or the likelihood that readers will return. Both have proven difficult to compute with data available to NewsREEL. Sessions tend to include few reads which is why readers returning with the same session key are an uncommon phenomenon. In addition, computing the dwell time requires the next read event. Moreover, a considerable subset of readers disallows session keys to be stored on their machines. As a result, we cannot distinguish them from one another

rendering dwell time estimation impossible. Third, we have to take user experience into account. Waiting for recommendation entails costs similar to irrelevant recommendations. Readers are unlikely to wait for suggestions. Therefore, we have to consider additional aspects of utility such as availability, responsiveness, and scalability. In the online task, we monitor error events. These occur in cases when recommendation services fail to deliver in time or deliver invalid suggestions. In the offline task, the contest server computes the distribution of response times. This information enables us to compare algorithms in an additional dimension. i.e., it allows us to focus on both effectiveness and technical constraints that could not be evaluated in an online setting.

3.5 NewsREEL Discussion

The variety of methods used to address NewsREEL's tasks indicate a large number of connected research challenges for the future. While a more detailed analysis of these challenges is currently under preparation, we conclude this section by briefly highlighting the main successes and challenges of our initiative:

Successes:

- Being the first implementation of a living lab for the evaluation of information access systems, NewsREEL pioneered a new level of collaboration that enabled university-based researchers to gain access to a company's IT infrastructure and user base. We argue that this model of cooperation has the potential to narrow the growing gap between academic and commercial research in the field of information access.
- All of the four main information access evaluation campaigns (i.e., TREC, NTCIR, CLEF, and FIRE) have used news corpora in the past to advance research on challenges including ad-hoc retrieval, known item search, multilingual retrieval, and related retrieval tasks. NewsREEL contributes to this tradition by allowing further research on challenges such as real-time and stream processing, click optimization, and user profiling.
- NewsREEL has been used by practitioners, teachers at universities, and researchers. A survey amongst participants (Lommatzsch et al, 2017) has revealed that one of the main motivations for them to participate was to acquire new skills that are currently in high demand in industry. At the same time, NewsREEL has also been successfully embedded in teaching since students experienced factors associated with working in industry (Hopfgartner et al, 2016).

Challenges:

- NewsREEL differs from the more traditional evaluation campaigns as participants had to ensure a high click-through rate under tight time constraints. We understand that these requirements were new to most researchers and that these

different entry requirements might hold them back from participating. We addressed this by offering tutorials (e.g., at ECIR'15 (Hopfgartner and Brodt, 2015) and ACM RecSys'15 (Hopfgartner et al, 2015b)), and by providing detailed instructions on how to get started on the NewsREEL website.

- In the online task, participants had to deal with fulfilling two goals at once. On the one hand, they had to optimize the click-through rate. On the other hand, they had to respond in a timely manner with valid items to guarantee a convenient user experience. The latter goal in particular, has caused major efforts as researchers tend to focus on algorithmic details rather than maintenance and scalability. Time constraints also had an effect on the computational complexity of algorithms. In addition, the real-time requirements render it difficult to debug the implementation. Although these are real issues and requirements that operators of online recommender systems face, we addressed this by introducing the offline task which allowed participants to implement and benchmark their algorithms and then deploy them to the online task.
- In the offline task, participants had to cope with the scale of the recorded data stream. Millions of events amount to gigabytes of data. Conducting experiments with the data takes a long time, in particular on personal computers. In order to address this, we released the benchmarking framework Idomaar that makes use of Big Data solutions such as Apache Kafka and Apache Flume. Idomaar can be deployed to Hadoop-based infrastructures that are able to cope with larger data streams (Scriminaci et al, 2016).
- In addition, the dynamic environment of news mandates continuous model updates. Seasonal trends, shifts in readers' interests, differences between working days and weekends or holidays produce varying behaviors of actors inside the news ecosphere. Breaking news events add another source for variation. This is in particular challenging for recommendation techniques that rely on exploiting users' prior interaction with news items (e.g., (Hopfgartner and Jose, 2014)).
- The online component of NewsREEL causes additional challenges that need to be considered in order to guarantee a fair and unbiased evaluation. For example, some participants might suffer from network latency, especially if they were located far from plista's data centre in Germany. We addressed this limitation by offering virtual machines for participants in plista's data centre that they could use to deploy their algorithms. This solution is in line with the idea of EaaS as described by (Hopfgartner et al, 2018).
- Receiving greatly varying numbers of requests can cause additional issues. For example, one participant may deliver a relatively high click-through rate with few requests, whereas another participants scores more clicks in total with more requests. Comparing these participants is difficult as the relatively high click-through rate could be due to chance.

4 Living Labs for Information Retrieval (LL4IR)

The main objective of the Living Labs for IR Evaluation (LL4IR) CLEF Lab was to provide a benchmarking platform for researchers to evaluate their ranking systems in a live setting. The lab acted as a proxy between commercial organizations (live environments) and lab participants (experimental systems), facilitated data exchange, and made comparisons between the participating systems. The lab focused on two use cases and one specific notion of what a living lab is. Use cases considered here were: product search (on an e-commerce site) and web search (through a commercial web search engine).

The LL4IR CLEF Lab contributed to the understanding of online evaluation as well as an understanding of the generalization of retrieval techniques across different use cases. Most importantly, it promoted IR evaluation that is more realistic, by allowing researchers to have access to historical search and usage data and by enabling them to validate their ideas in live settings with real users. This initiative was a first of its kind for IR.

This section reports on the results obtained during the official CLEF evaluation round that took place between May 1 and May 15, 2015. The positive feedback and growing interest from participants motivated us to organize a subsequent second unofficial evaluation round.

In the next section we describe the LL4IR API architecture and evaluation methodology. We then describe each of the two use cases in turn in Sections 4.2 and 4.3, and provide details and analysis of the submissions received. In Section 4.4, we conclude with a discussion on LL4IR.

4.1 LL4IR Architecture

For the LL4IR CLEF Lab, evaluation was conducted primarily through an API. We first describe the workings of this API, followed by the evaluation setup divided into training and test phases. We then describe how we computed evaluation metrics using interleaved comparisons. Finally, we describe how we aggregated interleaving outcomes.

4.1.1 LL4IR API

For each of the use cases, described in Sections 4.2 and 4.3, challenge participants took part in a live evaluation process. For this they used a set of frequent queries as training queries and a separate set of frequent queries as test queries. Candidate documents were provided for each query and historical information associated with the queries. When participants produced their rankings for each query, they uploaded these to the commercial provider use case through the provided LL4IR API. The commercial provider then interleaved a given participant's ranked list with their

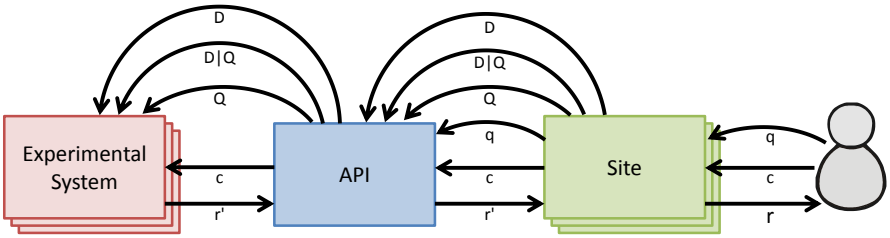

Fig. 5 Schematic representation of interaction with the LL4IR API, taken from (Balog et al, 2014b).

own ranking, and presented the user with the interleaved result list. Participants took turns in having their ranked list interleaved with the commercial providers ranked list. This process of interleaving a single experimental system with the production system at a time was orchestrated by the LL4IR API, such that each participant gets about the same number of impressions. The actions performed by the commercial providers' system users were then made available to the challenge participant (whose ranking had been shown) through the API; i.e., the interleaved ranking, resulting clicks, and (aggregated) interleaving outcomes.

Figure 5 shows the Living Labs architecture and how the participant interacted with the use cases through the LL4IR provided API. As can be seen, frequent queries (Q) with candidate documents for each query ($D|Q$) are sent from a site through the API to the experimental systems of participants. These systems upload their rankings (r') for each query to the API. When a user of the site issues one of these frequent queries (q), then the site requests a ranking (r') from the API and presents it interleaved with r to the users. Any interactions (c) of the user with this ranking are sent back to the API. Experimental systems can then obtain these interactions (c) from the API and update their ranking (r') if they wish. We provided participants with example code and guidelines to ease the adaptation to our setup.[3] Our evaluation methodology, including reasons for focusing on frequent queries, is described in more detail in (Balog et al, 2014b).

4.1.2 Training Phase

During the training phase, participants were free to update their rankings using feedback information. This feedback information was made available to them as soon as it arrived at the API. Their rankings could be updated at any time and as often as desired. Both click feedback and aggregated outcomes were made available directly and were updated constantly.

[3] `http://doc.living-labs.net/en/latest/guide-participant.html`

4.1.3 Test Phase

In the test phase, challenge participants received another set of frequent queries as test queries. Again, the associated historical click information as well as candidate results for these queries were made available. After downloading the test queries, participants could only upload their rankings until the test phase started or only once after it started. These rankings were then treated in the same way as training queries. That is, they were interleaved with the commercial providers' rankings for several weeks. As for the training phase, in the test phase each challenge participant was given an approximately equal numbers of impressions. A major difference is that for the test queries, the click feedback is not made available. Aggregated outcomes were provided only after the test phase had ended.

4.1.4 Evaluation Metric

The overall evaluation of challenge participants was based on the final system performance, and additionally on how the systems performed at each query issue. The primary metric used was aggregated interleaving outcomes, and in particular the fractions of winning system comparisons. See Section 2.2 for details on interleaving comparisons. There are two reasons for using interleaved comparisons. Firstly, interleaved comparisons ensure that at least half the ranking shown to users comes from the production system. This reduces the risk of showing bad rankings to users. Secondly, interleaved comparisons were shown to be two orders of magnitude more sensitive than other ways of performing online evaluation such as A/B testing (Schuth et al, 2015c; Chapelle et al, 2012). As mentioned in Section 2.2, this means that far fewer query impressions are required to make informed decisions on which ranker gives better performance.

Aggregated Outcomes

LL4IR reported the following aggregated interleaving metrics, where *Outcome* served as the primary metric for comparing participants rankings. These aggregations were constantly updated for training queries. For the test phase they were only computed after the phase had finished.

#Wins is defined as the number of wins of the participant against the production system, where a *win* is defined as the experimental system having more clicks on results assigned to it by TDI than clicks on results assigned to the production system;

#Losses is defined as the number of losses against the production system;

#Ties is defined as the number of ties with the production system;

#Impressions is the total number of times when rankings (for any of the test queries) from the participant have been displayed to users of the production system; and

Outcome is defined as the fraction of wins, so *#Wins*/ (*#Wins*+*#Losses*).

An *Outcome* value below the *expected outcome* (typically 0.5) means that the participant system performed worse than the production system (i.e., overall it had more losses than wins). Significance of outcomes was tested using a two-sided binomial test which used the expected outcome and reported p-values.

Note that using these metrics, we are in theory only able to say something about the relationship between the participant's system and the production system. However, Radlinski et al (2008) show experimentally that it is not unreasonable to assume transitivity. This allows us to also draw conclusions about how systems compare to each other. Ideally, instead of interleaving, we would have used multileaved comparison methods (Schuth et al, 2014, 2015b) which would directly give a ranking over rankers by comparing them all at once for each query.

4.2 LL4IR Use Case: Product Search

4.2.1 Task and Data

The *product search* use case is provided by REGIO Játék (REGIO Toy in English), the largest (offline) toy retailer in Hungary with currently over 30 stores. Their webshop[4] is among the top 5 in Hungary. The company is working on strengthening their online presence; improving the quality of product search in their online store is directed towards this larger goal. An excerpt from the search result page is shown in Figure 6.

As described in Section 4.1, we distinguished training and test phases. Queries are sampled from the set of frequent queries; these queries are very short (1.18 terms on average) and have a stable search volume. For each query, a set of candidate products (approximately 50 products per query) and historical click information (click-through rate) was made available. For each product a structured representation was supplied (see below). The task then was to rank the provided candidate set.

4.2.2 Product Descriptions

For each product a fielded document representation was provided, containing the attributes shown in Table 1. The amount of text available for individual products is limited (and is in Hungarian), but there are structural and semantic annotations, including:

- Organization of products into a two-level deep topical categorization system;
- Toy characters associated with the product (Barbie, Spiderman, Hello Kitty, etc.);
- Brand (Beados, LEGO, Simba, etc.);

[4] `http://www.regiojatek.hu/`

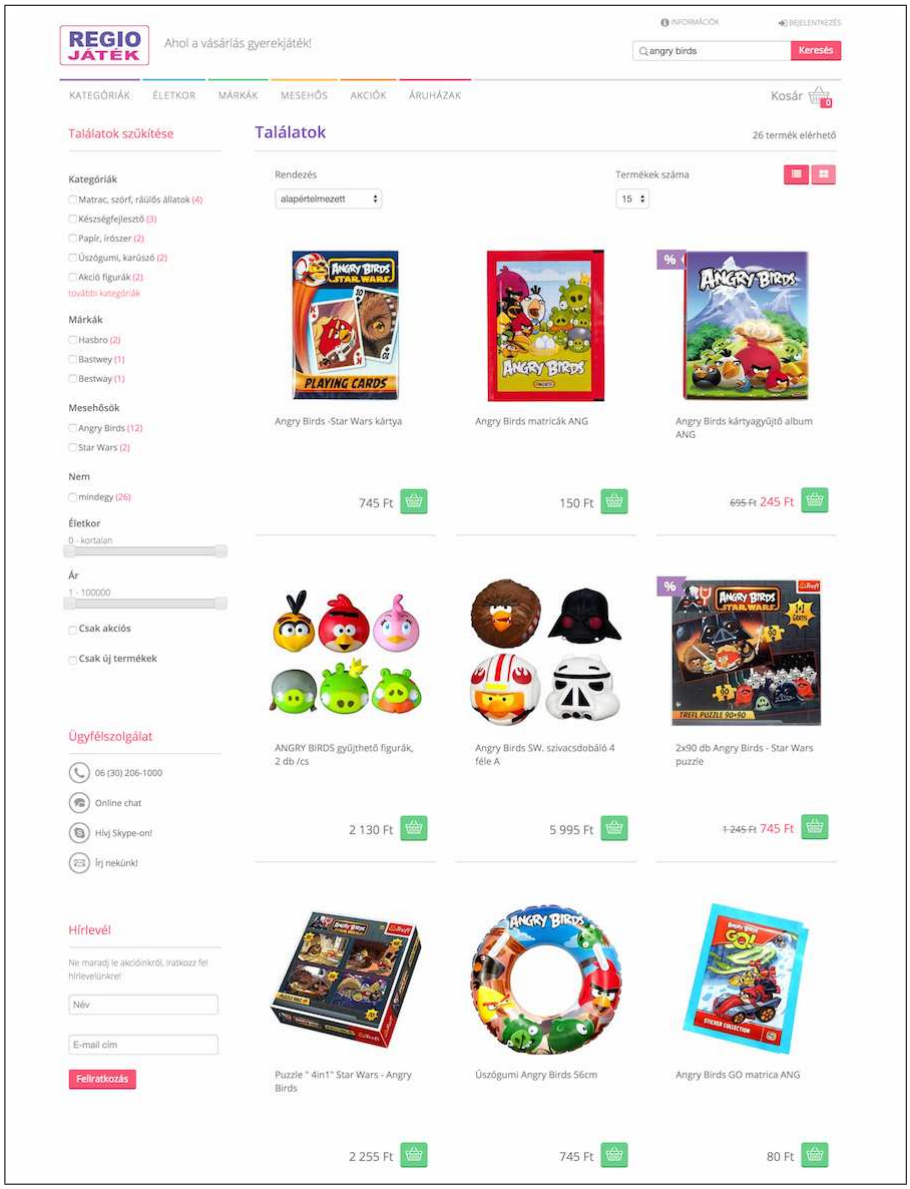

Fig. 6 Screenshot of REGIO, the LL4IR product search use case.

- Gender and age recommendations (for many products);
- Queries (and their distribution) that led to the given product.

Table 1 Fielded document representation of products in the LL4IR product search use case.

Field	Description
age_max	Recommended maximum age (may be empty, i.e., 0)
age_min	Recommended minimum age (may be empty, i.e., 0)
arrived	When the product arrived (first became available); only for products that arrived after 2014-08-28
available	Indicates if the product is currently available (1) or not (0)
bonus_price	Provided only if the product is on sale; this is the new (sales) price
brand	Name of the brand (may be empty)
category	Name of the (leaf-level) product category
category_id	Unique ID of the (leaf-level) product category
characters	List of toy characters associated with the product (may be empty)
description	Full textual description of the product (may be empty)
main_category	Name of the main (top-level) product category
main_category_id	Unique ID of the main (top-level) product category
gender	Gender recommendation. (0: for both girls and boys (or unclassified); 1: for boys; 2: for girls)
photos	List of photos about the product
price	Normal price
product_name	Name of the product
queries	Distribution of (frequent) queries that led to this product (may be empty)
short_description	Short textual description of the product (may be empty)

4.2.3 Candidate Products

The candidate set, to be ranked, contained all products that were available in the (recent) past. This comprises all products that were considered by the site's production search engine (in practice: all products that contain any of the query terms in any of their textual fields). One particular challenge for this use case is that the inventory (as well as the prices) are constantly changing; however, for challenge participants, a single ranking is used throughout the entire test period of the challenge, without the possibility of updating it. The candidate set therefore also includes products that may not be available at the moment (but might become available again in the future). Participating systems were strongly encouraged to consider all products from the provided candidate set. Those that were unavailable at a given point in time were not displayed to users of the REGIO online store. Further, it might happen (and as we show in (Schuth et al, 2015a) it indeed did happen) during the test period that new products arrive; experimental systems were unable to include these in their ranking (this was the same for all participants), while the production system might return them. This can potentially affect the number of wins against the production system (to the advantage of the production system), but it does not affect the comparison across experimental systems.

4.2.4 Submissions and Results

Two organizations submitted a total of four runs. In addition, a simple baseline provided by the challenge organizers was also included for reference. Table 2 presents the results.

4.2.5 Approaches

The organizers' baseline (BASELINE in Table 2) ranks products based on historical click-through rate. Only products that were clicked for the given query are returned; their attributes are ignored. In case historical clicks are unavailable (this happened for a single query `R-q97`), (all) candidate products are returned in an arbitrary order (in practice, in the same order as they were received from the API via the `doclist` request).

The University of Stavanger (Ghirmatsion and Balog, 2015) employed a fielded document retrieval approach based on language modeling techniques. Specifically, building upon the Probabilistic Retrieval Model for Semistructured Data by Kim et al (2009), they experimented with three different methods (UIS-*) for estimating term-field mapping probabilities. Their results show that term-specific field mapping in general is beneficial, but their attempt at estimating field importance based on historical click-through information met with limited success.

Team GESIS (Schaer and Tavakolpoursaleh, 2015) also used a fielded document representation. They used Solr for ranking products and incorporated historical click-through rates, if available, as a weighting factor.

4.2.6 Dealing with Inventory Changes

As mentioned in Section 4.2.1, the product inventory is subject to changes. Not all products that were part of the candidate set were available at all times. If all products were available, the expected probability of winning an interleaved comparison (assuming a randomly clicking user) would be 0.5. However, on average, 44% of the products were actually unavailable. These products were only ever present in the participants' ranking (the site's ranking never considered them). And, only *after* interleaving were these products removed from the resulting interleaved list. We note that this is undesired behavior, as they should have been filtered out *before* interleaving. The necessary adjustments were made to the implementation for the next round of the challenge. As for interpreting these results, this means that the chances for products from the participants ranking to be clicked were reduced. This in turn reduced the expected probability to win to:

$$\Pr(\text{participant} > \text{site}) = (1 - 0.44) \cdot 0.5 = 0.28.$$

Table 2 Results for the product search use case. The expected outcome under a randomly clicking user is 0.28. P-values are computed using a binomial test.

Submission	Outcome	#Wins	#Losses	#Ties	#Impressions	p-value
BASELINE	0.4691	91	103	467	661	< 0.01
UIS-MIRA (Ghirmatsion and Balog, 2015)	0.3413	71	137	517	725	0.053
UIS-JERN (Ghirmatsion and Balog, 2015)	0.3277	58	119	488	665	0.156
UIS-UIS (Ghirmatsion and Balog, 2015)	0.2827	54	137	508	699	0.936
GESIS (Schaer and Tavakolpoursaleh, 2015)	0.2685	40	109	374	523	0.785

Consequently, if a participant's system wins more than in 28% of the impressions, then this is more than expected. And thus the participant's system can be said to be better than the site's system if the outcome is (significantly) more than 28%.

4.2.7 Results

We find that at least three submissions are likely to have improved upon the production system's ranking. Somewhat surprisingly, the simple baseline performed by far the best, with an outcome of 0.4691. This was also the only system that significantly outperformed the production system. The best performing participant run is UIS-MIRA, with an outcome of 0.3413. A more in-depth analysis of the results is provided in the LL4IR extended lab overview paper (Schuth et al, 2015a).

4.3 LL4IR Use Case: Web Search

4.3.1 Task and Data

The *web search* use case has been provided by Seznam[5], a very large web search engine in the Czech Republic. See Figure 7 for a screenshot of the user interface.

Seznam serves almost half the country's search traffic and as such has very high site traffic. Queries are the typical web search queries, and thus are a mixed bag of navigational and transactional (Broder, 2002). In contrast to the product search use case, apart from the scale and the query types, Seznam did not make raw document and query content available, rather features computed for documents and queries. This is much like any learning to rank dataset, such as Letor (Liu et al, 2007). Queries and documents are only identified by a unique identifier and for each query, the candidate documents are represented with sparse feature vectors. Seznam provided a total of 557 features. These features were not described in any way. The challenge with this use case then is a learning to rank challenge (Liu, 2009).

[5] `http://search.seznam.cz/`

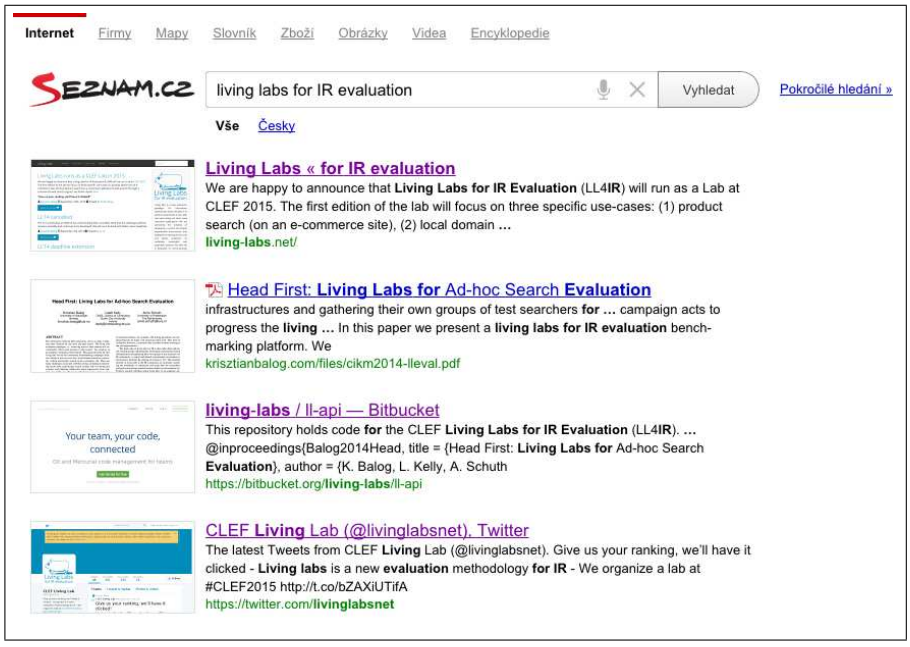

Fig. 7 Screenshot of Seznam, the LL4IR web search use case.

As described in subsection 4.1, the web search use case also consists of a training and test phase. For the test phase, there were 97 queries, for the training phase 100 queries were provided. On average, for each query there were about 179 candidate documents. In total, there were 35,322 documents.

4.3.2 Results

The web search use case attracted six teams that submitted runs for the training queries. However, none of them submitted runs for the test queries. Therefore, we can only report on two baseline systems, provided by the challenge organizers. Baseline 1, titled EXPLOITATIVE BASELINE in Table 3, uses the original Seznam ranking and was therefore expected to produce an outcome of 0.5.[6] Baseline 2, titled UNIFORM BASELINE in Table 3, assigned uniform weights to each feature and ranked by the weighted sum of feature values. This baseline was expected not to perform well.

There were over 440K impressions on Seznam through our Living Labs API. On average this amounts to 2,247 impressions for each query. Approximately 6% of all impressions were used for the testing period. As can be seen in Table 3, the

[6] If use cases uploaded their candidate documents in the order that represented their own ranking, then this was available to participants. We plan to change this in the future.

Table 3 Results for the web search use case. The expected outcome under a randomly clicking user is 0.5. P-values were computed using a binomial test.

Submission	Outcome	#Wins	#Losses	#Ties	#Impressions	p-value
EXPLOITATIVE BASELINE	0.5527	3030	2452	19055	24537	< 0.01
UNIFORM BASELINE	0.2161	430	1560	1346	3336	< 0.01

EXPLOITATIVE BASELINE outperformed the production system. An outcome (outcome measure described in Section 4.1) of 0.5527 has been achieved, with 3,030 wins and 2,452 losses against the production system, and 19,055 ties with it. As expected, the UNIFORM BASELINE lost many more comparisons than it won. Both outcomes were statistically significant according to a binomial test. Again, we refer to the LL4IR extended lab overview paper (Schuth et al, 2015a) for full details.

4.4 LL4IR Discussion

The living labs methodology offers great potential to evaluate information retrieval systems in live settings with real users. The LL4IR CLEF Lab represents the first attempt at a shared community benchmarking platform in this space. The first edition of LL4IR focused on two use-cases, product search and web search, using a commercial e-commerce website, REGIO, and a commercial web search engine, Seznam. Below, we identify some of the main successes and challenges of our initiative.

Successes:

- A major contribution of the lab is the development of the necessary API infrastructure, which has been made publicly available. Overall, we regard our effort successful in showing the feasibility and potential of this form of evaluation. For both use-cases, there was an experimental system that outperformed the corresponding production system significantly. It is somewhat unfortunate that in both cases that experimental system was a baseline approach provided by the challenge organizers, nevertheless, it demonstrates the potential benefits to use-case owners as well.
- The API infrastructure developed for the LL4IR CLEF Lab offers the potential to host ongoing IR evaluations in a live setting. As such, it is planned that these "challenges" will continue on an ongoing basis post-CLEF, with an expanding number of use-cases as well as refinements to the existing use-cases.[7] A more detailed analysis of the use-cases, including results from a second unofficial evaluation round, and a discussion of ideas and opportunities for future development is provided in the LL4IR extended lab overview paper (Schuth et al, 2015a).

[7] See `http://living-labs.net/` for details.

Challenges:

- *Startup challenge:* The LL4IR CLEF Lab attracted interest from dozens of teams. There were twelve active participants, but only two teams ended up submitting results for the official evaluation (excluding the organizers' baseline systems). We found that, while many researchers expressed and showed their interest in the lab, our setup with an API, instead of a static test collection, was a hurdle for many. We plan to ease this process of adapting to this new evaluation paradigm by providing even more examples and by organizing tutorials where we demonstrate working with our API.
- *Frequency of inventory change:* One particular issue that surfaced and needs addressing for the product search use-case is the frequent changes in inventory. This appears to be more severe than we first anticipated and represents some challenges, both technical and methodological.

5 Discussion and Conclusion

In this chapter, we have discussed the importance of conducting online evaluations using real participants conducting real tasks in the wild. We have presented two evaluation initiatives which address this need by offering shared challenges which operate in a living labs setting. Specifically, the NewsREEL shared challenge for recommender systems, and the LL4IR shared challenge for information retrieval. The aim of these initiatives is to close the gap that exists between industry and academia in the evaluation of information access systems. Both campaigns can be seen as initiatives that follow the Evaluation-as-a-Service paradigm discussed by Hopfgartner et al (2018).

We argue that access to living labs style shared challenges, which offer researchers the opportunity to evaluate their algorithms in an online setting with real users of systems, is essential for researchers to be able to study the performance of algorithms under real-world conditions. However, although continuous evaluation of large-scale information access systems is clearly an important tool for advancing the state of the art, we cannot expect living labs to arise spontaneously and automatically. Instead, creating and running initiatives that offer online opportunities for evaluation requires the investment of resources and a great deal of persistence on the part of organizers and participants. A detailed discussion on key technical aspects and efforts required to establish Evaluation-as-a-Service as a mature evaluation methodology is provided by Hopfgartner et al (2018). Extending on their discussion, we close this chapter by highlighting reasons that illustrate the necessity to continue to invest effort into promoting the living labs online evaluation paradigm. As summarized in Table 4, we concentrate our discussion on the differences between traditional evaluation campaigns based on static datasets and living lab campaigns.

- *Representativeness:* As discussed earlier, static test collections have played a significant role in the evaluation of information access methods. In fact, for many

Table 4 Comparison of static test collections and living labs.

	Static test collections	**Living labs**
Representativeness	Data is only as good as the guidelines	Real user data, real and representative information needs
Scalability	Not scalable in terms of users; very scalable in terms of participants	Very scalable in terms of users; for participants, scalability is limited by the site's traffic
Effort (organizers)	One-off	Continuous
Effort (participants)	Moderate	Increased
Reproducibility	Results of previous approaches are easily reproducible	For a fair comparison, a new online evaluation round is needed

years, test collections and related shared evaluation tasks were used to define and to study current research challenges. In the past few years, however, we could observe a paradigm shift, where commercial research on information access systems relies increasingly on online benchmarking, also referred to as A/B testing. The reason for this development is that users and their information needs have become a significant factor that affects retrieval and recommendation algorithms. Static test collections, however, are often not suitable for the development of user-centric techniques. First of all, the need to define search tasks might not really reflect users' real information needs. In addition, relevance judgements might be highly subjective and therefore could have a negative effect on personalization techniques. In addition, the dataset used might not be suitable, e.g., because it is outdated or because the users are not interested in its content. Living labs as described in this chapter, however, can help us to reduce these negative effects. They enable us to rely on real user interactions, i.e., users use the living lab service to satisfy their personal information needs. This allows us to avoid negative factors such as the observer expectancy effect that could impact any type of personalization method.

- *Scalability:* For many years, interactive information access methods were evaluated in relatively small user experiments with a limited number of search tasks and participants. For a detailed discussion on this, we refer the reader to Sakai (2018). University-based researchers in particular employed these small-scale experiments since they often lack access to resources required to perform larger user studies. Industry-based researchers, however, often have access to a large number of users and consequently, large-scale user experiments can nowadays be seen as the de-facto evaluation standard. This differing access to resources, however, has led to a growing gap between academia and industry. Living labs can help in narrowing this gap since they can enable university-based researchers to gain access to a larger user base.
- *Effort (Organizers):* One of the main advantages of shared evaluation tasks is that the effort that goes into their organization is restricted. Although work involved such as defining tasks, document procurement, topic development, conducting experiments, developing relevance assessments, or evaluating results can be time

consuming, they only have to be performed once. Living labs, however, require a continuous efforts from the organizers since they have to guarantee that the live service as well as all technical components that are involved in the evaluation campaign remain fully functional.

- *Effort (Participants):* One of the main advantages of shared evaluation campaigns that rely on static data collections is that these campaigns are often organized in a very similar fashion. Usually, participants are required to produce a ranked list of retrieval results for a given dataset and search task. Then, standard evaluation metrics are calculated, e.g., using the popular tool `trec_eval`.[8] Given this "standardized" approach, experienced information access researchers might find it easier to participate in these tasks since they have to put less effort into understanding the evaluation process. Living labs, however, are more demanding. For example, in NewsREEL, participants need to set up their own server and register it with the open recommendation platform to gain access to the data. Further, they have to make sure that their system is running smoothly over a longer period of time. Our observation from running NewsREEL is that implementing stable solutions that are able to operate over a longer time period was challenging for many participants.
- *Reproducibility*: Scientific progress requires accumulating experimental findings that are reproducible, i.e., ensuring that the findings of testing an algorithm on a test collection can be recreated by another team, thus enabling the new team to develop new approaches and compare them to the first approach. Freire et al (2016) discuss challenges related to reproducibility in offline data-oriented experiments in detail. The authors point out that reproducibility is made difficult by *volatility of the data*, pointing to the example of live streams in which the same situation never occurs again. Future work is needed in order to set up guidelines for reproducing an experiment without using exactly the same data. A related question is the ability to predict the results of online evaluation using offline experiments. We remark that most discussions on reproducibility assume that the evaluation metric is fixed. However, for information access systems, the ideal goal is to ensure that research results can be reproduced in terms of success criteria that go beyond specific evaluation metrics. User satisfaction is a key success criterion, yet, success has many facets (see, e.g., multi-dimensional evaluation models for recommender systems (Said et al, 2012)). It is clear that further work is needed on the development of metrics for evaluating the success of information access systems. Such work will help to further develop the usefulness of both the offline and the online evaluation paradigms.

In summary, there appears to be general agreement that the future of the evaluation of information access systems lies in evaluating under ever-more realistic conditions. In this chapter, we have emphasized the necessity for public benchmarks offering the possibility to test information access systems online in order to bridge the gap between academia and industry. Here, we would also like to point out that industry also stands to benefit from online evaluation initiatives. Internally, a com-

[8] `https://github.com/usnistgov/trec_eval`

pany can only test their own algorithms on their own data stream. Online evaluations offer a valuable opportunity to test algorithms head-to-head with the full range of participating algorithms on other data streams. The widespread agreement on the value of online evaluation stands in contrast to the relatively slow pace at which online evaluation has begun to be adopted in the research community. Our hope is that the motivation and description of online evaluation provided in this chapter will encourage others to continue to invest effort in evaluation that will allow continuous evaluation of large-scale information access systems to realize its full potential.

References

Allan J, Croft B, Moffat A, Sanderson M (2012) Frontiers, challenges, and opportunities for information retrieval: Report from SWIRL 2012 the second strategic workshop on information retrieval in lorne. SIGIR Forum 46(1):2–32

Azzopardi L, Balog K (2011) Towards a Living Lab for Information Retrieval Research and Development - A Proposal for a Living Lab for Product Search Tasks. In: Forner P, Gonzalo J, Kekäläinen J, Lalmas M, de Rijke M (eds) Multilingual and Multimodal Information Access Evaluation. Proceedings of the Second International Conference of the Cross-Language Evaluation Forum (CLEF 2011), Lecture Notes in Computer Science (LNCS) 6941, Springer, Heidelberg, Germany, pp 26–37

Balog K, Elsweiler D, Kanoulas E, Kelly L, Smucker MD (2014a) Report on the CIKM workshop on living labs for information retrieval evaluation. SIGIR Forum 48(1):21–28

Balog K, Kelly L, Schuth A (2014b) Head first: Living labs for ad-hoc search evaluation. In: Proceedings of the 23rd International Conference on Information and Knowledge Management (CIKM'14), ACM, pp 1815–1818

Beck PD, Blaser M, Michalke A, Lommatzsch A (2017) A System for Online News Recommendations in Real-Time with Apache Mahout. In: Working Notes of the 8th International Conference of the CLEF Initiative, Dublin, Ireland, CEUR Workshop Proceedings

Bons P, Evans N, Kampstra P, van Kessel T (2017) A News Recommender Engine with a Killer Sequence. In: Working Notes of the 8th International Conference of the CLEF Initiative, Dublin, Ireland, CEUR Workshop Proceedings

Broder A (2002) A taxonomy of web search. SIGIR Forum 36(2):3–10

Brodt T, Hopfgartner F (2014) Shedding Light on a Living Lab: The CLEF NewsREEL Open Recommendation Platform. In: Proceedings of the Information Interaction in Context conference, Springer-Verlag, IIiX'14, pp 223–226

Chapelle O, Joachims T, Radlinski F, Yue Y (2012) Large-scale validation and analysis of interleaved search evaluation. ACM Transactions on Information Systems (TOIS) 30:1–41

Ciobanu A, Lommatzsch A (2016) Development of a News Recommender System based on Apache Flink. In: Working Notes of the 7th International Conference of the CLEF Initiative, Evora, Portugal, CEUR Workshop Proceedings

Corsini F, Larson M (2016) CLEF NewsREEL 2016: Image based Recommendation. In: Working Notes of the 7th International Conference of the CLEF Initiative, Evora, Portugal, CEUR Workshop Proceedings

Diaz F, White R, Buscher G, Liebling D (2013) Robust models of mouse movement on dynamic web search results pages. In: Proceedings of the 22nd ACM International conference on Information and Knowledge Management (CIKM'13), pp 1451–1460

Domann J, Meiners J, Helmers L, Lommatzsch A (2016) Real-Time News Recommendations Using Apache Spark. In: Working Notes of the 7th International Conference of the CLEF Initiative, Evora, Portugal, CEUR Workshop Proceedings

Freire J, Fuhr N, Rauber A (2016) Reproducibility of data-oriented experiments in e-science (dagstuhl seminar 16041). In: Dagstuhl Reports, Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, vol 6

Gebremeskel G, de Vries AP (2015) The degree of randomness in a live recommender systems evaluation. In: Working Notes for CLEF 2015 Conference, Toulouse, France, CEUR

Ghirmatsion AB, Balog K (2015) Probabilistic field mapping for product search. In: CLEF 2015 Online Working Notes

Golian C, Kuchar J (2017) News Recommender System based on Association Rules at CLEF NewsREEL 2017. In: Working Notes of the 8th International Conference of the CLEF Initiative, Dublin, Ireland, CEUR Workshop Proceedings

Gunawardana A, Shani G (2009) A survey of accuracy evaluation metrics of recommendation tasks. Journal of Machine Learning Research 10(Dec):2935–2962

Hanbury A, Müller H, Balog K, Brodt T, Cormack GV, Eggel I, Gollub T, Hopfgartner F, Kalpathy-Cramer J, Kando N, Krithara A, Lin JJ, Mercer S, Potthast M (2015) Evaluation-as-a-service: Overview and outlook. CoRR abs/1512.07454

Hassan A, Shi X, Craswell N, Ramsey B (2013) Beyond clicks: query reformulation as a predictor of search satisfaction. In: Proceedings of the 22nd ACM International conference on Information and Knowledge Management (CIKM'13), ACM, pp 2019–2028

Hawking D (2015) If SIGIR had an academic track, what would be in it? In: Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, Santiago, Chile, August 9-13, 2015, p 1077

Hofmann K, Whiteson S, de Rijke M (2011) A probabilistic method for inferring preferences from clicks. In: Proceedings of the 20th Conference on Information and Knowledge Management (CIKM'11), ACM, p 249

Hopfgartner F, Brodt T (2015) Join the living lab: Evaluating news recommendations in real-time. In: Advances in Information Retrieval - 37th European Conference on IR Research, ECIR 2015, Vienna, Austria, March 29 - April 2, 2015. Proceedings, pp 826–829

Hopfgartner F, Jose JM (2014) An experimental evaluation of ontology-based user profiles. Multimedia Tools Appl 73(2):1029–1051

Hopfgartner F, Kille B, Lommatzsch A, Plumbaum T, Brodt T, Heintz T (2014) Benchmarking News Recommendations in a Living Lab. In: Kanoulas E, Lupu M, Clough P, Sanderson M, Hall M, Hanbury A, Toms E (eds) Information Access Evaluation – Multilinguality, Multimodality, and Interaction. Proceedings of the Fifth International Conference of the CLEF Initiative (CLEF 2014), Lecture Notes in Computer Science (LNCS) 8685, Springer, Heidelberg, Germany, pp 250–267

Hopfgartner F, Brodt T, Seiler J, Kille B, Lommatzsch A, Larson M, Turrin R, Serény A (2015a) Benchmarking news recommendations: The CLEF newsreel use case. SIGIR Forum 49(2):129–136

Hopfgartner F, Kille B, Heintz T, Turrin R (2015b) Real-time recommendation of streamed data. In: Proceedings of the 9th ACM Conference on Recommender Systems, RecSys 2015, Vienna, Austria, September 16-20, 2015, pp 361–362

Hopfgartner F, Lommatzsch A, Kille B, Larson M, Brodt T, Cremonesi P, Karatzoglou A (2016) The potentials of recommender systems challenges for student learning. In: Proceedings of CiML'16: Challenges in Machine Learning: Gaming and Education

Hopfgartner F, Hanbury A, Mueller H, Eggel I, Balog K, Brodt T, Cormack GV, Lin J, Kalpathy-Cramer J, Kando N, Kato MP, Krithara A, Gollub T, Potthast M, Viegas E, Mercer S (2018) Evaluation-as-a-service for the computational sciences: Overview and outlook. ACM Journal of Data and Information Quality

Jagerman R, Balog K, de Rijke M (2018) Opensearch: Lessons learned from an online evaluation campaign. J Data and Information Quality 10(3):13:1–13:15

Joachims T (2003) Evaluating retrieval performance using clickthrough data. In: Franke J, Nakhaeizadeh G, Renz I (eds) Text Mining, Physica/Springer, pp 79–96

Joachims T, Granka LA, Pan B, Hembrooke H, Radlinski F, Gay G (2007) Evaluating the accuracy of implicit feedback from clicks and query reformulations in web search. ACM Trans Inf Syst 25(2)

Kamps J, Geva S, Peters C, Sakai T, Trotman A, Voorhees E (2009) Report on the SIGIR 2009 workshop on the future of IR evaluation. SIGIR Forum 43(2):13–23

Kelly D, Dumais ST, Pedersen JO (2009) Evaluation challenges and directions for information-seeking support systems. IEEE Computer 42(3):60–66

Kelly L, Bunbury P, Jones GJF (2012) Evaluating personal information retrieval. In: Proceedings of the 34th European Conference on Information Retrieval (ECIR'12), Springer Verlag

Kille B, Hopfgartner F, Brodt T, Heintz T (2013) The plista dataset. In: NRS'13: Proceedings of the International Workshop and Challenge on News Recommender Systems, ACM, pp 14–21

Kille B, Lommatzsch A, Turrin R, Serény A, Larson M, Brodt T, Seiler J, Hopfgartner F (2015) Stream-Based Recommendations: Online and Offline Evaluation as a Service. In: Mothe J, Savoy J, Kamps J, Pinel-Sauvagnat K, Jones GJF, SanJuan E, Cappellato L, Ferro N (eds) Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Sixth International Conference of the CLEF Association (CLEF 2015), Lecture Notes in Computer Science (LNCS) 9283, Springer, Heidelberg, Germany, pp 497–517

Kim J, Xue X, Croft WB (2009) A probabilistic retrieval model for semistructured data. In: Proc. of the 31st European Conference on Information Retrieval (ECIR'09), Springer Verlag, pp 228–239

Kim Y, Hassan A, White R, Zitouni I (2014) Modeling dwell time to predict click-level satisfaction. In: Proc. of the 7th ACM international conference on Web search and data mining (WSDM'14), ACM, pp 193–202

Kohavi R (2015) Online Controlled Experiments: Lessons from Running A/B/n Tests for 12 Years. In: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney, NSW, Australia, August 10-13, 2015, p 1

Kumar V, Khattar D, Gupta S, Gupta M, Varma V (2017) Deep Neural Architecture for News Recommendation. In: Working Notes of the 8th International Conference of the CLEF Initiative, Dublin, Ireland, CEUR Workshop Proceedings

Li J, Huffman S, Tokuda A (2009) Good abandonment in mobile and pc internet search. In: Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval (SIGIR '09), ACM, pp 43–50

Liang Y, Loni B, Larson M (2017) CLEF NewsREEL 2017: Contextual Bandit News Recommendation. In: Working Notes of the 8th International Conference of the CLEF Initiative, Dublin, Ireland, CEUR Workshop Proceedings

Liu TY (2009) Learning to rank for information retrieval. Found Trends Inf Retr 3(3):225–331

Liu TY, Xu J, Qin T, Xiong W, Li H (2007) LETOR: Benchmark dataset for research on learning to rank for information retrieval. In: Proceedings of SIGIR 2007 Workshop on Learning to Rank for Information Retrieval (LR4IR'07), pp 346–374

Lommatzsch A, Albayrak S (2015) Real-time recommendations for user-item streams. In: Proc. of the 30th Symposium On Applied Computing, SAC 2015, ACM, SAC '15, pp 1039–1046

Lommatzsch A, Johannes N, Meiners J, Helmers L, Domann J (2016) Recommender Ensembles for News Articles based on Most-Popular Strategies. In: Working Notes of the 7th International Conference of the CLEF Initiative, Evora, Portugal, CEUR Workshop Proceedings

Lommatzsch A, Kille B, Hopfgartner F, Larson M, Brodt T, Seiler J, Özgöbek Ö (2017) CLEF 2017 NewsREEL Overview: A Stream-Based Recommender Task for Evaluation and Education. In: Jones GJF, Lawless S, Gonzalo J, Kelly L, Goeuriot L, Mandl T, Cappellato L, Ferro N (eds) Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Eighth International Conference of the CLEF Association (CLEF 2017), Lecture Notes in Computer Science (LNCS) 10456, Springer, Heidelberg, Germany, pp 239–254

Ludmann C (2017) Recommending News Articles in the CLEF News Recommendation Evaluation Lab with the Data Stream Management System Odysseus. In: Working Notes of the 8th International Conference of the CLEF Initiative, Dublin, Ireland, CEUR Workshop Proceedings

Radlinski F, Craswell N (2013) Optimized interleaving for online retrieval evaluation. In: Proc. of ACM Int. Conf. on Web Search and Data Mining (WSDM'13), ACM, pp 245–254

Radlinski F, Kurup M, Joachims T (2008) How does clickthrough data reflect retrieval quality? In: Proceedings of the 17th Conference on Information and Knowledge Management (CIKM'08), ACM, pp 43–52

Said A, Tikk D, Stumpf K, Shi Y, Larson M, Cremonesi P (2012) Recommender systems evaluation: A 3d benchmark. In: Proceedings of the Workshop on Recommendation Utility Evaluation: Beyond RMSE (RUE 2012), CEUR-WS Vol. 910, RUE'12, pp 21–23

Sakai T (2018) Laboratory Experiments in Information Retrieval. Springer Verlag

Schaer P, Tavakolpoursaleh N (2015) GESIS at CLEF LL4IR 2015. In: CLEF 2015 Online Working Notes

Schuth A, Sietsma F, Whiteson S, Lefortier D, de Rijke M (2014) Multileaved comparisons for fast online evaluation. In: Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management (CIKM'14), ACM, pp 71–80

Schuth A, Balog K, Kelly L (2015a) Extended overview of the Living Labs for Information Retrieval Evaluation (LL4IR) CLEF lab 2015. In: CLEF 2015 Online Working Notes

Schuth A, Bruintjes RJ, Büttner F, van Doorn J, Groenland C, Oosterhuis H, Tran CN, Veeling B, van der Velde J, Wechsler R, Woudenberg D, de Rijke M (2015b) Probabilistic multileave for online retrieval evaluation. In: Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'15), ACM, pp 955–958

Schuth A, Hofmann K, Radlinski F (2015c) Predicting search satisfaction metrics with interleaved comparisons. In: Proceedings of the 38th ACM International Conference on Information Retrieval (SIGIR'15), ACM, pp 463–472

Scriminaci M, Lommatzsch A, Kille B, Hopfgartner F, Larson M, Malagoli D, Serény A, Plumbaum T (2016) Idomaar: A framework for multi-dimensional benchmarking of recommender algorithms. In: Proceedings of the Poster Track of the 10th ACM Conference on Recommender Systems (RecSys 2016), Boston, USA, September 17, 2016.

Tavakolifard M, Gulla JA, Almeroth KC, Hopfgartner F, Kille B, Plumbaum T, Lommatzsch A, Brodt T, Bucko A, Heintz T (2013) Workshop and challenge on news recommender systems. In: Seventh ACM Conference on Recommender Systems, RecSys '13, Hong Kong, China, October 12-16, 2013, pp 481–482

Teevan J, Dumais S, Horvitz E (2007) The potential value of personalizing search. In: Proceedings of the ACM International Conference on Information Retrieval (SIGIR'07), ACM, pp 756–757

Turpin A, Scholar F (2006) User performance versus precision measures for simple search tasks. In: Proc. of the ACM International Conference on Information Retrieval (SIGIR'06), ACM, pp 11–18

Verbitskiy I, Probst P, Lommatzsch A (2015) Developing and evaluation of a highly scalable news recommender system. In: Working Notes for CLEF 2015 Conference, Toulouse, France, CEUR

Voorhees EM, Harman DK (2005) TREC: Experiment and Evaluation in Information Retrieval, 1st edn. MIT Press, Cambridge, MA, USA

Wang K, Gloy N, Li X (2010) Inferring search behaviors using partially observable markov (pom) model. In: WSDM'10, ACM, pp 211–220

Wilkins P, Byrne D, Jones GJF, Lee H, Keenan G, McGuinness K, O'Connor NE, O'Hare N, Smeaton AF, Adamek T, Troncy R, Amin A, Benmokhtar R, Dumont E, Huet B, Mérialdo B, Tolias G, Spyrou E, Avrithis YS, Papadopoulos GT, Mezaris V, Kompatsiaris I, Mörzinger R, Schallauer P, Bailer W, Chandramouli K, Izquierdo E, Goldmann L, Haller M, Samour A, Cobet A, Sikora T, Praks P, Hannah D, Halvey M, Hopfgartner F, Villa R, Punitha P, Goyal A, Jose JM (2008) K-space at trecvid 2008. In: TRECVID 2008 workshop participants notebook papers, Gaithersburg, MD, USA, November 2008

Yilmaz E, Verma M, Craswell N, Radlinski F, Bailey P (2014) Relevance and Effort: An Analysis of Document Utility. In: Proceedings of the 23rd ACM International Conference on Information and Knowledge Management (CIKM'14), ACM, pp 91–100