PROMISE

Participative Research labOratory for Multimedia and Multilingual Information Systems Evaluation

FP7 ICT 2009.4.3, Intelligent Information Management

Deliverable 3.4
Report on the outcomes of the continuous evaluation activities

Version 1.0, August 2013

Network of Excellence co-funded by the 7th Framework Program of the European Commission, grant agreement no. 258191

PROMISE
Participative Research labOratory for Multimedia and Multilingual Information Systems Evaluation

SEVENTH FRAMEWORK PROGRAMME

Document Information

Deliverable number:	3.4
Deliverable title:	Report on the outcomes of the continuous evaluation activities
Delivery date:	26/09/2013
Lead contractor for this deliverable	UvA
Author(s):	Giuseppe Bandiera, Richard Berendsen, Martin Braschler, Aleksandr Chuklin, Katja Hofmann, Nicola Ferro, Melanie Himof, Henning Müller, Maarten de Rijke
Participant(s):	UNIPD, UvA, HES-SO, ZHAW
Workpackage:	3
Workpackage title:	Evaluation infrastructure
Workpackage leader:	UNIPD
Dissemination Level:	PU – Public
Version:	1.0
Keywords:	Continuous evaluation, offline evaluation, user studies, online evaluation, evaluation infrastructure.

History of Versions

Version	Date	Status	Author (Partner)	Description/Approval Level
1.0	02/01/2013	Draft	UvA	Circulated to all partners
1.1	28/08/2013	Revised draft	UvA	Revised after partners' comments
1.2	15/09/2013	Final	UvA	Finalized

Abstract

This deliverable describes the activities on continuous evaluation carried in the context of task 3.4 of workpackage 4. It presents a substantial overview of the use of the DIRECT infrastructure in outside the familiar CLEF campaigns. In particular, the deliverable reports on the use of DIRECT for evaluating Europeana. In addition, the deliverable reports on a number of innovative case studies in continuous evaluation and the implications of the lessons learned for the DIRECT infrastructure.

Table of Contents

Executive summary

This deliverable reports on the continuous evaluation activities carried out within the PROMISE network of excellence. The main focus of the work, and of the deliverable, is on large scale uses of DIRECT outside the annual CLEF campaigns, especially concerning Europeana. In addition, we report on a series of case studies in continuous evaluation that have been carried outside DIRECT, but with a view to understanding how DIRECT can be used and/or extended to cater for evaluation setups that deviate from the traditional offline evaluation methodology, the so-called Cranfield methodology. Alternative evaluation methodologies considered in T3.4 on continuous evaluation (M13-36) and, further explored in detail in T4.4 on living retrieval laboratories (M1–36) fall under the "user studies" heading and under the "offline evaluation" heading. Each of the case studies describes its task setting, the mesurements conducted, the lessons learned and, especially, the broader implication for the potential of the DIRECT infrastructure to cater for the case study.

Our main finding is that the DIRECT infrastructure supports a very broad range of evaluation activities, far beyond the Cranfield-style offline settings that it was originally desined for. Thanks to the work carried out for the semantic enrichment (D3.6, M36) and bibliometrics (D6.4, M36), the DIRECT data model has been extended (D3.5, M36) to facilitate a broad set of evaluation activities beyond the traditional offline setting. For example, for some types of evaluation methodology, the inclusion of a user model as part of the metric is essential. The DIRECT infrastructure allows us to attach, with a confidence score, profiles to users while the profiles themselves can be linked together with semantic relations in order to favour navigation and further processing. Moreover, metrics, statistics, and visualizations can be computed and attached to both users and profiles. These features, which are already present, can be exploited to implement the above mentioned user models and the computation of offline metrics over them.

This deliverable has been delayed by six months, from month 30 to month 36. The deliverable builds on activities that ran until month 36, with deliverables in month 36, and allows us to bring together updates to other deliverables that are the result of comments by our reviewers but would otherwise not have had an appropriate deliverable as a landing place. Importantly, the delay of D3.4 from month 30 to month 36 did not affect the overall PROMISE workplan. On the contrary, the delay allowed us to pull together the outcomes of a broad range methodologically diverse evaluation activities and to assess their implications for the DIRECT infrastructure.

1 Introduction

In the setting of multimedia and multilingual information systems, the primary purpose of evaluation, in addition to gaining insight into prior or existing algorithmic solutions, is to enable reflection and assist in the identification of future improvements. There are three broad families of evaluation methods for information access systems, that inform us in different ways about this primary purpose of evaluation:

- In *offline evaluation*, we collect a set of queries, describe the information being sought, have assessors determine which documents are relevant, and then evaluate systems based on the quality of their rankings. The evaluation metrics used typically describe the quality of the ranking based on known relevant and non-relevant items.
- In *user studies* we provide a small set of users with several retrieval systems, ask them to complete several, potentially different tasks, and learn about system performance by observing what the users do and asking them why they did it.
- In *online evaluation*, we see how normal users of interact with a live retrieval system when just using it to achieve the tasks they want to achieve. Here, we observe implicit signals only (clicks, skips, saves, etc.) and we try to infer differences in behaviour from different flavors of the live system.

In this deliverable we showcase the potential of the DIRECT architecture in bringing automation into the evaluation process. We take "continuous evaluation" to mean two things:

- Continuous access of researchers to DIRECT, to submit runs and get evaluation results outside the annual evaluation cycles carried out in WP6; Direct mostly caters for offline evaluation.
- Continuous evaluation efforts within each of the three flavors of information access system evaluation listed above: offline, user study, and online.

Based on this perspective, the deliverable is organized along the following storyline. We report on two types of things, results and informed suggestions:

A. Evaluation work based on DIRECT outside the CLEF cycles. This material is covered in Sections 2 and 3.
B. Continuous evaluation work, both offline, user study-based and online, carried out by the PROMISE project partners plus a reflection on the degree to which this evaluation work could in principle be carried with DIRECT. This material is covered in Section 4.

Sections 2, 3 and 4 are organized around a fair number of individual case studies concerning continuous evaluation. In Section 5 we zoom out and conclude with implications for the DIRECT infrastructure.

2 DIRECT outside the CLEF campaigns: Evaluation of Europeana

During 2011, a joint effort between PROMISE and Europeana, at that time the EuropeanaConnect project in particular, has started to conduct a systematic evaluation of the multilingual information access components that were under development.

This effort took the form of a mini-evaluation campaign organized to assess and compare several alternative implementations of Europeana multilingual components where CLEF experimental collections have been used and the DIRECT system has been exploited to manage the evaluation process and compute the experimental results.

2.1 Experimental setup

2.1.1 Collections

In order to ensure comparability with existing literature and existing systems whose performances are known, we made use of the CLEF collections developed for the Ad-hoc TEL Tasks in CLEF 2008 and CLEF 2009 [Agirre et al., 2009; Ferro and Peters, 2010]. This task offered monolingual and cross-language search on library catalogues. It was organized in collaboration with *The European Library* and used three collections derived from the catalogs of the British Library, the Bibliothéque Nationale de France, and the Austrian National Library. These collections contain catalogue records expressed in an embryonal version of what then has become the Europeana Semantic Elements (ESE) and, thus, they are representative for what is currently used, before a full deployment of the newest Europeana Data Model (EDM).

We used three collections:

- **British Library (BL)**: 1,000,100 catalog records, 1.2 GByte of uncompressed XML;
- **Bibliothéque Nationale de France (BNF)**: 1,000,100 catalog records, 1.3 GByte of uncompressed XML;
- **Austrian National Library (ONB)**: 869,353 catalog records, 1.3 GByte of uncompressed XML.

We refer to the three collections (BL, BNF, ONB) as English, French and German because, in each case, this is the main and expected language of the collection. However, each of these collections is to some extent multilingual and contains documents in many additional languages; roughly speaking, about 60%-70% of the collections is in the “main language” and the remaining 30%-40% is in other languages as shows in Figure 1.

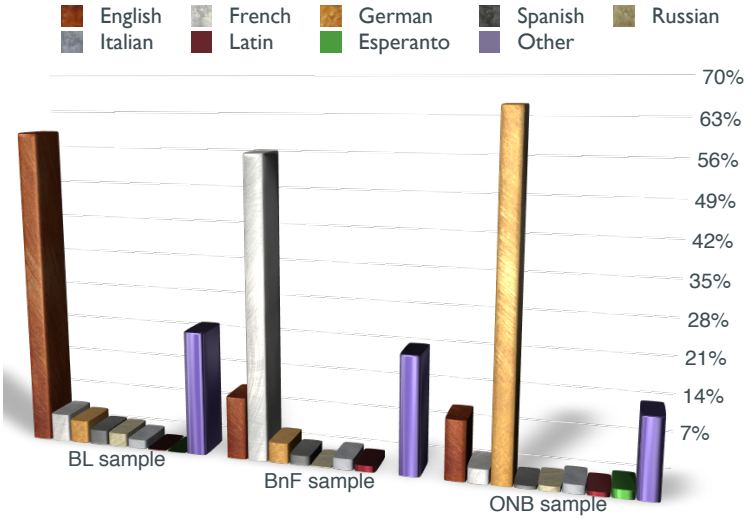

Figure 1: Distribution of the languages in the BL, BnF, and ONB collections.

Many records contain only title, author and subject heading information; other records provide more detail. The title and (if existing) abstract or description may be in a different language to that understood as the language of the collection. The subject heading information is normally in the main language of the collection. About 66% of the documents in the English and German collection have textual subject headings, while only 37% in the French collection. Dewey Classification (DDC) is not available in the French collection; negligible (<0.3%) in the German collection; but occurs in about half of the English documents (456,408 docs to be exact). Figure 2 shows the distributions of the records fields in the three collections.

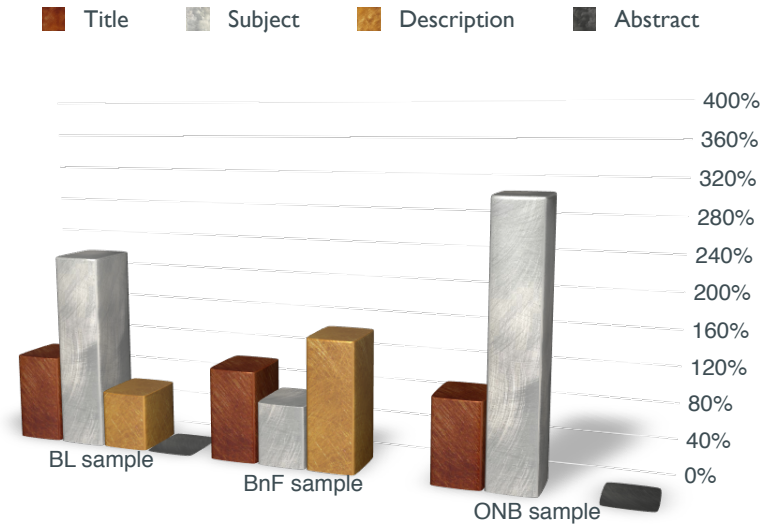

Figure 2: Distribution of the record fields in the BL, BnF, and ONB collections (percentages greater than 100% means that the field is repeated more than one in a record, on average).

2.1.2 Topics

For the evaluation, we used a common set of 100 topics in each of the 3 main collection languages (English, French and German). These topics were translated to all the 10 EuropeanaConnect languages, namely: English, French, German, Italian, Polish, Spanish, Portuguese, Swedish, Dutch and Hungarian.

Only the Title and Description fields were used in the evaluation because the narrative was prepared to provide information for the assessors on how the topics should be judged during CLEF. The topic sets were prepared on the basis of the contents of the collections, i.e. by interactively searching the collections to ensure the existence of relevant documents for each topic. More in detail, when a task uses data collections in more than one language, we consider it important to be able to use versions of the same core topic set to query all collections. This makes it easier to compare results over different collections and also facilitates the preparation of extra topic sets in additional languages. However, it is never easy to find topics that are effective for several different collections and the topic preparation stage requires considerable discussion between the coordinators for each collection in order to identify suitable common candidates. The sparseness of the data makes this particularly difficult for the TEL task and leads to the formulation of topics that are quite broad in scope so that at least some relevant documents could be found in each collection.

2.1.3 Relevance judgements

The number of documents in large test collections such as CLEF makes it impractical to judge every document for relevance. Instead approximate recall values are calculated using pooling techniques. The results submitted by the groups participating in the ad hoc tasks are used to form a pool of documents for each topic and language by collecting the highly ranked documents from selected runs according to a set of predefined criteria. Traditionally, the top 100 ranked documents from each of the runs selected are included in the pool; in such a case we say that the pool is of depth 100. This pool is then used for subsequent relevance judgments. After calculating the effectiveness measures, the results are analyzed and run statistics produced and distributed. The stability of pools constructed in this way and their reliability for post-campaign experiments is discussed in [Braschler, 2003].

The main criteria used when constructing the pools in CLEF are:

- favour diversity among approaches adopted by participants, according to the descriptions that they provide of their experiments;
- for each task, include at least one experiment from every participant, selected from the experiments indicated by the participants as having highest priority;
- ensure that, for each participant, at least one mandatory title + description experiment is included, even if not indicated as having high priority;
- add manual experiments, when provided;
- or bilingual tasks, ensure that each source topic language is represented.

For the CLEF 2008 ad hoc test collections, Tomlinson [2009] reported some sampling experiments aimed at estimating the judging coverage. He found that this tended to be lower than the estimates he produced for the CLEF 2007 ad hoc collections. With respect to the TEL collections, he estimated that at best 50% to 70% of the relevant documents were

included in the pools – and that most of the un-judged relevant documents were for the 10 or more queries that had the most known answers.

These discussions show how complex the creation of relevance judgements is and how much care is devoted to ensure that they are reliable and robust. The net results, from our evaluation point of view, is that the CLEF relevance judgements are fair also for systems that have not participated in the CLEF campaigns, as it is the case for Europeana, and can provide third-party assessment.

2.2 Tasks

For Europeana we evaluated the following:

- **monolingual tasks** where the language of the source query is the same as that of the target collection, for example an English query against an BL collection;
- **bilingual tasks** where the language of the source query is different from that of the target collection, for example a Dutch query against an BL collection.

Monolingual tasks offered the possibility of assessing the performances provided by different language resources. For each of the three target collections, we provided a baseline run and then evaluate the different language resources available in the Europeana language resources repository.

Bilingual tasks offered the possibility of evaluating the translation modules together with their interaction with language resources (both monolingual ones, such as stemmers, and bilingual ones, such as dictionaries).

For each of the three target collections, we provided a baseline run and then evaluate the different (translation module, language resource) pairs with respect to the ten EuropeanaConnect languages. In order to do that, CELI provides a standard information retrieval system, where all the components will be kept fixed except for the (translation module, language resource) under testing.

2.3 Experimental data management

The experimental data and the evaluation process have been managed by means of the DIRECT system, developed within the PROMISE NoE [Agosti et al, 2011b,c; 2012].

A specific instance of DIRECT has been setup and customized for supporting the evaluation of the multilingual components of Europeana. This concerned submission of experiments, computation of performance measures, computation of descriptive statistics, and access and re-use of submitted experiments.

Figure 3 shows the login page of the DIRECT instance customized for Europeana.

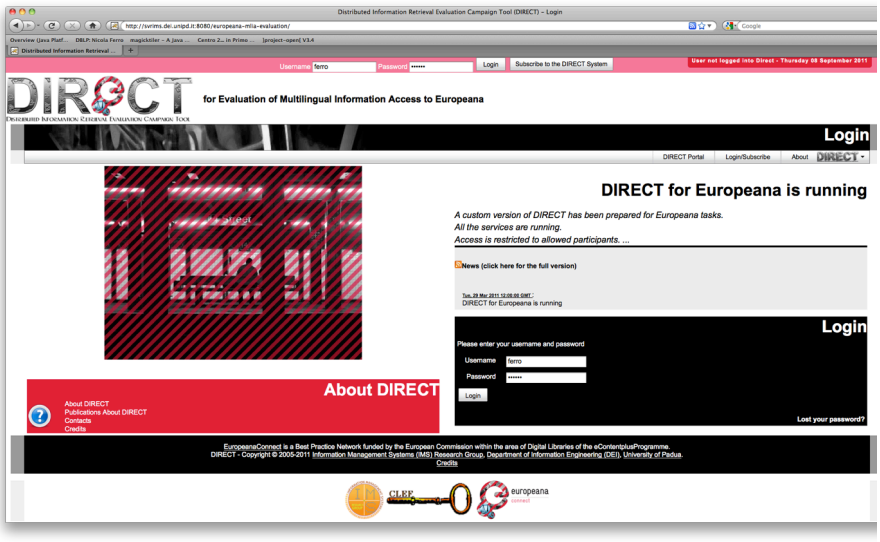

Figure 3: Screenshot of the login page of the DIRECT instance for Europeana MLIA evaluation.

A total of 374 have been submitted and managed via DIRECT, which amounts to 33,289,416 experiment items, 972,400 performance measures, and 165,308 descriptive statistics. This data is now available for comparison, re-use, and exploitation via the main DIRECT portal, where also the experimental data from the CLEF campaigns are. Figure 4 shows the Europeana experimental data accessible through DIRECT: the list of monolingual and bilingual tasks is shown in the tree on the left. When a task is select, it is possible:

- to download its relevance judgements, in the tree on the left;
- to download its topics, in the tree on the left;
- to see the full list of experiments for that task, in the content pane on the right.

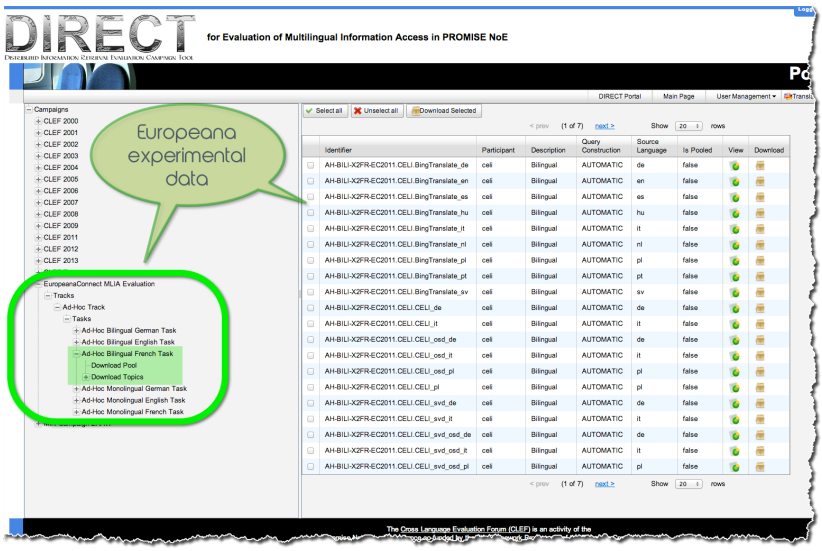

Figure 4: Screenshot of the login page of the DIRECT instance for Europeana MLIA evaluation.

In the content pane on the right, for each experiment, some summary information are reported and it is then possibile:

- to download the experiment itself, via the “Download” button;
- to access its performance measures and statistics, via the “View” button.

When you click on the “View” button, you are redirected to a page providing summary information about the experiment, as shown in Figure 5.

Figure 5: Summary information for one of the Europeana experiments.

From this page, you can get access to additional information about an experiment via the different tabs, such as topic-by-topic measures in the “Metrics” tab (Figure 6) or overall descriptive statistics for the whole experiment and for each of the measures in the “Descriptive Statsistics” tab (Figure 7).

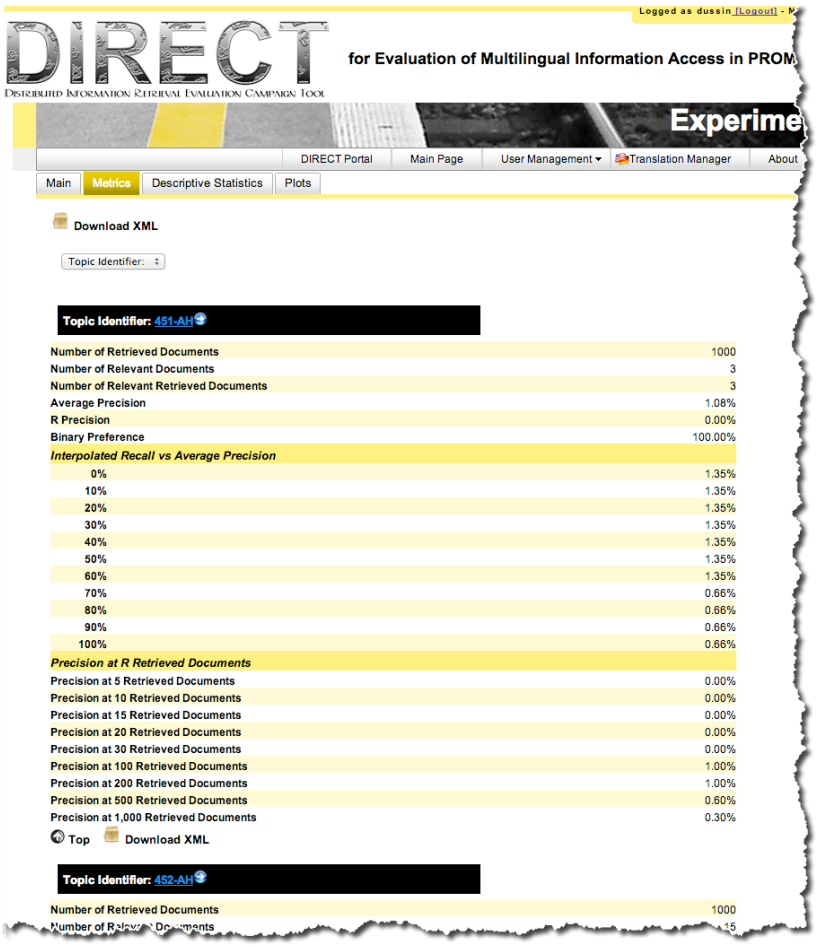

Figure 6: Performance measures about one of the Europeana experiments.

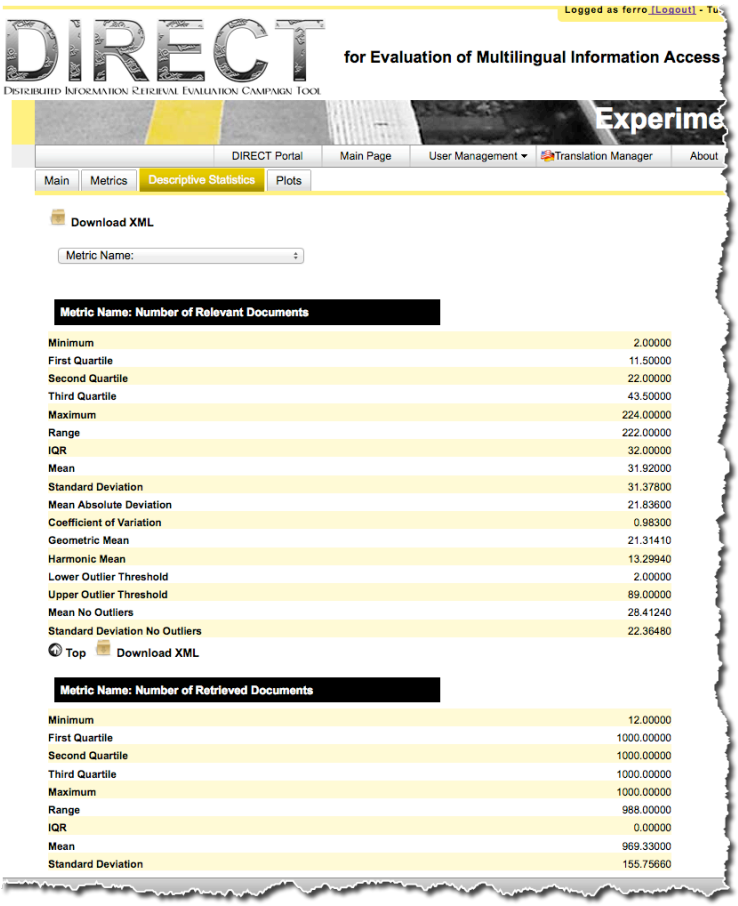

Figure 7: Descriptive stastistics about one of the Europeana experiments.

2.4 Experimental results

Among all the metrics computed by DIRECT, The following ones have been adopted by Europeana to assess its multilingual information access components:

- **average precision**: is a single-valued measure that reflects the performance over all relevant documents. It rewards systems that retrieve relevant documents quickly (highly ranked).
- **precision@5**: is the precision after 5 documents have been retrieved. If you think at it in terms of a results list of a search engine with 10 results displayed per page, it gives you an idea of the performances at the mid of the first page.
- **precision@10**: is the precision after 10 documents have been retrieved. If you think at it in terms of a results list of a search engine with 10 results displayed per page, it gives you an idea of the performances at the end of the first page.
- **precision@20**: is the precision after 5 documents have been retrieved. If you think at it in terms of a results list of a search engine with 10 results displayed per page, it gives you an idea of the performances at the end of the second page.

- **R_precision**: is the precision after R documents have been retrieved, where R is the total number of relevant document that can be retrieved. It de-emphasizes the exact ranking of the retrieved relevant documents.

2.4.1 Monolingual runs

Table 1 reports the best results achieved in the three monolingual tasks (English, French, and German) for the above descripted metrics.

Task	Best Mean Average Precision	Best Mean Precision@5	Best Mean Precision@10	Best Mean Precision@20	Best Mean R_Precision
Monolingual English	27.46%	51.20%	45.30%	36.35%	29.90%
Monolingual French	23.35%	39.20%	34.50%	27.05%	25.68%
Monolingual German	13.48%	33.00%	27.20%	20.85%	16.03%

Table 1: Best results for the monolingual tasks.

2.4.2 Bilingual runs

Table 2 reports the best results achieved in the three bilingual tasks (X → English, X → French, and X → German) for the above descripted metrics.

For each target language (English, French, and German), the results achieved with different source languages (Dutch, English, French, German, Hungarian, Italian, Polish, Portuguese, Spanish, and Swedish) are reported as well as the comparison with respect to the corresponding monolingual baseline.

For example, the best mean average precision for the bilingual Dutch to English experiments is 19.49% while the best mean average precision for the monolingual English experiments (see Table 1) is 27.46%; therefore, the bilingual Dutch to English achieves 70.98% of the performances of the monolingual English.

Task	Best Mean Average Precision	Best Mean Precision@5	Best Mean Precision@10	Best Mean Precision@20	Best Mean R_Precision
Bilingual To English					
Dutch	19.49%	40.80%	33.70%	22.30%	21.73%
wrt monolingual baseline	70.98%	79.69%	74.39%	61.35%	72.68%
French	19.18%	39.40%	33.40%	26.40%	21.73%
wrt monolingual baseline	69.85%	76.95%	73.73%	72.63%	72.68%

Task	Best Mean Average Precision	Best Mean Precision@5	Best Mean Precision@10	Best Mean Precision@20	Best Mean R_Precision
German	17.65%	34.40%	29.30%	24.60%	20.10%
wrt monolingual baseline	64.28%	67.19%	64.68%	67.68%	67.22%
Hungarian	14.02%	29.40%	25.90%	20.25%	15.91%
wrt monolingual baseline	51.06%	57.42%	57.17%	55.71%	53.21%
Italian	18.97%	38.60%	31.80%	25.80%	21.16%
wrt monolingual baseline	69.08%	75.39%	70.20%	70.98%	70.77%
Polish	18.24%	35.20%	29.40%	24.20%	20.52%
wrt monolingual baseline	66.42%	68.75%	64.90%	66.57%	68.63%
Portuguese	21.05%	39.80%	33.90%	28.85%	23.80%
wrt monolingual baseline	76.66%	77.73%	74.83%	79.37%	79.60%
Spanish	16.74%	36.60%	28.60%	23.85%	19.70%
wrt monolingual baseline	60.96%	71.48%	63.13%	65.61%	65.89%
Swedish	16.77%	33.00%	29.50%	25.00%	19.46%
wrt monolingual baseline	61.07%	64.45%	65.12%	68.78%	65.08%
Bilingual To French					
Dutch	11.76%	20.60%	18.10%	13.35%	12.53%
wrt monolingual baseline	50.36%	52.55%	52.46%	49.35%	48.79%
English	15.77%	27.40%	23.90%	18.40%	16.55%
wrt monolingual baseline	67.54%	69.90%	69.28%	68.02%	64.45%
German	12.77%	22.80%	18.80%	15.25%	13.78%
wrt monolingual baseline	54.69%	58.16%	54.49%	56.38%	53.66%
Hungarian	9.29%	17.00%	14.10%	11.20%	10.50%
wrt monolingual baseline	39.79%	43.37%	40.87%	41.40%	40.89%
Italian	15.73%	28.00%	22.60%	17.04%	16.50%
wrt monolingual baseline	67.37%	71.43%	65.51%	62.99%	64.25%
Polish	12.42%	23.80%	18.90%	14.10%	13.62%
wrt monolingual baseline	53.19%	60.71%	54.78%	52.13%	53.04%
Portuguese	15.74%	30.60%	25.30%	19.00%	16.90%
wrt monolingual baseline	67.41%	78.06%	73.33%	70.24%	65.81%
Spanish	10.17%	21.00%	17.20%	13.00%	11.76%
wrt monolingual baseline	43.55%	53.57%	49.86%	48.06%	45.79%
Swedish	11.29%	20.80%	18.10%	14.00%	12.97%

Task	Best Mean Average Precision	Best Mean Precision@5	Best Mean Precision@10	Best Mean Precision@20	Best Mean R_Precision
wrt monolingual baseline	48.35%	53.06%	52.46%	51.76%	50.51%
Bilingual To German					
Dutch	11.08%	22.60%	18.40%	15.10%	12.30%
wrt monolingual baseline	82.20%	68.48%	67.65%	72.42%	76.73%
English	9.12%	22.00%	16.80%	12.60%	10.14%
wrt monolingual baseline	67.66%	66.67%	61.76%	60.43%	63.26%
French	10.08%	23.00%	17.70%	13.55%	11.34%
wrt monolingual baseline	74.78%	69.70%	65.07%	64.99%	70.74%
Hungarian	9.02%	15.20%	14.00%	11.55%	9.92%
wrt monolingual baseline	66.91%	46.06%	51.47%	55.40%	61.88%
Italian	10.00%	23.40%	18.60%	13.50%	11.05%
wrt monolingual baseline	74.18%	70.91%	68.38%	64.75%	68.93%
Polish	10.10%	19.60%	17.20%	13.30%	11.65%
wrt monolingual baseline	74.93%	59.39%	63.24%	63.79%	72.68%
Portuguese	12.77%	27.20%	23.40%	18.25%	14.55%
wrt monolingual baseline	94.73%	82.42%	86.03%	87.53%	90.77%
Spanish	9.52%	21.00%	16.00%	12.95%	10.58%
wrt monolingual baseline	70.62%	63.64%	58.82%	62.11%	66.00%
Swedish	11.12%	22.40%	19.40%	15.50%	11.80%
wrt monolingual baseline	82.49%	67.88%	71.32%	74.34%	73.61%

Table 2: Best results for the bilingual tasks.

3 Continuous evaluation of Europeana

After the positive experience in evaluating Europeana components on CLEF datasets using DIRECT, during 2012 and 2013 it was decided to perform an additional step and introduce the possibility of continuously evaluating the production Europeana platform.

3.1 Workflow and architecture

Figure 8 shows the main actors involved in the continuous evaluation process and the interactions among the:

- **Europeana** (on the left): the Europena production system, accessible through the Europeana API[1];
- **Continuous Evaluation Manager** (in the middle): the component of the PROMISE evaluation infrastructure in charge of managing the overall evaluation process;
- **DIRECT** (on the right): the DIRECT RESTful Web service to manage and access the experimental data.

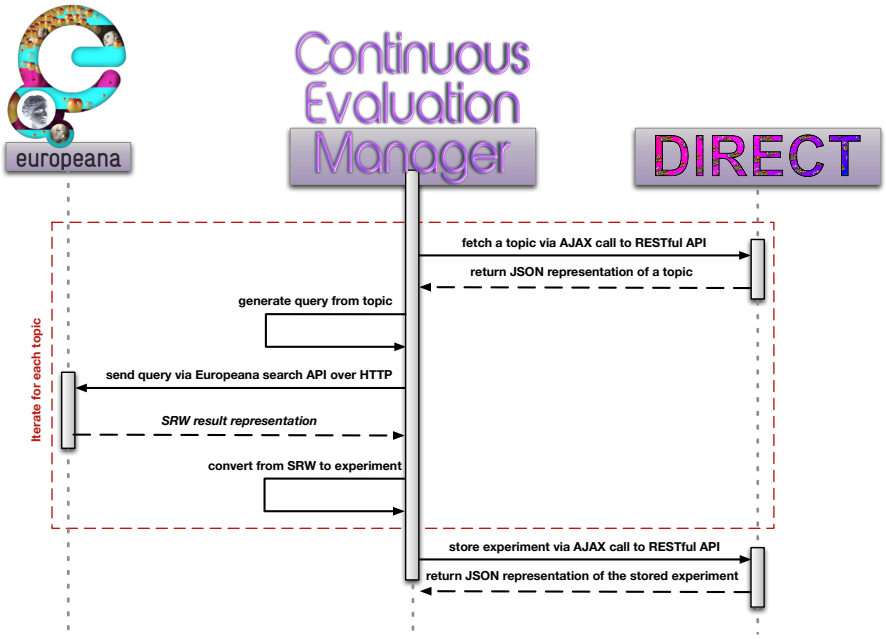

Figure 8: Continuous evaluation workflow.

The continuous evaluation process works as follows: for each topic in the data set, the continuous evaluation manager fetches the topic from DIRECT using the RESTful API described in D3.3 [Agosti et al., 2012] and generates a query to be sent to Europeana according to the topic fields indicated by the user. Then, it send via AJAX the query to

[1] http://pro.europeana.eu/api

Europeana and retrieves the results in the SRW format. Finally, it converts the results from the SRW format to the DIRECT experiment format, described in D3.3.

When all the topics have been processed, the continuous evaluation manager assembles a whole experiment and stores it in the DIRECT system.

Figure 9 shows how the architecture of the PROMISE evaluation infrastructure, which has been introduced in D3.3, has been extended to support the continuous evaluation process.

The continuous evaluation manager, whose functioning has been described above, has been developed as a portlet in the PROMISE evaluation portal hosted on a Liferay instance. This portlet manages the interaction with the users, getting the setup for the experiment, showing the progress of the evaluation, and summarizing the results before submission. It then manages the workflow and interacts with Europena and DIRECT via AJAX calls according to the respective APIs.

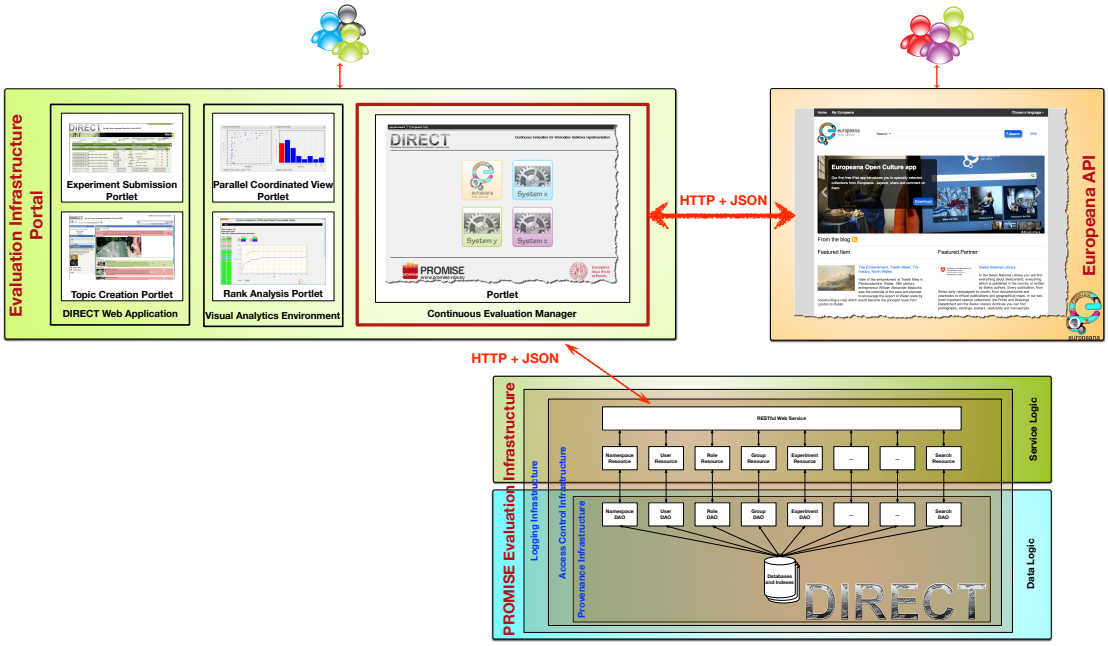

Figure 9: Architecture of the PROMISE evaluation infrastructure extended with the continuous evaluaton manager.

3.2 Data set

In order to run the continuous evaluation activities, we used the experimental collections developed in the CHiC 2011 [Gäde et al., 2011] and CHiC 2012 [Petras et al., 2012] evaluation labs, which correspond to the “Unlocking culture” use case of PROMISE and make use of real data coming from Europeana.

3.2.1 Collections

In March 2012, the complete Europeana data index was downloaded for collection preparation. The Europeana index as used in Europeana’s Solr search portal contained 23,300,932 documents with a size of 132 GB.

Europeana data consists of metadata records describing digital representations of cultural heritage objects, e.g. the scanned version of a manuscript, an image of a painting of sculpture or an audio or video recording. Roughly 62% of the metadata records describe images, 35% describe text, 2% describe audio and 1% video recordings. The metadata contains title and description data, media type and chronological data as well as provider information. For ca. 30% of the records, content-related enrichment keywords were added automatically by Europeana.

The original Europeana index contained fields from different schemas: Simple Dublin Core, e.g. dc_title, dc_description, Qualified Dublin Core, e.g. dcterms_provenance, dcterms_spatial and Europeana Semantic Elements, e.g. europena_type, europeana_isShownAt. On top of these schema-related fields, there were additional fields used internally in the Lucene index to improve search performance or to support specific application functionalities.

These fields were removed from the data collection and the index data was wrapped in a special XML format. The whole collection was then divided into 14 subcollections according to the language of the content provider of the record (which usually indicates the language of the metadata record). If all the provider languages had been used, the number of subcollections would have reached 30. Thus, in order to reduce this amount, a threshold was set: all the languages with less than 100,000 documents were grouped together under the name “Other”.

The resultant 14 subcollections are listed in Table 3.

Language	Sound	Text	Image	Video	Total
German	23,370	664,816	3,169,122	8,372	**3,865,680**
French	13,051	1,080,176	2,439,767	102,394	**3,635,388**
Swedish	1	1,029,834	1,329,593	622	**2,360,050**
Italian	21,056	85,644	1,991,227	22,132	**2,120,059**
Spanish	1,036	1,741,837	208,061	2,190	**1,953,124**
Norwegian	14,576	207,442	1,335,247	555	**1,557,820**
Dutch	324	60,705	1,187,256	2,742	**1,251,027**
English	5,169	45,821	1,049,622	6,564	**1,107,176**
Polish	230	975,818	117,075	582	**1,093,705**
Finnish	473	653,427	145,703	699	**800,302**
Slovenian	112	195,871	50,248	721	**246,952**
Greek	0	127,369	67,546	2,456	**197,371**
Hungarian	34	14,134	107,603	0	**121,771**
Others	375,730	1,488,687	1,106,220	19,870	**2,990,507**
Total	**455,162**	**8,371,581**	**14,304,289**	**169,899**	**23,300,932**

Table 3: CHiC Collections by language and media type.

Figure 10 shows an extract example record from the Europeana CHiC collection.

```
<ims:metadata
ims:identifier="http://www.europeana.eu/resolve/record/10105/5E1618B
FAF072B8953B30701A6A6C3BB655ACF9D"
ims:namespace="http://www.europeana.eu/" ims:language="eng">
<ims:fields>
<dc:identifier>Orn.0240</dc:identifier>
<dc:subject>Tachymarptis melba</dc:subject>
<dc:title>Rundun Zaqqu Bajda (Orn.0240)</dc:title>
<dc:title>Alpine Swift (Orn.0240)</dc:title>
<dc:type>mounted specimen</dc:type>
<europeana:country>malta</europeana:country>
<europeana:dataProvider>Heritage Malta</europeana:dataProvider>
<europeana:isShownAt>http://www.heritagemalta.org/sterna/orn.php?id=
0240</europeana:isShownAt>
<europeana:language>en</europeana:language>
<europeana:provider>STERNA</europeana:provider>
<europeana:type>IMAGE</europeana:type>
<europeana:uri>http://www.europeana.eu/resolve/record/10105/5E1618BF
AF072B8953B30701A6A6C3BB655ACF9D</europeana:uri>
</ims:fields>
</ims:metadata>
```

Figure 10: Europeana CHiC Collection Sample Record.

3.2.2 Topics

For all experiments, original user queries were extracted from Europeana query logs. From all user search sessions in August 2010, those queries were extracted that resulted in a user viewing the complete object (in order to ensure that the session contained more than one user-system interaction). The queries were then further filtered to not include wildcards or automatically generated content (for example by Europeana features).

Over 500 queries were then annotated according to their query category, i.e. topical, personal name, geographical name, work title or other. Queries could be either in the English language or ambiguous in language but would also appear in English. Ambiguous queries could include personal or location names that do not change across languages, e.g. William Shakespeare.

For CHiC, 50 queries were selected that covered a wide range of topics and represented a distribution of query categories that was found in a previous study [Stiller et al., 2010]. For later relevance assessments, descriptions of the underlying information need were added, but were not admissible for information retrieval. The underlying information need for a query can be ambiguous, if the intention of the query is not clear. In this case, the research

group discussed the query and agreed on the most likely information need. Figure 11 shows an example of an English query.

```
<topic lang="en">
<identifier>CHIC-004</identifier>
<title>silent film</title>
<description>documents on the history of silent film, silent film videos, biographies of actors and directors, characteristics of silent film and decline of this genre</description>
</topic>
```

Figure 11: CHiC English Example Query.

3.3 Running prototype

Figure 12 shows the homepage of the "Continuous Evaluation Manager" portlet, accessible in the PROMISE evaluation infrastructure. It allows the user to select the system which she/he wants to evaluate. At the moment, only the continuous evaluation of Europeana is implemented. Nevertheless, this portlet is realized in a modular way which allows, in the future, to plug continuous evaluation components for other systems of interest.

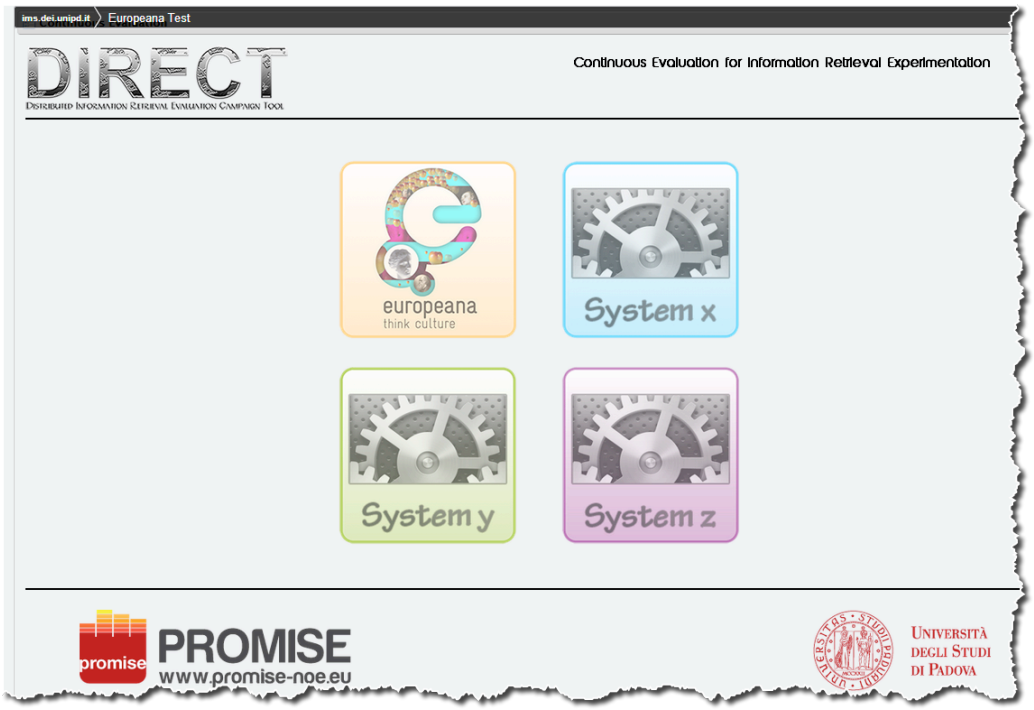

Figure 12: Homepage of the continuous evaluation manager.

Once the user has selected the system to evaluate by pressing the appropriate button in the homepage, she/he has to setup the experiment, as shown in Figure 13. The following information has to be provided:

- the task which will be used for fetching the topics;

- the user who owns the experiment;
- the identifier of the experiment;
- information about the scope of the experiment (private, shared, public) and, in the case of shared experiments, the access permissions to it;
- a description explaining the experiment.
- the topic fields to be used for creating the query to be sent to the system under evaluation.

Once all the needed information are provided and the user presses the "Create" button, a summary page is shown, as in Figure 14, which also reports the content of actual topic which will be used in the experiment.

Figure 13: Continuous evaluation manager – experiment setup (1/2).

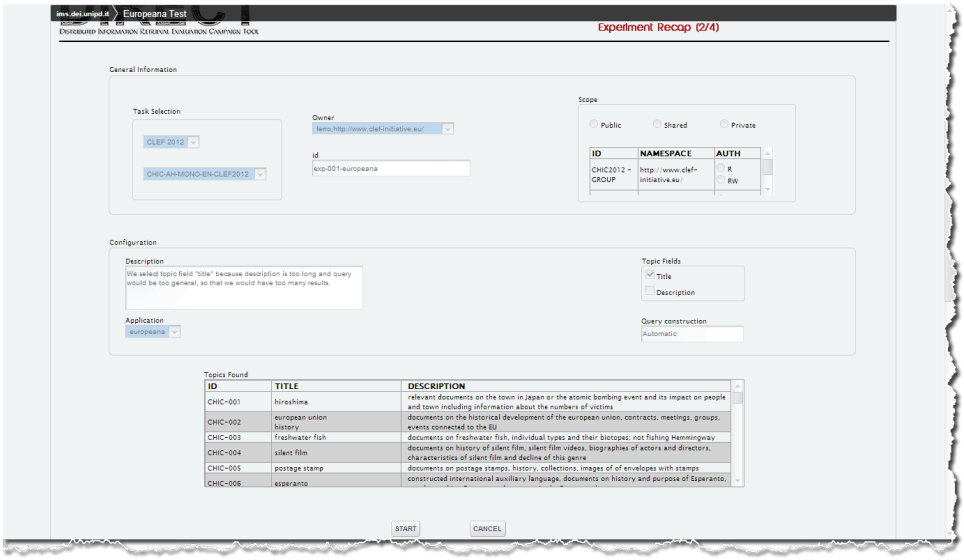

Figure 14: Continuous evaluation manager – experiment setup (2/2).

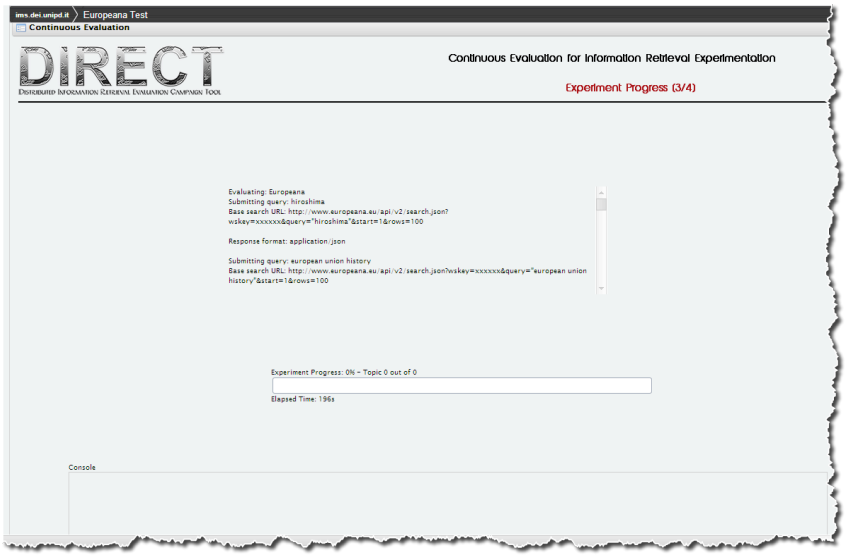

Figure 15: Continuous evaluation manager – experiment progress.

When the user presses the “Start” button, the experiment is initiated the queries are sent to Europeana and the results are fetched by the continuous evaluation manager, as shown in Figure 15. A progress bar and a log inform the user about the status of the experiment, the time elapsed so far, and its completion rate.

Once all the topics have been submitted, a summary page is shown, as displayed in Figure 16, which recaps all the general information about the experiment introduced in the experiment setup phase and reports, for each topic, the number of retrieved results as well as the elapsed time and any error condition which may have occurred.

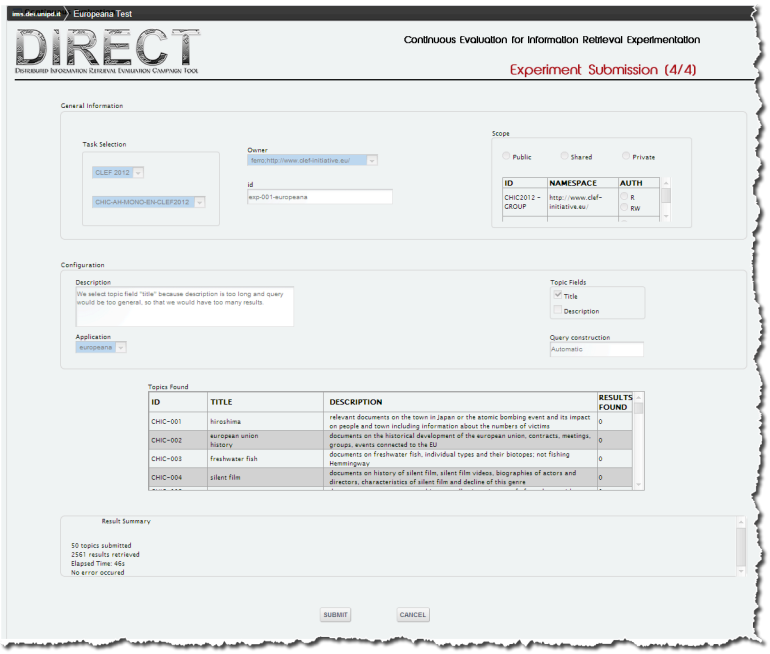

Figure 16: Continuous evaluation manager – experiment submission.

4 Lessons from other continuous evaluation activities

Here we report on other continuous evaluation activities that we have pursued within PROMISE that have so far not made use of the DIRECT infrastructure. The collection of continuous evaluation activities on which we report below, is a diverse collection, which reflects the complexity of the evaluation problem. To situate the continuous evaluation activities in the broader evaluation landscape, the following table is useful:

Offline evaluation	User studies	Online evaluation
Pseudo test collections	Exploratory search	Click models
	Aggregated search	Interleaving
	Black box evaluation	Simulations

4.1 Offline evaluation

Without doubt, one of the great advantages of offline evaluation in the Cranfield tradition is that the experimental condition is fixed: the same queries are being used, the same colelction, and the same relevance judgements. This makes evaluations reproducible and keeps experimenters "honest:" by experimenting on the same set of queries and judgements, we can can better understand how system one system is better than another. Some of the big disavantages offline evaluation are that human assessors that judge documents relevant/non-relevant are expensive, that the assessors used are rarely the users based on whose information needs test queries are produced and, hence, that judgements are made "out of context." Moreover, the offline evaluation paradigm assumes that relevance is the same for every user.

The other two evaluation paradigms (user studies and online evaluation) overcome some of these limitations, but recent years have also witnessed extensive efforts aimed at overcoming the limitations *within* the offline paradigm. Within PROMISE we have been particularly focused on so-called pseudo text collections as a why of addressing two downsides of the offline evaluation paradigm: the effort involved and the lack of diversity in typical offline test collections.

4.1.1 Pseudo test collection generation

Within PROMISE we have been investigating the generation of so-called pseudo test collections. Pseudo test collections (PTCs) are like standard Cranfield style test collections in that they consist of a set of queries together with relevance assessments. However, the creation process is automated in the case of PTCs. We generate both queries and relevance assessments. We report on our PTC generation methods in deliverable D4.3 and in [Berendsen et al., 2012, 2013]. In [Berendsen et al., 2012] we generate PTCs for use in the digital libraries domain. In [Berendsen et al., 2013], we generate PTCs for use in microblog search. Since PTCs are generated automatically, they can be continuously updated with

speed and ease as a document collections expands. Generating queries as well as relevance assessments allows the creation of a test collection that potentially assesses the retrieval of everything that the collection has to offer, see e.g., [Azzopardi et al, 2011].

There are several ways in which one might use a PTC:

- *Obtain an estimate of the performance that a retrieval algorithm would obtain on a TREC-style test collection, in an absolute sense*. This was analyzed in [Azzopardi et al, 2007] and found to be difficult.
- *Obtain an estimate of the performance a set of retrieval algorithms would obtain on a TREC-style test collection, relative to each other*. This is normally done by correlating system rankings. Among others, this was studied in [Beitzel et al, 2003; Huurnink et al., 2010; Berendsen et al., 2012] and also found to be quite difficult. Beitzel et al. [2003] obtain reasonable correlations on a set of navigational queries, Huurnink et al. [2010] get reasonable correlations on a set of known-item queries, and Berendsen et al. [2012] get mixed results for a set of informational queries. Note that in these studies both queries and relevance assessments are generated automatically. One of the things that complicate the matter in this case is that we are comparing system performance on different sets of queries. Even in different TREC-style test collections that use the same document collections, relative system performance may vary. In addition, both query generation and relevance assessment generation are hard processes, even for people. Generating the latter depends on the first, or vice-versa. Automating both of these processes at once is therefore very ambitious indeed.

  In the case that the queries are already given, and annotators are available to produce a limited amount of relevance assessments, research has shown that is quite feasible to obtain system rankings on a PTC that highly correlate with system rankings on a TREC-style test collection, e.g., [Carterette et al, 2006; Rajput et al, 2012]. In the case that the queries are already given, but no manual relevance assessments are available, there is a line of work that aims to automatically evaluate a given set of runs by combining these rankings in some way to produce a set of pseudo-judgments, e.g., [Soboroff et al., 2001; Wu & Crestani, 2003]. This leads to significant but somewhat weaker correlations in system ranking.
- *Train or tune a retrieval algorithm, to optimize its performance on a TREC-style test collection.* This was done in [Asadi et al, 2011] and [Berendsen et al, 2012, 2013]. In this line of work the TREC-style creation process of test collections is not challenged. Rather, the aim is to generate training material to optimize performance on these TREC-style test collections. Modern retrieval algorithms come with many free parameters. Tuning these parameters typically requires a great deal of diverse training material, and PTCs are a natural alternative to look at for this. In this line of work, PTCs as training material are compared to TREC-style collections as training material. Results in both studies show that using PTCs in this way is feasible.

  Performance when training on PTCs is often not statistically different from performance when training on TREC-style test collections. In some cases even performance increases are seen. This leads to the conclusion that in settings where hiring qualified manual assessors is impractical or too expensive, generating PTCs is an alternative worthy of serious consideration. Again, as document collections expand, PTCs can be

expanded with ease, and retrieval models can be adapted by training them again on the freshest PTC.

Zooming out, we have sufficient ground to claim that PTCs offer a viable alternative to traditional editorial test collections when it comes to training and tuning rankers. While there is little added value in incorporating PTC-generation algorithms into DIRECT, it does make good sense to use DIRECT for storing the settings and evaluation outcomes of PTC-generation methods. Because of steps in the random character of the PTC-generation process, care should be taken that training and tuning runs are repeated "sufficiently often;" DIRECT can potentially play a key role in understanding the variance between different training and tuning runs.

4.2 User studies

In comparison to offline evaluation, one of the big advantages of user studies is that they provide us with very detailed data about users' reactions to systems. In reality, a search is done to accomplish a higher-level task; in user studies, this task can be manipulated and studied; in other words, the experimental 'starting-point' need not be the query but a more complex position in what one might call an information game.

Within PROMISE, we have been conducting a number of user studies, both for their own sake, i.e., to assess an information retrieval algorithm or system, and to help us understand the possible role of Direct in support of user studies. We start with a report on two traditional user studies and conclude with a brief report on a black box application, that is half way between a traditional user study and online evaluation.

4.2.1 Two examples

During the PROMISE project the field of digital humanities has grown in attention and importance. After the exact sciences, such as astronomy and physics, and the live and pharmaceutical sciences, the humanities are the next in line to turn into a data-driven science, now that large-scale heritage and research collections have become available. With this development come unique information needs. We piggybackked with two user studies aimed at understanding the relative effectiveness of both traditional and special-purpose retrieval technology for digital humanities scholars—the aim for PROMISE being the identify possible extensions needed for DIRECT in order for it to be useful in support of user studies. We briefly report on those two user studies [Bron et al., 2012, 2013] and then zoom out to discuss the broader system implications for Direct.

The two user studies that we use as a test case are both centered around the research practice of media studies researchers. Media studies concerns the study of production, content, and/or reception of various types of media. Today's continuous production and storage of media is changing the way media studies researchers work and requires the development of new search models and tools. In the first user study [Bron et al., 2012], we have investigated the research cycle of media studies researchers and have found that it is an iterative process consisting of several search processes in which data is gathered and the research question is refined, as illustrated in Figure 17.

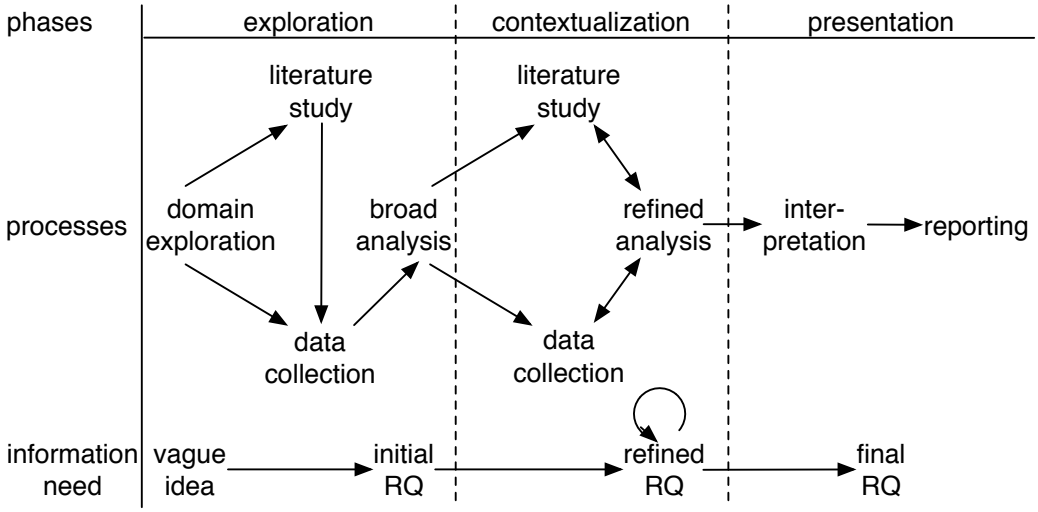

Figure 17: Overview of the phases in the media studies research cycle with assoicated search processes and changes in the research questions (RQ). Arrows indicate possible sequences.

Changes in the media studies researchers' research question trigger new data gathering processes. Based on these outcomes we have proposed a subjunctive exploratory search interface to support media studies researchers in refining their research question in an earlier stage of their research. To assess the proposed environment and its value for media studies researchers, performed a user study. We found that with the subjunctive interface media studies researchers are able to formulate more queries and bookmark more diverse documents compared to a standard exploratory search interface. In a qualitative analysis of the research questions formulated by media studies researchers we have found evidence to suggest that the influence of the subjunctive interface is predominantly on the scope of the research question. Specificly, users of the subjunctive interface incorporate more views on a topic in their research question than users of the standard exploratory search interface. We have observed no advantage for other types of defining the scope as visualizations in both interfaces enable spotting trends in the data. In terms of usability, media studies researchers report that the subjunctive interface is intuitive and not difficult to use, suggesting that the additional complexity in terms of features in the subjunctive interface does not reduce its usability.

In our second user study, we focused on aggregated search facilities, again for media studies researchers. Aggregated search interfaces provide users with an overview of results from various sources. Two general types of display exist: tabbed, with access to each source in a separate tab, and blended, which combines multiple sources into a single result page. Multisession search tasks, e.g., a research project, consist of multiple stages, each with its own sub-tasks. Several factors involved in multi-session search tasks have been found to influence user search behavior. We investigated whether user preference for source presentation changes during a multi-session search task. The dynamic nature of multi-session search tasks makes the design of a controlled experiment a non-trivial challenge. We adopted a methodology based on triangulation and conduct two types of observational study: a longitudinal study and a laboratory study. In the longitudinal study we follow the use of tabbed and blended displays by 25 students during a project. We found that while a tabbed display is used more than a blended display, subjects repeatedly switch

between displays during the project. Use of the tabbed display is motivated by a need to zoom in on a specific source, while the blended display is used to explore available material across sources whenever the information need changes. In a laboratory study 44 students completed a multi-session search task composed of three sub-tasks, the first with a tabbed display, the second and third with blended displays. The tasks were manipulated by either providing three tasks about the same topic or about three different topics. We found that a stable information need over multiple sub-tasks negatively influences perceived usability of the blended displays, while we do not find an influence when the information need changes.

What are the implications of these two user studies for the future development of DIRECT? If it is to support user studies, a number of facilities are desirable or even essential. First, an environment is needed to log all potentially interesting actions and interactions, at many levels of granularity (query, session, task, user). Second, facilities are needed to group sequences of actions and interactions into behavioral patterns. And third, in parallel to quantitative analyses, qualitative analyses need to be facilitated too, with annotation capabilities so as to be able to link observations to interview data. In addition, dashboard functionality is needed to be able to monitor logging activities. DIRECT is ready to accommodate information about users, projects (which can be modelled as a subclass of experiments), behavioural patterns (which can be modelled as subclass of experiment items) and annotations, while the semistructured and streaming nature of event data may be better catered for using a NoSQL solution. An initial design of a suitable architecture is included in Figure 18 below; while not formally a deliverable of the PROMISE project, the logging facility has been completed and is currently being deployed in a small number of test projects; integration with DIRECT is part of future work.

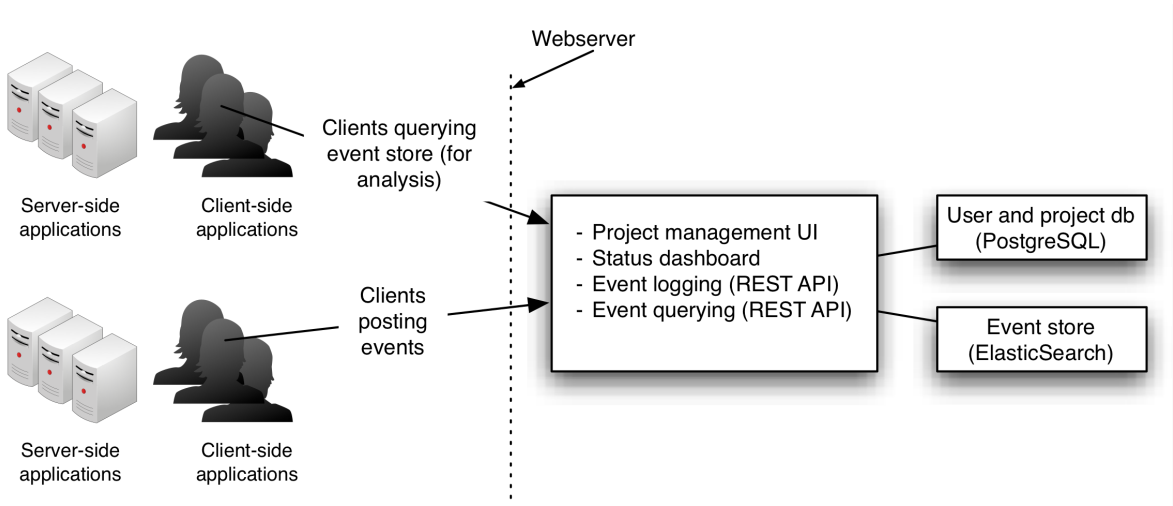

Figure 18: Initial design of a logging architecture for user studies.

4.2.2 Continuous use of black box application evaluation

In the PROMISE deliverable D4.2 [Rietberger et al. 2012] we introduced and validated black box application evaluation as a methodology to measure an estimate of user perception of an operational application. The methodology is based on a large number of independent tests, the results of which are aggregated. The tests are executed according to scripts, which model prototypical user behaviour and assumed user preferences. Several usage scenarios for the methodology have been described, namely: evaluation as a campaign, comparison and monitoring. The monitoring case is a prime candidate for continuous evaluation.

The black box application evaluation monitoring scenario is designed to observe differences in the evaluation results over time. Differences are expected to occur whenever application features, configuration or interfacing systems change, but may also—perhaps more critically—occur because of changing content and learning functions in application components. Since all of these changes are likely to happen in operational settings, monitoring can provide valuable insight to company stakeholders about the stability of their search application's quality. A properly set up monitoring environment may ultimately even provide an application "health meter", e.g., displaying green, yellow and red status according to a set of rules that need to be defined by a company. This requires tests to be run automatically, which until now has not been examined as part of the black box application evaluation methodology. Test scripts in the methodology are designed to avoid "creativity" of testers. The results obtained should be representative of the "prototypical user" that is modelled, not of the individual tester. Therefore, automation is a viable approach for carrying out at least a subset of the test script, and can be simply a matter of overcoming technological difficulties in some cases. On the other hand, there are some tests where automation yields no practical results.

When validating the methodology, we only tested publicly accessible web-based search applications. Using automation frameworks (e.g. Selenium[2] for web applications), such search applications can be instrumented. Search boxes and result lists need to be identified to allow automated query issuing and results checking. The following automation considerations assume the usage of an automation framework to perform tests.

Feature Tests

When only single features are tested, an automated test is set up to check if a particular feature is still present in its current form. Changes to the applications might then cause tests to fail if they are not adapted accordingly. In both cases of pro-active adaptation and retro-active correction any tested feature is being checked rigorously and application quality and functionality is assured. The generic approach for feature tests operating on actual queries is straight-forward. A number of queries are scripted which are expected to yield results with specific characteristics. Much like unit tests in software engineering, the expected results are checked using a formalized description. The following is an excerpt of the stemming test in the deliverable D4.2, which is used as an example to show a specific automation approach:

[2] http://docs.seleniumhq.org/

Assumption

Users enter queries based on their intent as a set of key words or as a full question. If a term is entered as a noun, adjective, verb or adverbially may differ from session to session while the intent may not. Stemming counteracts by reducing different grammatical word forms to single stem forms, thereby increasing the probability of matching the intended word irrespectively of its form.

Test

1. Enter a few single term queries using singular words, plurals, different verbal and adjective forms.
2. Score success (1) if different forms of the same word in a query return the same results. Otherwise score failure (0).

To automate this stemming test a range of word forms, which are expected to be stemmed, is configured as queries. Example queries for English:

Query	Expected terms in result documents
"new"	"new", "news", "newly"
"stained"	"stained", "staining", "stain", "stains"
"financially"	"financially", "finance", "finances", "financial"

Queries like these are issued to applications under scrutiny. The result list is then scanned for derivatives of the query terms' stems, as shown in the examples above. Since results are bound to change as the application content does, a sufficiently large number of diverse queries should be used to compensate for false negatives. A success rate above a threshold of confidence is then considered a successful test run.

Content-Based Tests

Tests based on application content are much harder to automate in a consistent manner. Content is expected to change in varying intervals and any application change, especially changing locations of content (even if only in terms of layout), may break automated tests.

As an example we consider testing index freshness, wherein the objective is to assess if new content of an application is indexed in a suitable time frame. The test script asks to first identify a new document (usually from a news section) and then search for it to check if it has been indexed already. To automate this test, a suitable application section has to be identified, which can be checked for new documents. A query for the document is issued using a characteristic phrase (e.g. the document's title). The result list's top ten entries are then checked if the new document is retrievable. This of course only works if there actually are new documents. The test script takes this into account by having a test aborted if there are no new documents. Depending on how regularly the evaluation is repeated, there may be many aborts, which need to be interpreted separately so as to avoid unnecessarily skewing evaluation results.

Impractical Tests

Some tests are impractical to automate, as they require intellectual assessment of usability, aesthetics, large result sets or other similar aspects.

Testing the general result list presentation, for example, requires human assessment and is therefore entirely unsuitable for automation. Consider this description of the associated test:

Test

Score according to your impression of the result list presentation (0, 1, 2):

1. 2 for good (useful layout, visually pleasing)
2. 1 for sufficient / decent (practical, functional, basic)
3. 0 for bad (unwieldy layout, cluttered, confusing)

Human testers can provide a consistent assessment of the qualities mentioned in the test script while it is hard for machines. Although heuristics for usability testing have been described, e.g., in [Nielsen & Molich 1990], these still rely on human assessors to execute and aesthetics are not covered, either. Two main points remain in the continuous use of black box application evaluation: Full description of automation of tests and possible integration of automation facilities in the DIRECT evaluation infrastructure. As part of our on-going work to revise the methodology in D4.2, we are going to investigate and document automation procedures for previously described and possibly also new tests. The descriptions will be expanded with suggestions for repetition intervals, limitations and high-level implementation guides. Revisiting the stemming example, an accordingly expanded criterion/test description would then look as follows:[3]

Stemming

Assumption

[…]

Irregularity

[…]

Root Cause

[…]

Test

[…]

Repetition Interval

- Does not require regular repetition because it is not content-dependent
- Repeat when changes are made to the IR system, e.g. introduction of another language or changes in the indexing process

[3] Parts irrelevant to the clarity of the example or which have been previously shown are omitted. The interested reader is referred to the PROMISE deliverable D4.2 [Rietberger et al. 2012] for fully detailed descriptions.

Automation

1. Define a set of one-word queries using plurals, adjectives and adverbs. Make sure the chosen words occur regularly within the application's data.
2. For each query, define which words based on the common stem should match.
3. Let each query be issued by your automation framework.
4. Check the results for each query against the previously defined expected word list.
5. Score success (1) if 2/3 or more of the expected words could be found in returned documents. Score failure (0) otherwise.

Additionally, we plan to investigate the possibility to integrate automation in Direct. If (web-based) application instrumentation for the purposes of black box evaluation is made possible, the infrastructure can be used by a variety of organizations to regularly and automatically evaluate their applications with much lower effort as required when performing evaluation on-site.

4.3 Online evaluation

We turn to the third type of evaluation considered for continuous evaluation: online evaluation. The big advantage of online evaluation is that system usage is naturalistic; users are situated in their natural context and often do not know that a test is being conducted. Moreover, evaluation can include lots of users. A big disadvantage, especially in an academic environment is that online evaluation requires a service with lots of users (enough of them to potential hurt performance for some). This is often referred to as the "cold-start problem." As online evaluation only deals with implicit feedback (clicks, forwards, saves, etc.), a good understanding on how different implicit feedback signals predict positive and negative user experiences is essential. Finally, experiments conducted as part of online evaluation are difficult to repeat.

4.3.1 Interleaving comparison methods

In interleaving comparison methods rankers are assessed using implicit feedback from actual users, such as click behavior, touch behavior, query reformulations, etc. A common approach is to use interleaved comparison methods [Chapelle et al., 2013; Chuklin et al., 2013b; Hofmann et al., 2013b; Joachims, 2003], in which the document lists proposed by two candidate rankers for a given query are interleaved and the resulting list presented to the user, whose clicks are used to infer a noisy preference for one ranker over the other. Recently, interleaving methods have been successfully applied in large-scale settings [Chapelle et al., 2013; Chuklin et al., 2013b]. In comparison to absolute click metrics typically used in A/B testing, interleaved comparison methods reduce variance (briefly, this is because they perform within-subject as opposed to between-subject comparisons), and make different assumptions about how clicks should be interpreted (as relative, as opposed to absolute feedback).

Outcomes of interleaved comparisons can be stored in Direct (which of two rankers wins a comparison), thus in principle facilitating the re-use of historical data. Until recently, it was not clear how interleaved comparison methods could reuse historical data. However, the recently developed probabilistic interleave method bridges this gap [Hofmann et al., 2011; 2013c]. Probabilistic interleave is based on a probabilistic interpretation of interleaved comparisons, which allows it to infer comparison outcomes using data from arbitrary result lists, even if they were obtained in comparisons of rankers different from the current target rankers. In probabilistic interleave, the interleaved document list is constructed, not from fixed lists but from softmax functions that depend on the query. The use of softmax functions ensures that every document has a non-zero probability of being selected by each ranker. As a result, the distribution of credit accumulated for clicks is smoothed, based on the relative rank of the document in the original result lists.

Probabilistic interleave has recently been generalized in a number of directions. For instance, Chuklin et al. [2013b] demonstrate how it can compare search engine result pages that contain grouped vertical documents. In addition, the probabilistic nature of probabilistic interleave makes possible the reuse of historical data via importance sampling [Hofmann et al., 2013a]. So long as the distribution under which the historical data was gathered is known, even if the data was not gathered using probabilistic interleave, importance weights can be computed that enable probabilistic interleave to compute an unbiased and consistent estimate of the relative quality of the interleaved rankers.

In sum, when running online evaluation experiments based on probabilistic interleave, there is a clear and natural role for future versions of the Direct infrastructure: to store and serve historical performance data and to maintain scores of ongoing evaluations using traditional offline and online metrics.

4.3.2 Click models

One of the main advantages of online evaluation schemes is that they are user-based and, as a result, often assumed to give us more realistic insights into the real system quality. Interleaving experiments are now widely being used by large commercial search engines like Bing, Yahoo! and Yandex as well as studied in academia [Chapelle et al., 2013; Hofmann et al., 2013abc]. However, they are harder to reproduce than offline measurements, whereas in the traditional Cranfield approach one can re-use the same set of judged documents to evaluate any ranking. This makes the use of offline editor-based evaluation methods unavoidable during the early development phase of ranking algorithms. One should take care, however, that the resulting editor-based measurements agree with the outcomes of online experiments—online comparison is often used as the final validation step before releasing a new version of a ranking algorithm.

In order to bring the two evaluation approaches closer to each other, we propose a method for building an offline information retrieval (IR) metric from a user click model. Click models, probabilistic models of the behavior of search engine users, have been studied extensively by the information retrieval community during the last five years. The main purpose of predicting clicks, as seen in previous works, is: (1) modeling user behavior when real users are not available (see, e.g., [Hofmann et al., 2013a] and below); (2) improving ranking using relevance inferred from clicks.

Apart from click events, a click model usually has hidden variables corresponding to events such as “he user examined the snippet of the k-th document.” These hidden variables are often used to gain deeper insights into users' behavior. For example, Chapelle and Zhang [2009] used a click model to predict relevance and train a ranking function and in [Dupret and Piwowarski, 2008] the parameters of the click model were analysed to explain how previous user clicks influence future clicks.

As part of our work on T3.4, we have investigated the hypothesis that click models can also be turned into offline metrics and the resulting click model-based metrics should be closely tied to the user and hence should better correlate with online measurements than traditional offline metrics. There is a growing trend to ground offline metrics in a user model and that is exactly what click modeling does—trying to propose a better user model.

In [Chuklin et al., 2013a] we have proposed a framework of click model-based metrics to build an offline evaluation measure on top of any click model. Our main findings are as follows. Click model-based metrics generally differ from traditional offline metrics, while they are quite similar to each other. Moreover, utility-based metrics are significantly different from effort-based metrics in terms of system ranking. All click model-based metrics generally show high agreement with the outcomes of online interleaving experiments and relatively high agreement with absolute click measures. However, correlation with absolute metrics is low for all offline metrics (both traditional and click model-based) compared to the correlation with interleaving outcomes. Unjudged documents may decrease correlation values with interleaving outcomes but by using thresholds we can overcome this issue for click model-based metrics. Condensation and thresholding of offline metrics are effective ways of stabilizing correlations with interleaving outcomes in the presence of unjudged documents.

The main implications of these outcomes for future versions of the DIRECT infrastructure concern the use of a user model. While click models can be estimated with existing open source solutions[4], the use of such models as an essential ingredient of offline metrics has implications for the way such metrics are currently implemented in DIRECT. Thanks to the work carried out for the semantic enrichment (D3.6) and bibliometrics (D6.4), the DIRECT data model has been extended (D3.5) in order to allow us to attach, with a confidence score, profiles to users and the profiles themselves can be linked together with semantic relations in order to favour navigation and further processing. Moreover, metrics, statistics, and visualizations can be computed and attached to both users and profiles. These features, which are already present, can be exploited to implement the above mentioned user models and the computation of offline metrics over them.

4.3.3 Simulations

Evaluating the ability of a retrieval algorithm to maximize cumulative performance in an online information retrieval setting poses unique experimental challenges. The most realistic experimental setup—in a live setting with actual users—is risky because users may get frustrated with bad search results. The typical TREC-like setup used in supervised learning to rank for information retrieval is not sufficient for assessing retrieval approaches that rely

[4] See, e.g., https://github.com/varepsilon/clickmodels for the implementation of probabilistic inference for a number of popular click models.

on user behavior. Of course, there is a second challenge to overcome: we simply may not have access to a live system with a substantial user population.

To address these challenges, we have proposed an evaluation setup that simulates user interactions [Hofmann et al., 2011; 2013c]. This setup combines data sets with explicit relevance judgments that are typically used for supervised learning to rank with recently developed click models. Given a data set with queries and explicit relevance judgments, interactions between the retrieval system and the user are simulated; cf., the box labeled "user/environment" in Figure 19 below.

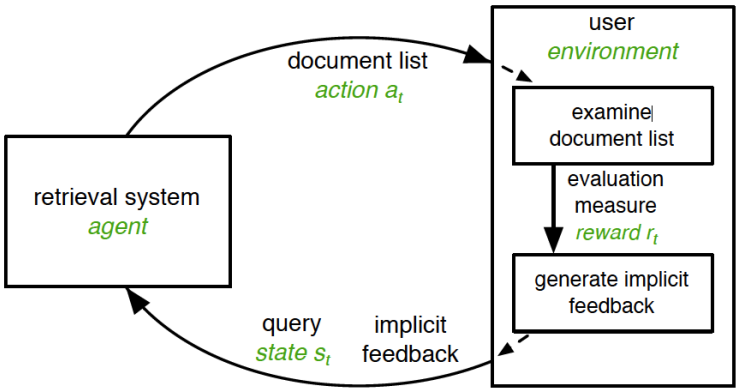

Figure 19: Online learning to rank formulated as a reinforcement learning problem.

Submitting a query is simulated by random sampling from the set of queries. After the system has generated a result list for the query, feedback is generated using a click model and the relevance judgments provided with the data set. Note that the explicit judgments from the data set are not directly shown to the retrieval system but are used to simulate the user feedback and measure cumulative performance.

We have used this evaluation setup in two scenarios, an online evaluation scenario and an online learning to rank scenario. Online evaluation is both a goal in itself and a subproblem of online learning to rank. By itself, it allows the assessment of rankers that were tuned e.g., manually, or using offline learning to rank, using real search engine traffic. As a subproblem of online learning to rank, online evaluation provides the mechanism for inferring feedback for learning. The goal of our online evaluation scenario was to assess the efficiency of interleaved comparison methods when comparing different rankers, and therefore we measured how much interaction data a method needs to distinguish two rankers [Hofmann et al., 2013c].

Using simulated evaluations naturally has limitations, but allows us to systematically investigate online evaluation and online learning to rank methods, without the risks associated with experiments involving real users. For instance, we can show how learning methods behave under different assumptions about user behavior, but to what degree these assumptions apply in specific practical settings still needs to be studied in more detail.

The implications of these outcomes for future editions of the DIRECT infrastructure are as follows. First, parameters of the simulators need to be stored. Second, as the simulators will be (partly) based on stochastic components, a variance analysis and similar facilities are essential. Third, there needs to be a procedure for sampling from a standard annotated test collection—a minor change given that DIRECT already deals with a large array of retrieval test collections.

5 Conclusions

In this deliverable we have explored the potential of the DIRECT evaluation infrastructure to bringing automation into the evaluation process, by promoting its "continuous" use outside the annual evaluation cycles carried out in WP6. The expanded use of DIRECT promises to greatly increase the gathered evaluation knowledge base.

We examined the use of DIRECT to support the evaluation of Europeana components on CLEF datasets using during 2012 and 2013. Based on the positive outcomes of this experience, we decided to perform an additional step and introduce the possibility of continuously evaluating the production Europeana platform.

Next, we discussed continuous evaluation experiments, and studies aimed at components meant to facilitate such experiments, along three lines: offline evaluation ("the Cranfield paradigm"), user studies, and online evaluation. In each case, we reported on innovative methodological work carried within PROMISE, work that has a broad range of implications for future iterations of the DIRECT evaluation infrastructure. Lessons were formulated in one new functionality (e.g., support for qualitative analysis, logging of interactions), new types of concept (e.g., tasks and not just queries or sessions), new types of source for ground truth (e.g., pseudo test collections), new types of metric (e.g., cumulative and model-based metrics) and new types of use (e.g., to monitor the progress of user studies or of online evaluations). While it would be naïve to assume that all of these possible extensions to the Direct infrastructure could be rolled out together, for instance because of the costs involved, there is no conceptual reason that excludes this possibility.

References

[Agirre et al., 2009] Agirre, E., Di Nunzio, G. M., Ferro, N., Mandl, T., Peters, C. (2009). CLEF 2008: Ad Hoc Track Overview. In Peters, C., Deselaers, T., Ferro, N., Gonzalo, J., Jones, G. J. F., Kurimo, M., Mandl, T., Peñas, A., editors, *Evaluating Systems for Multilingual and Multimodal Information Access: Ninth Workshop of the Cross–Language Evaluation Forum (CLEF 2008). Revised Selected Papers*, pages 15–37. LNCS 5706, Springer, Heidelberg, Germany.

[Agost et al., 2011a] Agosti, M., Bosca, A., Crivellari, F., Deambrosis, G. Di Nunzio, G. M., Dussin, M., Ferro, N., Gäde, M., Petras, V. (2011). *Report on Evaluation of Multilingual Information Access to Europeana*. EuropeanaConnect, contract n. ECP-2008-DILI-528001, http://www.europeanaconnect.eu/documents/Task2_5-EuropeanaConnect-Evaluation_Report.pdf

[Agosti et al., 2011b] Agosti, M., Braschler, M., Di Buccio, E., Dussin, M., Ferro, N., Granato, G. L., Masiero, I., Pianta, E., Santucci, G., Silvello, G., Tino, G. (2011). *Deliverable D3.2 – Specification of the evaluation infrastructure based on user requirements*. PROMISE Network of Excellence, EU 7FP, Contract N. 258191. http://www.promise-noe.eu/documents/10156/fdf43394-0997-4638-9f99-38b2e9c63802.

[Agosti et al., 2011c] Agosti, M., Di Nunzio, G. M., Ferro, N. (2011). *Deliverable D3.1 – Initial prototype of the evaluation infrastructure*. PROMISE Network of Excellence, EU 7FP, Contract N. 258191. http://www.promise-noe.eu/documents/10156/e0df8a3c-388f-40e8-bfbd-04434a393004.

[Agosti et al., 2012] Agosti, M., Di Buccio, E., Ferro, N., Masiero, I., Nicchio, M., Peruzzo, S., Silvello, G. (2012). *Deliverable D3.3 – Prototype of the Evaluation Infrastructure*. PROMISE Network of Excellence, EU 7FP, Contract N. 258191. http://www.promise-noe.eu/documents/10156/ 3783730a-bce3-481b-83df-48e209c6286a.

[Asadi et al, 2011] Asadi, N., Metzler, D., Elsayed, T., Lin, J. J. (2011). Pseudo test collections for learning web search ranking functions. In SIGIR 2011, pp. 1073-1082).

[Azzopardi et al, 2007] Azzopardi, L., de Rijke, M., & Balog, K. (2007). Building simulated queries for known-item topics: an analysis using six european languages. In SIGIR 2007 (pp. 455-462). ACM.

[Azzopardi et al, 2011] Azzopardi, L. (2011). The economics in interactive information retrieval. In SIGIR 2011 (pp. 15-24). ACM.

[Berendsen et al, 2012] Berendsen, R., Tsagkias, M., de Rijke, M., Meij, E. (2012). Generating pseudo test collections for learning to rank scientific articles. In Information Access Evaluation. Multilinguality, Multimodality, and Visual Analytics (pp. 42-53). Springer Berlin Heidelberg.

[Berendsen et al, 2013] Berendsen, R., Tsagkias, M., Weerkamp, W., de Rijke, M. (2013). Pseudo Test Collections for Training and Tuning Microblog Rankers. In SIGIR 2013. ACM.

[Beitzel et al, 2003] Beitzel, S. M., Jensen, E. C., Chowdhury, A., Grossman, D. (2003). Using titles and category names from editor-driven taxonomies for automatic evaluation. In CIKM 2003 (pp. 17-23). ACM.

[Braschler, 2003] Braschler, M. (2003). CLEF 2002 – Overview of Results. In Peters, C., Braschler, M., Gonzalo, J., and Kluck, M., editors, *Advances in Cross-Language Information Retrieval: Third Workshop of the Cross–Language Evaluation Forum (CLEF 2002) Revised Papers*, pages 9–27. LNCS 2785, Springer, Heidelberg, Germany.

[Bron et al., 2012] Bron M., van Gorp J., Nack F., de Rijke M., de Leeuw S. (2012). A Subjunctive Exploratory Search Interface to Support Media Studies Researchers. In SIGIR '12. ACM.

[Bron et al., 2013] Bron M., van Gorp J., Nack F., Baltussen L.B., de Rijke, M. (2013). Aggregated Search Interface Preferences in Multi-Session Search Tasks. In SIGIR'13. ACM.

[Carterette et al, 2006] Carterette, B., Allan, J., Sitaraman, R. (2006). Minimal test collections for retrieval evaluation. In SIGIR 2006 (pp. 268-275). ACM.

[Chapelle et al., 2013] Chapelle, O., Joachims, T., Radlinski, F., Yue, Y. (2013). Large-scale validatino and analysis of interleaved search evaluation. *ACM Transactions on Information Systems* 30(1):1–41.

[Chapelle and Zhang, 2009] Chapelle, O., Zhang, Y. (2009). A dynamic bayesian network click model for web search ranking. In WWW'09. ACM.

[Chuklin et al., 2013a] Chuklin, A., Serdyukov, P., de Rijke, M. (2013a). Click model-based information retrieval metrics. In SIGIR 2013. ACM.

[Chuklin et al., 2013b] Chuklin, A., Schuth, A., Hofmann, K., Serdyukov, P., de Rijke, M. (2013b). Evaluating aggregated search using interleaving. In CIKM 2013. ACM.

[Dupret and Piwowarski, 2008] Dupret, G., Piwowarski, B. (2008). A user browsing model to predict search engine click data from past observations. In SIGIR'08. ACM.

[Ferro and Peters, 2010] Ferro, N., Peters, C. (2010). CLEF 2009 Ad Hoc Track Overview: TEL & Persian Tasks. In Peters, C., Di Nunzio, G. M., Kurimo, M., Mandl, T., Mostefa, D., Peñas, A., and Roda, G., editors, *Multilingual Information Access Evaluation Vol. I Text Retrieval Experiments – Tenth Workshop of the Cross–Language Evaluation Forum (CLEF 2009). Revised Selected Papers*, pages 13–35. LNCS 6241, Springer, Heidelberg, Germany.

[Gäde et al., 2011] Gäde, M., Ferro, N., Lestari Paramita, M. (2011). CHiC 2011 – Cultural Heritage in CLEF: From Use Cases to Evaluation in Practice for Multilingual Information Access to Cultural Heritage. In Petras, V., Forner, P., and Clough, P., editors, *CLEF 2011 Labs and Workshops, Notebook Papers*. MINT srl, Trento, Italy.

[Hofmann et al., 2011] Hofmann, K., Whiteson, S., de Rijke, M. (2011). A probabilistic method for inferring preferences from clicks. In CIKM'11. ACM

[Hofmann et al., 2013a] Hofmann, K., Schuth, A., Whiteson, S., de Rijke, M. (2013a). Reusing historical interaction data for faster online learning to rank for IR. In WSDM'13. ACM.

[Hofmann et al., 2013b] Hofmann, K., Whiteson, S., de Rijke, M. (2013b). Balancing exploration and exploitation in listwise and pairwise online learning to rank for information retrieval. *Information Retrieval* 16(1):63–90.

[Hofmann et al., 2013c] Hofmann, K., Whiteson, S., de Rijke, M. (2013c). fidelity, soundness, and efficiency of interleaved comparison methods. *ACM Transactions on Information Systems 31(4)*.

[Huurnink et al, 2010] Huurnink, B., Hofmann, K., de Rijke, M., Bron, M. (2010). Validating query simulators: An experiment using commercial searches and purchases. In Multilingual and Multimodal Information Access Evaluation (pp. 40-51). Springer Berlin Heidelberg.

[Joachims, 2003] Joachims, T. (2003). Evaluating retrieval performance using clickthrough data. In *Text Mining* (pp. 79–96). Springer.

[Nielsen & Molich 1990] Nielsen, J., Molich, R. (1990). Heuristic evaluation of user interfaces. Proc. ACM CHI'90, 249-256.

[Petras et al., 2012] Petras, V., Ferro, N., Gäde, M., Isaac, A., Kleineberg, M., Masiero, I., Nicchio, M., Stiller, J. (2012). Cultural Heritage in CLEF (CHiC) Overview 2012. In Forner, P., Karlgren, J., and Womser-Hacker, C., editors, CLEF 2012 Labs and Workshops, Notebook Papers. MINT srl, Trento, Italy.

[Rajput et al, 2012] Rajput, S., Ekstrand-Abueg, M., Pavlu, V., Aslam, J. A. (2012). Constructing test collections by inferring document relevance via extracted relevant information. In CIKM 2012 (pp. 145-154). ACM.

[Rietberger et al. 2012] Rietberger, S., Imhof, M., Braschler, M., Berendsen, R., Järvelin, A., Hansen, P., García Seco de Herrera, A., Tsikrika, T., Lupu, M., Petras, V., Gäde, M., Kleineberg, M., Choukri, K. (2012). *Tutorial on Evaluation in the Wild*. PROMISE Deliverable 4.2.

[Soboroff et al, 2001] Soboroff, I., Nicholas, C., Cahan, P. (2001). Ranking retrieval systems without relevance judgments. In SIGIR 2001 (pp. 66-73). ACM.

[Stiller et al., 2010] Stiller, J., Gäde, M., Petras, V. (2010). Ambiguity of Queries and the Challenges for Query Lan-guage Detection. *CLEF 2010 LogCLEF Workshop*. In: Braschler, M., Harman, D., Pianta, E. (eds) CLEF 2010 Labs and Workshops Notebook Papers. Padua, Italy, 22-23 September 2010.

[Tomlinson, 2009] Tomlinson, S. (2009). Sampling Precision to Depth 10000 at CLEF 2008. In Peters, C., Deselaers, T., Ferro, N., Gonzalo, J., Jones, G. J. F., Kurimo, M., Mandl, T., and Peñas, A., editors, *Evaluating Systems for Multilingual and Multimodal Information Access: Ninth Workshop of the Cross–Language Evaluation Forum (CLEF 2008). Revised Selected Papers*, pages 163–169. LNCS 5706, Springer, Heidelberg, Germany.

[Wu & Crestani, 2003] Wu, S., Crestani, F. (2003). Methods for ranking information retrieval systems without relevance judgments. In Proceedings of the 2003 ACM symposium on Applied computing (pp. 811-816). ACM.