Overview of the INEX 2012 Social Book Search Track

Marijn Koolen[1], Gabriella Kazai[2], Jaap Kamps[1], Michael Preminger[3], Antoine Doucet[4], and Monica Landoni[5]

[1] University of Amsterdam, Netherlands
{marijn.koolen,kamps}@uva.nl
[2] Microsoft Research, United Kingdom
v-gabkaz@microsoft.com
[3] Oslo and Akershus University College of Applied Sciences, Norway
michaelp@hioa.no
[4] University of Caen, France
doucet@info.unicaen.fr
[5] University of Lugano
monica.landoni@unisi.ch

Abstract. The goal of the INEX 2012 Social Book Search Track is to evaluate approaches for supporting users in reading, searching, and navigating book metadata and full texts of digitised books as well as associated user-generated content. The investigation is focused around two tasks: 1) the Social Book Search task investigates the complex nature relevance in book search and the role of user information and traditional and user-generated book metadata for retrieval, 2) the Prove It task evaluates focused retrieval approaches for searching pages in books that support or refute a given factual claim. There are two additional tasks that did not run this year. The Structure Extraction task tests automatic techniques for deriving structure from OCR and layout information, and the Active Reading Task aims to explore suitable user interfaces for eBooks enabling reading, annotation, review, and summary across multiple books. We report on the setup and the results of the two search tasks.

1 Introduction

Prompted by the availability of large collections of digitised books, e.g., the Million Book project[6] and the Google Books Library project,[7] the Social Book Search Track[8] was launched in 2007 with the aim to promote research into techniques for supporting users in searching, navigating and reading book metadata and full texts of digitised books. Toward this goal, the track provides opportunities to explore research questions around five areas:

[6] http://www.ulib.org/
[7] http://books.google.com/
[8] Previously known as the Book Track (2007–2010) and the Books and Social Search Track (2011).

– Evaluation methodologies for book search tasks that combine aspects of retrieval and recommendation,
– Information retrieval techniques for dealing with professional and user-generated metadata,
– Information retrieval techniques for searching collections of digitised books,
– Mechanisms to increase accessibility to the contents of digitised books, and
– Users' interactions with eBooks and collections of digitised books.

Based around these main themes, the following four tasks were defined:

1. The *Social Book Search* (SBS) task, framed within the scenario of a user searching a large online book catalogue for a given topic of interest, aims at exploring techniques to deal with both complex information needs of searchers—which go beyond topical relevance and can include aspects such as genre, recency, engagement, interestingness, quality and how well-written it is—and complex information sources including user profiles and personal catalogues, and book descriptions containing both professional metadata and user-generated content.
2. The *Prove It* (PI) task aims to test focused retrieval approaches on collections of books, where users expect to be pointed directly at relevant book parts that may help to confirm or refute a factual claim;
3. The *Structure Extraction* (SE) task aims at evaluating automatic techniques for deriving structure from OCR and building hyperlinked table of contents;
4. The *Active Reading task* (ART) aims to explore suitable user interfaces to read, annotate, review, and summarize multiple books.

In this paper, we report on the setup and the results of each of the two search tasks, SBS and PI, at INEX 2012. First, in Section 2, we give a brief summary of the participating organisations. The SBS task is described in detail in Section 3, and the PI task in Section 4. We close in Section 5 with a summary and plans for INEX 2013.

2 Participating Organisations

A total of 55 organisations registered for the track (compared with 47 in 2011, 82 in 2010, 84 in 2009, 54 in 2008, and 27 in 2007). At the time of writing, we counted 5 active groups (compared with 10 in 2011 and 2010, 16 in 2009, 15 in 2008, and 9 in 2007), see Table 1.[9]

3 The Social Book Search Task

The goal of the Social Book Search (SBS) task is to evaluate the value of professional metadata and user-generated content for book search on the web. Through social media have extended book descriptions far beyond what is traditionally

[9] SE is biennial and will occur again in 2013.

Table 1. Active participants of the INEX 2012 Social Book Search Track, the task they were active in, and number of contributed runs (SBS = Social Book Search, PI = Prove It , SE = Structure Extraction, ART = Active Reading Task)

ID	Institute	Tasks	Runs
4	University of Amsterdam	SBS, PI	2 SBS, 5 PI
5	University of Michigan	PI	6 PI
54	Royal School of Library and Information Science	SBS	6 SBS
62	LIA, University of Avignon	SBS	5 SBS
100	Oslo and Akershus University College of Applied Sciences	SB, PI	4 SBS, – PI

stored in professional catalogues. Not only are books described in the users' own vocabulary, but are also reviewed and discussed online, and added to personal catalogues of individual readers. This additional information is subjective and personal, and allows users to search for books in different ways. Traditional descriptions have formal and subject access points for identification, known-item search and subject search. Yet readers use many more aspects of books to help them decide which book to read next [5], such as how engaging, fun, educational or well-written a book is. This results in a search task that requires a different model than traditional ad hoc search [3].

The SBS task investigates book requests and suggestions from the LibraryThing discussion forums as a way to model book search in a social environment. The discussions in these forums show that readers frequently turn to others to get recommendations and tap into the collective knowledge of a group of readers interested in the same topics.

As a source book descriptions, the INEX Amazon/LibraryThing collection [1] is used, which contains 2.8 million book descriptions from Amazon, enriched with content from LibraryThing. This collection contains both professional metadata and user-generated content. An additional goal of the SBS task is to evaluate the relative value of controlled book metadata, such as classification labels, subject headings and controlled keywords, versus user-generated or social metadata, such as tags, ratings and reviews, for retrieving the most relevant books for a given user request.

The SBS task aims to address the following research questions:

- Can we build reliable and reusable test collections for social book search based on book requests and suggestions from the LibraryThing discussion forums?
- Can we simulate book suggestions with judgements from Mechanical Turk?
- Can user-dependent evidence improve retrieval performance for social book search.
- Can personal, affective aspects of book search relevance be captured by systems that incorporate user-generated content and user profiles?

– What is the relative value of social and controlled book metadata for book search?

3.1 Scenario

The scenario is that of a user turning to Amazon Books and LibraryThing to search for books they want to read, buy or add to their personal catalogue. Both services host large collaborative book catalogues that may be used to locate books of interest.

On LibraryThing, users can catalogue the books they read, manually index them by assigning tags, and write reviews for others to read. Users can also post messages on a discussion forum asking for help in finding new, fun, interesting, or relevant books to read. The forums allow users to tap into the collective bibliographic knowledge of hundreds of thousands of book enthusiasts. On Amazon, users can read and write book reviews and browse to similar books based on links such as "customers who bought this book also bought... ".

Users can search online book collections with different intentions. They can search for specific books of which they know all the relevant details with the intention to obtain them (buy, download, print). In other cases, they search for a specific book of which they do not know those details, with the intention of identifying that book and find certain information about it. Another possibility is that they are not looking for a specific book, but hope to discover one or more books meeting some criteria. These criteria can be related to subject, author, genre, edition, work, series or some other aspect, but also more serendipitously, such as books that merely look interesting or fun to read.

3.2 Task description

Although book metadata can often be used for browsing, this task assumes a user issues a query to a retrieval system, which returns a (ranked) list of book records as results. This query can be a number of keywords, but also one or more book records as positive or negative examples.

We assume the user inspects the results list starting from the top and works her way down until she has either satisfied her information need or gives up. The retrieval system is expected to order results by relevance to the user's information need.

The SBS task is to reply to a user's request that has been posted on the LibraryThing forums (see Section 3.5) by returning a list of recommended books. The books must be selected from a corpus that consists a collection of book metadata extracted from Amazon Books and LibraryThing, extended with associated records from library catalogues of the Library of Congress and the British Library (see the next section). The collection includes both curated and social metadata. User requests vary from asking for books on a particular genre, looking for books on a particular topic or period or books by a given author. The level of detail also varies, from a brief statement to detailed descriptions of what the user is looking for. Some requests include examples of the kinds of

books that are sought by the user, asking for similar books. Other requests list examples of known books that are related to the topic but are specifically of no interest. The challenge is to develop a retrieval method that can cope with such diverse requests. Participants of the SB task are provided with a set of book search requests and are asked to submit the results returned by their systems as ranked lists.

3.3 Submissions

We want to evaluate the book ranking of retrieval systems, specifically the top ranks. We adopt the submission format of TREC, with a separate line for each retrieval result, consisting of six columns:

1. topic_id: the topic number, which is based on the LibraryThing forum thread number.
2. Q0: the query number. Unused, so should always be Q0.
3. isbn: the ISBN of the book, which corresponds to the file name of the book description.
4. rank: the rank at which the document is retrieved.
5. rsv: retrieval status value, in the form of a score. For evaluation, results are ordered by descending score.
6. run_id: a code to identify the participating group and the run.

Participants are allowed to submit up to six runs, of which at least one should use only the *title* field of the topic statements (the topic format is described in Section 3.5). For the other five runs, participants could use any field in the topic statement.

3.4 Data

To study the relative value of social and controlled metadata for book search, we need a large collection of book records that contains controlled subject headings and classification codes as well as social descriptions such as tags and reviews, for a set of books that is representative of what readers are searching for. We use the Amazon/LibraryThing corpus crawled by the University of Duisburg-Essen for the INEX Interactive Track [1].

The collection consists of 2.8 million book records from Amazon, extended with social metadata from LibraryThing. This set represents the books available through Amazon. These records contain title information as well as a Dewey Decimal Classification (DDC) code and category and subject information supplied by Amazon. From a sample of Amazon records we noticed the subject descriptors to be noisy, with many inappropriately assigned descriptors that seem unrelated to the books to which they have been assigned.

The Amazon/LibraryThing collection has a limited amount of professional metadata. Only 61% of the books have a DDC code and the Amazon subjects are noisy with many seemingly unrelated subject headings assigned to books. To

make sure there is enough high-quality metadata from traditional library catalogues, we extended the data set with library catalogue records from the Library of Congress and the British Library. We only use library records of ISBNs that are already in the collection. These records contain formal metadata such as classification codes (mainly DDC and LCC) and rich subject headings based on the Library of Congress Subject Headings (LCSH).[10] Both the LoC records and the BL records are in MARCXML[11] format. We obtained MARCXML records for 1.76 million books in the collection. There are 1,248,816 records from the Library of Congress and 1,158,070 records in MARC format from the British Library. Combined, there are 2,406,886 records covering 1,823,998 of the ISBNs in the Amazon/LibraryThing collection (66%). Although there is no single library catalogue that covers all books available on Amazon, we think these combined library catalogues can improve both the quality and quantity of professional book metadata.

Each book is identified by ISBN. Since different editions of the same work have different ISBNs, there can be multiple records for a single intellectual work. The corpus consists of a collection of 2.8 million records from Amazon Books and LibraryThing.com. See https://inex.mmci.uni-saarland.de/data/nd-agreements.jsp for information on how to get access to this collection. Each book record is an XML file with fields like ¡isbn¿, ¡title¿, ¡author¿, ¡publisher¿, ¡dimensions¿, ¡numberofpage¿ and ¡publicationdate¿. Curated metadata comes in the form of a Dewey Decimal Classification in the ¡dewey¿ field, Amazon subject headings are stored in the ¡subject¿ field, and Amazon category labels can be found in the ¡browseNode¿ fields. The social metadata from Amazon and LibraryThing is stored in the ¡tag¿, ¡rating¿, and ¡review¿ fields. The full list of fields is shown in Table 2.

How many of the book records have curated metadata? There is a DDC code for 61% of the descriptions and 57% of the collection has at least one subject heading. The classification codes and subject headings cover the majority of records in the collection.

More than 1.2 million descriptions (43%) have at least one review and 82% of the collection has at least one LibraryThing tag.

3.5 Information needs

LibraryThing users discuss their books in the discussion forums. Many of the topic threads are started with a request from a member for interesting, fun new books to read. They describe what they are looking for, give examples of what they like and do not like, indicate which books they already know and ask other members for recommendations. Other members often reply with links to works catalogued on LibraryThing, which have direct links to the corresponding records on Amazon. These requests for recommendation are natural expressions

[10] For more information see: http://www.loc.gov/aba/cataloging/subject/

[11] MARCXML is an XML version of the well-known MARC format. See: http://www.loc.gov/standards/marcxml/

Table 2. A list of all element names in the book descriptions

tag name			
book	similarproducts	title	imagecategory
dimensions	tags	edition	name
reviews	isbn	dewey	role
editorialreviews	ean	creator	blurber
images	binding	review	dedication
creators	label	rating	epigraph
blurbers	listprice	authorid	firstwordsitem
dedications	manufacturer	totalvotes	lastwordsitem
epigraphs	numberofpages	helpfulvotes	quotation
firstwords	publisher	date	seriesitem
lastwords	height	summary	award
quotations	width	editorialreview	browseNode
series	length	content	character
awards	weight	source	place
browseNodes	readinglevel	image	subject
characters	releasedate	imageCategories	similarproduct
places	publicationdate	url	tag
subjects	studio	data	

of information needs for a large collection of online book records. We use a selection of these forum topics to evaluate systems participating in the SBS task.

Each topic has a title and is associated with a group on the discussion forums. For instance, topic 99309 in Figure 1 has title *Politics of Multiculturalism Recommendations?* and was posted in the group *Political Philosophy*. The books suggested by members in the thread are collected in a list on the side of the topic thread (see Figure 1). A technique called *touchstone* can be used by members to easily identify books they mention in the topic thread, giving other readers of the thread direct access to a book record on LibraryThing, with associated ISBNs and links to Amazon. We use these suggested books as initial relevance judgements for evaluation. In the rest of this paper, we use the term *suggestion* for books identified in the Touchstone lists in forum topics. Since all suggestions are made by forum members, we assume they are valuable judgements for the relevance of books. We first describe the topic selection procedure and then how we used LibraryThing user profiles to assign relevance values to the suggestions.

Topic selection Topic selection was done the same as last year **(author?)** [4]. We crawled close to 60,000 topic threads and selected threads where at least one book is suggested and the first message contains a book request. First, we identified topics where the topic title reflects the information need expressed in it. For this we used the topic titles as queries and ran them against a full-text index of the A/LT collection. We consider a title to be a decent reflection of

Fig. 1. A topic thread in LibraryThing, with suggested books listed on the right hand side.

the information need if the full-text index found it least one suggestion in the top 1000 results. This left 6510 topics. Next, we used short regular expressions to select messages containing any of a list of phrases like *looking for*, *suggest*, *recommend*. From this set we randomly selected topics and manually selected those topics where the initial message contains an actual request for book recommendations, until we had 89 new topics. We also labeled each selected topic with topic type (requests for books related to *subject*, *author*, *genre*, *edition* etc.) and genre information (*Fiction*, *Non-fiction* or both).

We included the 211 topics from the 2011 Social Search for Best Books task and adjusted them to the simpler topic format of this year. All genre labels were changed to either *Fiction* or *Non-fiction*. The label *Literature* was changed to *Fiction* and all other labels were changed to *Non-fiction*. Specificity labels and examples were removed.

To illustrate how we marked up the topics, we show topic 99309 from Figure 1 as an example:

```
<topic id="99309">
  <title>Politics of Multiculturalism</title>
  <group>Political Philosophy</group>
  <narrative>I'm new, and would appreciate any recommended reading on the
    politics of multiculturalism. Parekh's Rethinking Multiculturalism:
    Cultural Diversity and Political Theory (which I just finished) in the
    end left me unconvinced, though I did find much of value I thought he
    depended way too much on being able to talk out the details later. It
    may be that I found his writing style really irritating so adopted a
    defiant skepticism, but still... Anyway, I've read Sen, Rawls,
```

```
    Habermas, and Nussbaum, still don't feel like I've wrapped my little
    brain around the issue very well and would appreciate any suggestions
    for further anyone might offer.
  </narrative>
  <type>subject</type>
  <genre>non-fiction</genre>
</topic>
```

We think this set represents a broad range of book information needs. We note that the titles and messages of the topic threads may be different from what these users would submit as queries to a book search system such as Amazon, LibraryThing, the Library of Congress or the British Library. Our topic selection method is an attempt to identify topics where the topic title describes the information need. Like last year, we ask the participants to generate queries from the title and initial message of each topic. In the future, we could approach the topic creators on LibraryThing and ask them to supply queries or set up a crowdsourcing task where participants have to search the Amazon/LibraryThing collection for relevant books based on the topic narrative, and we pool the queries they type, and provide the most common query to INEX participants.

User profiles and personal catalogues We can distinguish different relevance signals in these suggestions if we compare them against the books that the topic creator added to her personal catalogue before (pre-catalogued) or after (post-catalogued) starting the topic. We obtained user profiles for each of the topic creators of the topics selected for evaluation and distributed these to the participants. Each profile contains a list of all the books a user has in her personal catalogue, with per book the date on which it was added, and the tags the user assigned to the book. The profiles were crawled at least 4 months after the topic threads were crawled. We assume that within this time frame all topic creators had enough time to decide which suggestions to catalogue.

Catalogued suggestions The list of books suggested for a topic can be split into three subsets. The subset of books that the topic creator had already catalogued before starting the topic (Pre-catalogued suggestions, or Pre-CSs), the subset of books that the topic creator catalogued after starting the topic (Post-catalogued suggestions or Post-CSs) and the subset that the topic creator had not catalogued at the time of crawling the profiles (Non-catalogued suggestions, or Non-CSs).

Members sometimes suggest books that the topic creator already has in her catalogue. In this case, the suggestion is less valuable for the topic creator, but still a sign that for the topic creator that the suggestion makes sense. Similarly, if a topic creator does not catalogue a suggestion, before or after creating the topic, we consider this a signal that the topic creator found the suggestion not valuable enough. In both cases, the suggestion is still a valuable relevance judgement in itself that goes beyond mere topical relevance [3]. In contrast, when the topic creator adds a suggestion to her catalogue after starting the topic (topic creation

is the first signal that she has that particular information need), we assume the suggestion is of great value to the topic creator.

Self-supplied suggestions Some of the books in the Touchstone list are suggestions by the topic creator herself. One reason for these suggestions could be that the creator wants to let others know which books she already knows or has read. Another reason could be that she discovered these books but considered them not good enough for whatever reason. A third reason could be that she discovered these books and wants the opinions of others to help her decide whether it is good enough or not. Because it is hard to identify the reason for a self-supplied suggestion, we consider these suggestions as not relevant, except for the self-supplied suggestions the topic creator later added to her personal catalogue. In this case, the post-cataloguing action is a signal that creator eventually considered it good enough.

Touchstone Suggestions as Judgements This year we used a topic set of 300 topics, including the 211 topics from last year and the 89 new topics. We also provided user profiles of the topic creators as context for generating recommendations. These profiles contain information on which books the user has catalogued and on which the date.

Because we want to focus on suggestions that the topic creator is most interested in, we filtered the 300 topics and retained only those topics where the creator added at least one of the suggested books to her personal catalogue on or after the date she created the forum topic. This resulted in a subset of 96 topics, which is used for evaluation (Section 3.7). The next section describes our method for generating relevance judgements.

3.6 From Suggestions to Relevance Judgements

A system presenting a user with book suggested on the forum fulfils the library objective of helping users to find or locate relevant items, whereas a system presenting the user with books she will add to her catalogue, we argue that it fulfils the library objective of helping her *choose* which of the relevant items to obtain [6]. Based on this correspondence to library cataloguing objective, we assign higher relevance values to books that are post-catalogued than to other suggestions.

We use the following terminology:

Creator The topic creator, who has the information need and formulated the request.

Suggestion Suggestions are books mentioned in the messages of the topic thread, and that are identified via Touchstone.

Suggestor The forum member who first suggests a book. The thread is parsed from first to last message, and the first member to mention the book is considered the suggestor. Note that this can be the topic creator. Perhaps

she suggests books because she wants others to comment on them or because she wants to show she already knows about these books.

Pre-catalogued Suggestion a suggestion that the creator catalogues before starting the topic.

Post-catalogued Suggestion a suggestion that the creator catalogues after having started the topic.

Non-catalogued Suggestion a suggestion that the creator did not catalogue before or after having started the topic.

To operationalise the suggestions as relevance judgements, we use different relevance values (rv):

Highly relevant (rv=4) Post-catalogued Suggestions are considered the best suggestions, regardless of who the suggestor is.

Relevant (rv=1) Pre- and Non-catalogued suggestions where the suggestor is not the creator. Suggestions from others that the creator already has are good suggestions in general (perhaps not useful for the creator, but still relevant to the request).

Non-relevant (rv=0) Pre- and Non-catalogued suggestions that the creator suggested herself, i.e., the suggestor is the creator. These are either books the creator already has (pre-catalogued) or may be negative examples (*I'm not looking for books like this*), or are mentioned for some other reason. The creator already knows about these books.

We use the recommended books for a topic as relevance judgements for evaluation. Each book in the Touchstone list is considered relevant. How many books are recommended to LT members requesting recommendations in the discussion groups? Are other members compiling exhaustive lists of possibly interesting books or do they only suggest a small number of the best available books? Statistics on the number of books recommended for the Full set of 300 topics and the PCS subset of 96 topics with post-catalogued suggestions are given in Table 3.

We first compare the suggestions for the full topic set with those of 2011. The two sets of suggestions are similar in terms of minimum, median and mean number of suggestions per topic. The maximum has increased somewhat this year. Split over genres we see that the Fiction topics tend to get more suggestions than Non-Fiction topics. Topics where creators explicitly mention both fiction and non-fiction recommendation are welcome—denoted Mix—are more similar to Non-Fiction topics in terms of maximum and median number of suggestions, but closer to Fiction topics in terms of mean number of suggestions.

If we zoom in on the PCS topics, we see they have a larger number of suggestions per topic than the Full set, with a mean (median) of 16.2 (9). Most of the suggestions are not catalogued by the topic creator and made by others (RV_1). In most topics there is at least one book first mentioned by the topic creator (RV_0 has a median of 1), and only a small number suggestions are post-catalogued by the creator (RV_4 has a mean of 1.7 and median of 1). What does it mean that the PCS topics get more suggestions than the other topics in the Full set? One

Table 3. Statistics on the number of recommended books for the 101 topics from the LT discussion groups

RV	# topics	# suggest.	min.	max.	mdn.	mean	std. dev.
Full							
2011	211	2377	1	79	7	11.3	12.5
2012	300	3533	1	101	7	11.8	14.5
Fiction	135	2143	1	101	9	15.9	18.0
Non-fiction	146	1098	1	56	5	7.5	7.8
Mix	19	292	1	59	7	15.4	16.2
PCS							
RV_{0+1+4}	96	1558	1	98	9	16.2	17.7
RV_0	96	194	0	21	1	2.0	3.8
RV_1	96	1200	0	80	7	12.5	16.1
RV_4	96	164	1	14	1	1.7	1.6

Table 4. Statistics on the number of topics per genre for the full set of 300 topics and the 96 topics with PCSs

	Topic set	
Genre	Full	PCS
All	300	96
Fiction	135 (45%)	49 (51%)
Non-fiction	146 (49%)	36 (38%)
Mix	19 (6%)	9 (9%)
Subject	207 (69%)	60 (63%)
Author	36 (12%)	16 (17%)
Genre	64 (21%)	21 (22%)
Known-item	15 (5%)	5 (5%)

reason might be that with more suggestions, there is a larger a priori probability that the topic creator will catalogue at least one of them. Another, related, reason is that a larger number of suggestions means the list of relevant books is more complete, which could make the topic creator more confident that she can make an informed choice. Yet another reason may be that PCS topics are dominated by Fiction topics, which have more suggestions than the Non-Fiction topics.

In Table 4 we show the number of Fiction, Non-Fiction and Mix topics in the Full and PCS topics sets. In the Full set, there are a few more Non-Fiction topics (146, or 49%) than Fiction topics (135 or 45%), with only 19 (6%) Mix topics. In the PCS set, this is the other way around, with 49 Fiction topics (51%), 36 Non-fiction topics (38%) and 9 Mix topics (9%). This partly explains why the PCS topics have more suggestions. Post-cataloguing tends to happen more often in topic threads related to fiction.

Judgements from Mechanical Turk To get a better understanding of the nature of book suggestions and book selection, we plan to gather rich relevance judgements from Mechanical Turk that cover different aspects of relevance. Workers will judge the relevance of books based on the book descriptions in the collection and the topic statement from the LT forum. Instead of asking them to judge the overall relevance of books, we plan to ask them to identify different relevance aspects of the information need and to judge the books on each of these aspects separately. Additionally, we ask them to identify which part of the description (title, subject headings, reviews or tags) is useful to determine the relevance of the book for each relevance aspect in the request. Of course, workers are not able to judge books on the user-dependent (personal, affective relevance aspects) of the topic creator. For these aspect we would need judgements from the topic creator herself. One possibility is to approach topic creator on the forums or via private messages to they LT profile.

We are currently in the process of setting up the Mechanical Turk experiment and hope to have results for the final report in the official proceedings.

ISBNs and intellectual works Each record in the collection corresponds to an ISBN, and each ISBN corresponds to a particular intellectual work. An intellectual work can have different editions, each with their own ISBN. The ISBN-to-work relation is a many-to-one relation. In many cases, we assume the user is not interested in all the different editions, but in different intellectual works. For evaluation we collapse multiple ISBN to a single work. The highest ranked ISBN is evaluated and all lower ranked ISBNs of the same work ignored. Although some of the topics on LibraryThing are requests to recommend a particular edition of a work—in which case the distinction between different ISBNs for the same work are important—we leave ignore these distinctions to make evaluation easier. This turns edition-related topics into known-item topics.

However, one problem remains. Mapping ISBNs of different editions to a single work is not trivial. Different editions may have different titles and even have different authors (some editions have a foreword by another author, or a translator, while others have not), so detecting which ISBNs actually represent the same work is a challenge. We solve this problem by using mappings made by the collective work of LibraryThing members. LT members can indicate that two books with different ISBNs are actually different manifestations of the same intellectual work. Each intellectual work on LibraryThing has a unique work ID, and the mappings from ISBNs to work IDs is made available by LibraryThing.[12]

The mappings are not complete and might contain errors. Furthermore, the mappings form a many-to-many relationship, as two people with the same edition of a book might independently create a new book page, each with a unique work ID. It takes time for members to discover such cases and merge the two work IDs, which means that at any time, some ISBNs map to multiple work IDs even though they represent the same intellectual work. LibraryThing can detect such cases but, to avoid making mistakes, leaves it to members to merge them. The

[12] See: http://www.librarything.com/feeds/thingISBN.xml.gz

Table 5. Evaluation results for the official submissions. Best scores are in bold

Run	MRR	nDCG@10	P@10	R@10	R@1000
p54.run2.all-topic-fields.all-doc-fields	0.3069	**0.1492**	0.1198	0.1527	0.5736
p54.run3.all-topic-fields.QIT.alpha0.99	0.3066	0.1488	0.1198	0.1527	0.5736
p4.inex2012SBS.xml_social.fb.10.50	**0.3616**	0.1437	0.1219	0.1494	**0.5775**
p62.B_IT30_30	0.3410	0.1339	**0.1260**	**0.1659**	0.5130
p4.inex2012SBS.xml_social	0.3256	0.1297	0.1135	0.1476	0.5588
p62.mrf-booklike	0.3584	0.1295	0.1250	0.1514	0.5242
p54.run5.title.II.alpha0.94	0.2558	0.1173	0.1073	0.1289	0.4891
p62.IOT30	0.2933	0.1141	0.1240	0.1503	0.5864
p62.IT30	0.2999	0.1082	0.1187	0.1426	0.5864
p54.run6.title.II.alpha0.97	0.2392	0.0958	0.0823	0.0941	0.4891
p62.lcm-2	0.2149	0.0901	0.0667	0.1026	0.5054
p100.sb_g0	0.2394	0.0884	0.0844	0.1145	0.5524
p54.run4.title.QIT.alpha0.65	0.1762	0.0875	0.0719	0.0949	0.4891
p100.sb_g_ttl_nar0	0.1581	0.0740	0.0594	0.0939	0.4634
p54.run1.title.all-doc-fields	0.1341	0.0678	0.0583	0.0729	0.4891
p100.sb_2xsh_ttl_nar0	0.0157	0.0057	0.0021	0.0022	0.0393
p100.sb_2xsh0	0.0199	0.0042	0.0021	0.0020	0.0647

fraction of works with multiple ISBNs is small so we expect this problem to have a negligible impact on evaluation.

3.7 Evaluation

This year four teams together submitted 17 runs. The Oslo and Akershus University College of Applied Sciences (OAUCAS) submitted 4 runs, the Royal School of Library and Information Science (RSLIS) submitted 6 runs, the University of Amsterdam (UAms) submitted 2 runs and the LIA group of the University of Avignon (LIA) submitted 5 runs.

The official evaluation measure for this task is nDCG@10. It takes graded relevance values into account and concentrates on the top retrieved results. The set of PCS topics and corresponding suggestions form the official topics and relevance judgements for this year's evaluation. The results are shown in Table 5.

The best performing run is *p54.run2.all-topic-fields.all-doc-fields* by **RSLIS**, which used all topic fields combined against an index containing all available document fields.

The best run by **UAms** is *p4.inex2012SBS.xml_social.fb.10.50*, which uses only the topic titles and ran against an index containing the title information fields (title, author, edition, publisher, year) and the user-generated content fields (tags, reviews and awards). Blind relevance feedback was applied using the top 50 terms from the top 10 initial retrieval results.

The best run by **LIA** is *p62.B_IT30_30*.

The best run by **OAUCAS** is *p100.sb_g0*.

We note that the best run does not use any information from the user profiles. The best performing run that incorporates user profile information is the second best run, *p54.run3.all-topic-fields.QIT.alpha0.99* by RSLIS. Like the best performing run, it uses all topic fields against all document fields, but re-ranks the results list based on the LT profile of the topic creator. Retrieved books that share a lot of tags associated books already present in the user's catalog are regarded as a more appropriate match. The final retrieval score is a linear combination of the original content-based score and the cosine similarity between a tag vector containing the tag counts from a user's personal catalog and the tag vectors of the retrieved books.

The run *p4.inex2012SBS.xml_social.fb.10.50* achieves the highest MRR score (0.3616), which means that on average, it retrieves the first relevant book at or above rank 3. The nine best systems achieve a P@10 score just above 0.1, which means on average they have one suggestion in the top 10 results. Most systems are able to retrieve an average of around 50% of the suggestions in the top 1000 results.

Note that the three highest scores for P@10 (0.1260, 0.1250 and 0.1240) correspond with the 4th, 6th and 8th highest scores for nDCG@10. The highest nDCG@10 score corresponds to the 5th highest P@10 score. This could mean that top performing system is not better than the other systems at retrieving suggestions in general, but that it is better at retrieving PCSs, which are the most important suggestions. The top two runs have similar nDCG@10 scores and the same P@10 scores and retrieve more PCSs in the top 10 (36 over all 96 topics) than the other runs, the best of which retrieves only 26 PCSs in the top 10, over all 96 topics. The full topic statement is a more effective description of the books that the topic creator will catalogue than the topic title alone.

In sum, systems that incorporate user profile information have so far not been able to improve upon a plain text retrieval baseline. The best systems for retrieving PCSs use the full topic statement.

Recall that the evaluation is done on a subset of the Full set of 300 topics. In Section 3.5 we found that the PCS topics have more suggestions per topic than the rest of the topics in the Full set, and that the fraction of Fiction topics is also higher in the PCS set. To what extent does this difference in genre and number of suggestions result in differences in evaluation?

We compare the system rankings of the official relevance judgements (PCS topics with differentiated relevance value, denoted PCS(RV_{0+1+4}) with two alternative sets. One based on the same topics but with all suggestions mapped to relevance value $rv = 1$ (denoted PCS(RV_{flat}) and the other is the set of judgements for the Full set of 300 topics, where all suggestions were also all mapped to relevance value $rv = 1$, denoted Full(RV_{flat}. The PCS(RV_{flat} set allows us to see whether the differentiation between suggestions affects the ranking. The comparison between PCS(RV_{flat} and Full(RV_{flat} can show whether the different topic selection criteria lead to different system rankings.

Table 6 shows the Kendall's Tau (column 2) and Tau$_{AP}$ (column 3) ranking correlations over the 18 official submissions for nDCG@10. The Tau$_{AP}$ ranking

Table 6. Kendall's Tau and Tau$_{AP}$ system ranking correlations between the relevance judgements of Full and PCS topics. PCS(RV_{flat} represents judgements where all suggestions have $RV = 1$, PCS(RV_{0+1+4} represents the suggestion with differentiated relevance values.

Qrels	τ	τ_{AP}
Full(RV_{flat} / PCS(RV_{flat}	0.91	0.79
Full(RV_{flat} / PCS(RV_{0+1+4}	0.85	0.73
PCS(RV_{flat} / PCS(RV_{0+1+4}	0.91	0.93

correlation puts more weight on the top-ranked systems [7], emphasising how well the evaluations agree on ranking the best systems. The standard Kendall Tau correlation is very strong (> 0.9) between Full(RV_{flat} and PCS(RV_{flat}, suggesting the topic selection plays little role. The correlation between PCS(RV_{flat} and PCS(RV_{0+1+4} is also very high, furthermore suggesting that the differentiation between suggestions has no impact on the ranking. However, the Tau$_{AP}$ correlations show that disagreement between Full(RV_{flat} and PCS(RV_{flat} is bigger among the top ranked system than the on the lower scoring systems. The two PCS sets have very strongly correlated system rankings. From this we conclude that the differentiation between suggestions in terms of relevance value has little impact, but that the PCS topics are somewhat different in nature than the other topics.

4 The Prove It (PI) Task

The goal of this task was to investigate the application of focused retrieval approaches to a collection of digitised books. The scenario underlying this task is that of a user searching for specific information in a library of books that can provide evidence to confirm or reject a given factual statement. Users are assumed to view the ranked list of book parts, moving from the top of the list down, examining each result. No browsing is considered (only the returned book parts are viewed by users).

Participants could submit up to 10 runs. Each run could contain, for each of the 83 topics (see Section 4.2), a maximum of 1,000 book pages estimated relevant to the given aspect, ordered by decreasing value of relevance.

A total of 18 runs were submitted by 2 groups (6 runs by UMass Amhers (ID=50) and 12 runs by Oslo University College (ID=100)), see Table 1.

4.1 The Digitized Book Corpus

The track builds on a collection of 50,239 out-of-copyright books[13], digitised by Microsoft. The corpus is made up of books of different genre, including history books, biographies, literary studies, religious texts and teachings, reference

[13] Also available from the Internet Archive (although in a different XML format)

works, encyclopaedias, essays, proceedings, novels, and poetry. 50,099 of the books also come with an associated MAchine-Readable Cataloging (MARC) record, which contains publication (author, title, etc.) and classification information. Each book in the corpus is identified by a 16 character long bookID – the name of the directory that contains the book's OCR file, e.g., A1CD363253B0F403.

The OCR text of the books has been converted from the original DjVu format to an XML format referred to as BookML, developed by Microsoft Development Center Serbia. BookML provides additional structure information, including markup for table of contents entries. The basic XML structure of a typical book in BookML is a sequence of pages containing nested structures of regions, sections, lines, and words, most of them with associated coordinate information, defining the position of a bounding rectangle ([coords]):

```
<document>
 <page pageNumber="1" label="PT_CHAPTER" [coords] key="0" id="0">
  <region regionType="Text" [coords] key="0" id="0">
   <section label="SEC_BODY" key="408" id="0">
    <line [coords] key="0" id="0">
     <word [coords] key="0" id="0" val="Moby"/>
     <word [coords] key="1" id="1" val="Dick"/>
    </line>
    <line [...]><word [...] val="Melville"/>[...]</line>[...]
   </section>   [...]
  </region>     [...]
 </page>        [...]
</document>
```

BookML provides a set of labels (as attributes) indicating structure information in the full text of a book and additional marker elements for more complex structures, such as a table of contents. For example, the first label attribute in the XML extract above signals the start of a new chapter on page 1 (label="PT_CHAPTER"). Other semantic units include headers (SEC_HEADER), footers (SEC_FOOTER), back-of-book index (SEC_INDEX), table of contents (SEC_TOC). Marker elements provide detailed markup, e.g., for table of contents, indicating entry titles (TOC_TITLE), and page numbers (TOC_CH_PN), etc.

The full corpus, totaling around 400GB, was made available on USB HDDs. In addition, a reduced version (50GB, or 13GB compressed) was made available for download. The reduced version was generated by removing the word tags and propagating the values of the `val` attributes as text content into the parent (i.e., line) elements.

4.2 Topics

In recent years we have had a topic-base of 83 topics, 21 of which we have collected relevance judgments for using crowdsourcing through the Amazon Mechanical Turk infrastructure [2].

The ambition this year has been two-fold:

– To increase the number of topics
– To further develop the relevance judgment method, so as to combat the effect of the statement complexity on the assessment consistency.

For the second point above, we have been attempting to divide each topics into its primitive aspects (a process we refer to as "aspectization"). To this end we developed a simple web-application with a database back-end, to allow anyone to aspectize topics. This resulted in 30 topics

For each page being assessed for confirmation / refutation of a topic, the assessor is presented with a user interface similar to Figure 2

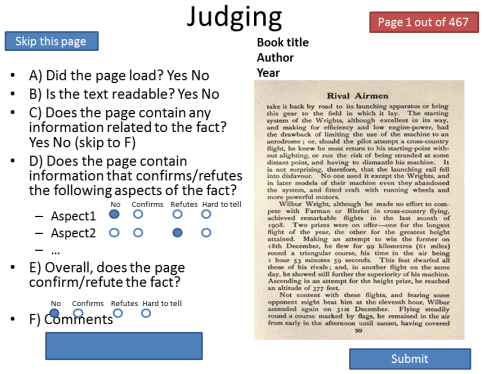

Fig. 2. An illustration of the planned assessment interface

This means that we go from a discrete (confirms / refute / none) assessment to a graded assessment, where a page may e.g. be assessed by a certain as 33 percent confirming a topic, if one of three aspects is judged as confirmed by him/her for that page.

For the current assessment we have prepared 30 topics, for which the number of aspects range from 1 (very simple statements) to 6 per topic with an average of 2,83 aspects per topic.

4.3 Collected Relevance Assessments

At the time of writing this years relevance assessment are still not collected yet.

4.4 Evaluation Measures and Results

Result publication is awaiting the conclusion of the relevance assessment process.

5 Conclusions and plans

This paper presents an overview of the INEX 2012 Social Book Search Track. This year, the track ran two tasks: the Social Book Search task, and the Prove It task.

The Social Book Search (SBS) task changed focus from the relative value of professional and user-generated metadata, to the complexity of book search information needs.

We extended our investigation into the nature of book requests and suggestions from the LibraryThing forums as statements of information needs and relevance judgements. By differentiating between who the suggestor is and whether the topic creator subsequently adds a suggestion to her catalogue or not (post catalogued suggestions), we want to focus even more on the personal, affective aspects of relevance judgement in social book search. We operationalised this by differentiating in relevance values, giving higher values for post-catalogued suggestions than for other suggestions.

Our choice to focus on topics with post-catalogued suggestions (PCS topics) resulted in a topic set that is slightly different from the topics we used last year, where we ignored the personal catalogued of the topic creator and considered all topics that have a book request, a descriptive title and at least one suggestion. The PCS topics have more suggestions on average than other topics, and a larger fraction of them is focused on fiction books. This results in a difference in system ranking, which is mainly due to the different nature of the topics, and not in the differentiation of the relevance values.

In addition to the topic statements extracted from the forum discussions, we extracted user profiles of the topic creators, which contain full catalogue information on which books they have in the personal catalogues, when each book was added to the catalogue and which tags the user assigned to each book. These profiles were distributed along with the topic statements, to allow participants to build systems that incorporate both the topical description of the information need and personal behaviour, preferences and interests of the topic creators.

The evaluation has shown that the most effective systems incorporate the full topic statement, which includes the title of the topic thread, the name of the discussion group, and the full first message that elaborates on the request. However, the best system did not use any user profile information. So far, the best system is a plain full-text retrieval system.

Next year, we continue with the task to further investigate the role of user information. We also plan to enrich the relevance judgements with further judgements on the relevance of books to specific relevance aspects of the information need. For this, we plan to use either Mechanical Turk or approach the topic creators on LibraryThing to obtain more specific judgements directly from the person with the actual information need.

This year the Prove It task has undergone some changes when it comes to assessments. The number of participants for the PI task is still low, which also puts some limitations on what we are able to do collaboratively, but based

on the changes introduced this year which will hopefully give us more useful assessments, we hope to increase the number of participants, further vitalizing the task.

Acknowledgments We are very grateful to Justin van Wees for providing us with the user profiles of the topic creators for this year's evaluation. This research was supported by the Netherlands Organization for Scientific Research (NWO projects # 612.066.513, 639.072.601, and 640.005.001) and by the European Communitys Seventh Framework Program (FP7 2007/2013, Grant Agreement 270404).

Bibliography

[1] Thomas Beckers, Norbert Fuhr, Nils Pharo, Ragnar Nordlie, and Khairun Nisa Fachry. Overview and results of the inex 2009 interactive track. In Mounia Lalmas, Joemon M. Jose, Andreas Rauber, Fabrizio Sebastiani, and Ingo Frommholz, editors, *ECDL*, volume 6273 of *Lecture Notes in Computer Science*, pages 409–412. Springer, 2010.

[2] Gabriella Kazai, Jaap Kamps, Marijn Koolen, and Natasa Milic-Frayling. Crowdsourcing for book search evaluation: Impact of hit design on comparative system ranking. In *Proceedings of the 34th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 205–214. ACM Press, New York NY, 2011.

[3] Marijn Koolen, Jaap Kamps, and Gabriella Kazai. Social Book Search: The Impact of Professional and User-Generated Content on Book Suggestions. In *Proceedings of the International Conference on Information and Knowledge Management (CIKM 2012)*. ACM, 2012.

[4] Marijn Koolen, Gabriella Kazai, Jaap Kamps, Antoine Doucet, and Monica Landoni. Overview of the INEX 2011 books and social search track. In Shlomo Geva, Jaap Kamps, and Ralf Schenkel, editors, *Focused Retrieval of Content and Structure: 10th International Workshop of the Initiative for the Evaluation of XML Retrieval (INEX 2011)*, volume 7424 of *LNCS*. Springer, 2012.

[5] Kara Reuter. Assessing aesthetic relevance: Children's book selection in a digital library. *JASIST*, 58(12):1745–1763, 2007.

[6] Elaine Svenonius. *The Intellectual Foundation of Information Organization*. MIT Press, 2000.

[7] Emine Yilmaz, Javed A. Aslam, and Stephen Robertson. A new rank correlation coefficient for information retrieval. In Sung-Hyon Myaeng, Douglas W. Oard, Fabrizio Sebastiani, Tat-Seng Chua, and Mun-Kew Leong, editors, *SIGIR*, pages 587–594. ACM, 2008.