INEX 2010 Book Track
Tasks and Submission Guidelines

Gabriella Kazai, Marijn Koolen, Monica Landoni, Antoine Doucet   Version 1.0

1   Introduction

The goal of the Book track is to evaluate techniques that provide various services on collections of digitized books, e.g., browsing, searching, and study user interface issues and the behaviour of the users of such services. Towards this goal, the track investigates the following tasks: the Best Books to Reference search task (Section 3), the Prove It search task (Section 4), the Structure Extraction task (Section 5) and the Active Reading task (Section 6).

Participants may take part in any or all the tasks. The minimum participation requirement is to contribute results or study participants to at least one of the tasks AND to contribute relevance judgements.

2   What's new in 2010 compared to 2009?

- The topic format has changed significantly: Topics now contain factual statements.
- We have two new search tasks: "Prove It" and "Best Book to Reference" (or Best Books for short). In the Prove It task, systems need to find evidence in books that can be used to either confirm or refute a factual statement. In the Best Books task, systems need to return the most relevant books on the general subject area of the factual statement.
- Only page level results are accepted in the Prove It task (no passages or other XML elements).
- Max 100 books can be returned for the Best Books task, and maximum 1000 pages for the Prove It task.
- In the Prove It task, systems need to return a ranked list of pages – not grouped by books.
- The submission DTDs are new. Please take care to conform to these when submitting your runs.
- You can submit up to 12 runs per task.

3   Best Books to Reference task

3.1   Goals

The goal of this task is to compare book-specific IR techniques with standard IR methods for the retrieval of books, where the results are whole books.

3.2   User scenario

The scenario underlying this task is that of a user searching for books on a given topic with the intent to build a reading or reference list. The search results are whole books, presented in a ranked list. Users view the ranked list moving from the top of the list down, examining each book result. Relevant books are those that the user would add to their reading list for the given topic. Books that are dedicated to the topic of the request (i.e., are focused on the topic) are considered more relevant. The reading list that the user is building is an ordered list of books, where the most relevant book is at the top of the list.

3.3 Task description

The task is to return, for each test topic, a ranked list of 100 (one hundred) books estimated relevant to the general subject area of the factual statement expressed within the test topics, ranked in order of estimated relevance.

Both standard IR and focused or structured document retrieval (SDR) approaches may be employed to generate the results. Approaches may be either general (i.e., not specific to the book domain) or may be book-specific that make use of domain-specific information or technology (e.g., library catalogue information, back of book indexes, etc., or specialised ranking strategies or tuning methods).

The task builds on the corpus of over 50,000 digitized books. Search engine performance will be evaluated using relevance judgements collected at page level, which will be aggregated to give book level relevance scores: the more relevant pages a book has relative to its length the more relevant it is considered to be. Book-level precision/recall, MRR and NDCG metrics will be reported.

Participants may submit up to 12 runs in total, either single runs or pairs of runs (i.e., 12 single runs, 6 pairs of runs, or any other combination of individual or paired runs). A single run may either be the result of generic (non-specific) or book-specific IR methods. When pairs of runs are submitted, one run should be the result of applying non-specific IR techniques; and the other run should be generated using the same techniques (where possible), but with the use of additional book-specific features (e.g., back-of-book index, citation statistics, book reviews, etc.) or specifically tuned methods.

3.4 Submission format

The DTD describing the submission format for the Best Books task is as follows:

```
<!ELEMENT bs-submission (topic-fields, description, topic+)>
<!ATTLIST bs-submission
participant-id      CDATA       #REQUIRED
run-id              CDATA       #REQUIRED
paired-run-id       CDATA       #REQUIRED
task                (book-retrieval) #REQUIRED
query               (automatic | manual) #REQUIRED
result-type         (book)       #REQUIRED
retrieval-type      (non-specific | book-specific) #REQUIRED
>
<!ELEMENT topic-fields EMPTY>
<!ATTLIST topic-fields
fact                (yes|no) #REQUIRED
subject             (yes|no) #REQUIRED
query               (yes|no) #REQUIRED
narrative           (yes|no) #REQUIRED
>
<!ELEMENT description (#PCDATA)>
<!ELEMENT topic (book+)>
<!ATTLIST topic topic-id CDATA #REQUIRED >
<!ELEMENT book (bookid, rank?, rsv?)>
<!ELEMENT bookid  (#PCDATA)>
<!ELEMENT rank    (#PCDATA)>
<!ELEMENT rsv     (#PCDATA)>
```

Each submission run must contain the following:

- @participant-id: The Participant ID number of the submitting institute (available from the INEX website at: http://www.inex.otago.ac.nz/people/participants.asp).
- @run-id: A run ID (which must be unique across all submissions sent from one organization – also please use meaningful, but short names if possible).
- @paired-run-id: The run-id identifying the run that the current submission is paired with (i.e., if the current run is the book-specific ranking then the paired run-id should be the id of the

generic ranking – these two runs can then be compared to each other). If a single run is submitted, please use "NA".

- @task: Identification of the task – please set this to "book-retrieval".
- @query: Specification whether the search query used to generate the run was constructed automatically ("automatic") or manually ("manual") from the topic.
- @result-type: Specification of the result-type, which should be set to "book".
- @retrieval-type: Specifies whether the run is a result of generic ("non-specific") IR methods, or "book-specific" IR techniques that make use of book-specific features or algorithms.
- topic-fields: Specification of which topic fields were used for constructing the search query (i.e., fact and/or subject and/or narrative, etc.).
- description: A description of the retrieval approach used to generate the run. Please add as much detail as you can, as this would help with the comparison and analysis of the results later on, as well as in the summary paper.

Furthermore, a run should contain the search results for each topic, confirming to the following criteria:

- topic: Contains the ranked list of books estimated relevant to the given topic, ordered by decreasing value of relevance. Only **a maximum of 100** books should be returned for each topic.
- @topic topic-id: Identifies the topic.
- book: Contains information for each book result in the ranking as follows.
- bookid: Each book should be identified using its bookID, which is the name of the directory that contains the XML source of the book.
- rank/rsv: The rank position and/or the relevance status value (RSV) can be recorded for each book in the ranking. Please note, however, that the evaluation will rely on the actual ordering of results alone (values of the rank and rsv fields will be ignored).

An example submission is:

```
<bs-submission participant-id="25" run-id="BM25F-With-ToC-BackOfBookIndex-Streams"
               paired-run-id="BM25" task="book-retrieval" query="automatic"
               result-type="book" retrieval-type="book-specific">
 <topic-fields fact="yes" subject="no" query="no" question="no" narrative="no"/>
 <description>BM25F using 2 streams extracted from the table of contents and the back-of-book
              index sections of books. The rest of the book content is ignored. Parameters of
              BM25F were trained using RankNet.</description>
<topic topic-id="01">
  <book>
    <bookid>300A5334B2869F47</bookid>
    <rank>1</rank>
  </book>
  <book>
    <bookid>BAD598FB0A7D02E2</bookid>
    <rank>2</rank>
  </book>
  <book>…</book>
 …
</topic>
<topic> … </topic>
</bs-submission>
```

4 Prove It task

4.1 Goals

The goal of this task is to investigate the application of focused retrieval approaches to a collection of digitized books.

4.2 User scenario

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).

4.3 Task description

The task is to return a ranked list of 1000 (one thousand) book pages (given by their XPaths)**, containing relevant information that can be used to either confirm or reject the factual statement expressed in the topic, ranked in order of estimated relevance.**

The track builds the same corpus as the Best Books task. Relevance judgements will be collected at the page level, asking judges to mark a page relevant if it either confirms or refutes the factual statement of the topic. Page-level precision/recall and NDCG metrics will be reported (subject to change).

Participants may submit up to 12 runs.

4.4 Submission format

Submissions for the Prove It task should conform to the following DTD:

```
<!ELEMENT bs-submission (topic-fields, description, topic+)>
<!ATTLIST bs-submission
participant-id      CDATA      #REQUIRED
run-id              CDATA      #REQUIRED
task                (focused) #REQUIRED
query               (automatic | manual) #REQUIRED
result-type         (page)      #REQUIRED
>
<!ELEMENT topic-fields EMPTY>
<!ATTLIST topic-fields
fact                (yes|no) #REQUIRED
subject             (yes|no) #REQUIRED
query               (yes|no) #REQUIRED
narrative           (yes|no) #REQUIRED
>
<!ELEMENT description (#PCDATA)>
<!ELEMENT topic (book+)>
<!ATTLIST topic topic-id CDATA #REQUIRED >
<!ELEMENT result (bookid, path, rank?, rsv?)>
<!ELEMENT bookid  (#PCDATA)>
<!ELEMENT path    (#PCDATA)>
<!ELEMENT rank    (#PCDATA)>
<!ELEMENT rsv     (#PCDATA)>
```

Each submission must contain the following:

- @participant-id: The Participant ID number of the submitting institute (available from the INEX website at: http://www.inex.otago.ac.nz/people/participants.asp).
- @run-id: A run ID (which must be unique across all submissions sent from one organization – also please use meaningful, but short names if possible).
- @task: Identification of the task – please set this to "focused".
- @query: Specification whether the search query was constructed automatically ("automatic") or manually ("manual") from the topic.
- @result-type: Specification of the result-type – please set this to "page". A page is an XML element that should be given by its XPath (see Appendix A).
- topic-fields: Specification of which topic fields were used for constructing the search query (i.e., fact and/or query and/or narrative, etc.).

- description: A description of the retrieval approach applied to generate the run. Please add as much detail as you can, as this would help with the comparison and analysis of the results later on.

Furthermore, a run should contain the search results for each topic confirming to the following criteria:

- topic: Contains the ranked list of books estimated relevant to the given topic, ordered by decreasing value of relevance. Only a maximum of 1000 books should be returned for each topic.
- @topic topic-id: Identifies the topic.
- result: Contains information for each book-part result in the ranking..
- bookid: Each book should be identified using its bookID, which is the name of the directory that contains the XML source of the book.
- path: Book page results, identified by their XPaths, please see Appendix A.
- rank/rsv: For each result inside a book, its rank and/or RSV score can be recorded. Please note that the evaluation may rely on the rank order of the books and of the results inside books alone (values of the rank and rsv fields may be ignored).

An example submission may be as follows:

```
<bs-submission participant-id="25" run-id="BM25F-Focused-PageLevelRetrieval-With-ToC-
            BackOfBookIndex-Streams" task="book-focused" query="automatic"
            result-type="page">
 <topic-fields title="yes" description="no" narrative="no"/>
 <description>BM25F using 2 streams extracted from the table of contents and the back-of-book
            index sections, indexing and retrieval only at page level, no relevance
            propagation</description>
<topic topic-id="01">
 <result>
   <bookid>384D10DAEA4E34A8</bookid>
   <path>/document[1]/page[27]</path>
   <rank>1</rank>
 </result>
 <result>
   <bookid>384D10DAEA4E34A8</bookid>
   <path>/ document[1]/page [122]</path>
   <rank>2</rank>
 </result>
 <result>
   <bookid>5AFEE130174076E3</bookid>
   <path>/ document[1]/page [5]</path>
   <rank>3</rank>
 </result>
   …
</topic>
<topic> … </topic>
</bs-submission>
```

5   Structure Extraction task

5.1   Goals

The goal of this task is to test and compare automatic techniques for deriving structural information from digitized books in order to build a hyperlinked table of contents. This is motivated by current digitization and OCR technologies which produce the full text of digitized books with only minimal structure information. Pages and paragraphs are usually identified and marked up in the OCR, but more sophisticated structures, such as chapters, sections, etc., are currently not recognised.

5.2   Task description

The task is to build the table of contents (ToC) for digitized books, using information from the OCR (in DjVu XML format), PDF or JPEG image files.

The track uses a sample collection of 1000 digitized books of different genre and style. For each book, the OCR output (DjVu file), PDF file, and the JPEG images of each scanned page are made available.

The ToCs created by participants will be compared to a manually built ground truth (from the PDF of a book), and will be evaluated using recall/precision like measures at different structural levels (i.e., different depths in the table of contents). In addition, the quality of the created ToCs will also be evaluated independently: Participants will be asked to grade them on a multi-level quality scale.

Participants may submit up to 10 runs, each containing the ToC for all 1000 books in the test set.

5.3 Application

The generated ToCs may be used by an e-book reader system and presented to users as a hyperlinked hierarchy. Users expect to see the section titles as entries and should be able to click on an entry and jump to the start of the selected section in the book.

5.4 Use of ToCs in other tasks

Participants are welcome to apply their ToC extraction techniques to the main corpus of 50,000 books and submit runs to one of the search tasks exploiting this additional information. Note, however, that for the main corpus only the OCR text, in OCRML format (different from the DjVu), is available as input (no PDF or JPEGs). The generated ToCs may also be used and evaluated through user studies in the Active Reading task.

5.5 Submission format

Submissions for the Structure Extraction task should conform to the following DTD:

```
<!ELEMENT bs-submission (source-files, description, book+)>
<!ATTLIST bs-submission
participant-id      CDATA       #REQUIRED
run-id              CDATA       #REQUIRED
task                (book-toc) #REQUIRED
toc-creation        (automatic | semi-automatic) #REQUIRED
toc-source          (book-toc | no-book-toc | full-content | other) #REQUIRED
>
<!ELEMENT source-files EMPTY>
<!ATTLIST source-files
xml                 (yes|no) #REQUIRED
pdf                 (yes|no) #REQUIRED
jpg                 (yes|no) #REQUIRED
>
<!ELEMENT description (#PCDATA)>
<!ELEMENT book (bookid, toc-entry+)>
<!ELEMENT bookid  (#PCDATA)>
<!ELEMENT toc-entry(toc-entry*)>
<!ATTLIST toc-entry
        title        (#PCDATA) #REQUIRED
        page         (#PCDATA) #REQUIRED
>
```

Each submission must contain the following:

- @participant-id: The Participant ID number of the submitting institute (available from the INEX website at: http://www.inex.otago.ac.nz/people/participants.asp).
- @run-id: A run ID (which must be unique across all submissions sent from one organization – also please use meaningful, but short names if possible).
- @task: Identification of the task, which should just be "book-toc".
- @toc-creation: Specification whether the ToC was constructed fully automatically ("automatic") or with some manual aid ("semi-automatic").

- **@toc-source:** Specification of whether the ToC was built based only on the table of contents part of the book ("book-toc"), any other part of the book excluding the ToC pages ("no-book-toc"), or based on the full content of the book ("full-content"). If neither of these applies, please specify or simply use "other".
- **source-files:** Specification of the source files used as input, i.e., the XML file (@xml="yes"), the PDF file (@pdf="yes"), and/or the JPEG files (@jpg="yes").
- **description:** A description of the approach used to generate the ToC. Please add as much detail as you can, as this would help with the comparison and analysis of the results later on.

Furthermore, a run should contain the search results for each topic confirming to the following criteria:

- **book:** Contains the ToC information for each book.
- **bookid:** Each book should be identified using its bookID, which is the name of the directory that contains the XML source of the book (along with the MARC metadata file).
- **toc-entry:** Contains details of each entry of the table of contents for a given book. Entries may be nested, e.g., sections in a chapter should be nested within the ToC entry of the chapter.
- **@title:** The title of the ToC entry (e.g., chapter title).
- **@page** The page counter that corresponds to the start of the section represented by the ToC entry. The page counter starts with 1 on the first page of the book (i.e., cover page). Note that this is different from the page number that may be printed in the book itself (which may only start on the first content page and may include different formats, e.g., v, xii, 2-18, etc.).

An example submission may be as follows:

```
<bs-submission participant-id="25" run-id="ToCExtractedDirectlyFromBookToC" task="book-toc" toc-
            creation="automatic" toc-source="full">
<source-files xml="yes" pdf="no" jpg="no"/>
<description>Extraction applied directly to recognised ToC pages of the book. The page numbers are
            then converted to page counters using a pre-built page lookup table. The ToC levels
            are estimated based on the layout indentation of a ToC entry.</description>
<book>
  <bookid>384D10DAEA4E34A8</bookid>
  <toc-entry title="Introduction" page="7">
    <toc-entry title="What is covered?" page="8"></toc-entry>
    <toc-entry title="Recommended reading order" page="11"></toc-entry>
  </toc-entry>
  …
</book>
<book>
  <toc-entry title="Preface" page="6"></toc-entry>
    …
</book>
…
</bs-submission>
```

6 Active reading task

6.1 Goals

The main aim of this task is to explore how hardware or software tools for reading e-books can provide support to users engaged with a variety of reading related activities, such as fact finding or learning.

6.2 Motivation

Software and hardware e-readers have moved on quite quickly with new models recently coming on the market and getting a lot of attention (e.g., Amazon's Kindle and iRex's Ilaid Reader). Researchers, from a number of related communities, are actively involved in the study and analysis of user requirements and consequently the design of more usable tools to support reading of large structured electronic documents: e-books. Progress in this area, however, suffers by the lack of common practices when it

comes to conducting usability studies. Current user studies focus on specific content and user groups and follow a variety of different procedures that make comparison, reflection and better understanding of related problems difficult. This track offers an ideal arena for researchers involved in these kinds of studies with the crucial opportunity to access a large selection of titles, representing different genres and appealing to a variety of potential users, as well as benefiting from established methodology and guidelines for organising effective evaluation experiments.

6.3 Task description

The track is based on the large evaluation experience of EBONI[1], and adopts its evaluation framework with the aim to guide participants in organising and running user studies whose results could then be compared.

The task is to run one or more user studies in order to test the usability of novel e-readers by following the provided EBONI based procedure and focusing on INEX content. Participants should then gather and analyse results according to the EBONI approach and submit these for overall comparison and evaluation.

The evaluation will be task-oriented. Participants will be able to tailor their own evaluation experiments, inside the EBONI framework, according to resources available to them. In order to gather user feedback, participants may choose from a variety of methods, from low-effort online questionnaires to more time consuming one to one interviews, and think aloud sessions.

6.4 Requirements for participation

Participants should have access to one or more software/hardware e-readers (already on the market or in prototype version) and be able to feed these with a subset of the book corpus. Titles will be provided according to participants' needs and objectives. For instance, a research group interested in validating a novel interface for e-books in education, involving history students, will be provided with content in this subject and asked to provide representative tasks. The main requirement is that each research group should have access to a representative sample of users pertinent to their research aims and objectives and would be willing to create and submit related tasks. Tasks may be of varying cognitive levels, from fact finding or memory tasks to learning or understanding tasks.

Participants will be asked to involve a minimum sample of 10 users who will be asked to complete three growing complexity tasks and fill in a customised version of the EBONI subjective questionnaire, usually taking no longer than half an hour in total, allowing to gather meaningful and comparable evidence. Additional user tasks and different methods for gathering feedback (e.g., video capture) may be added optionally.

6.5 Assumptions

Usability is an essential component of the e-reading experience. As there is not such a thing as a universal e-reader, equally usable by any user interested in any content, each user-content pair should be supported by a specifically designed e-reader.

6.6 Example research questions

How does user intent, task, purpose of reading, and genre affect usage requirements for e-readers? What kinds of user interface features are best suited for a given user task and genre of books? How does user behaviour vary across different tasks? How does usage relate to mobility? How important is it to access books on the run? What are the requirements of e-readers to support collaborative reading? How important is it to share access to the same book? How to best present an overview of a whole book to users? How can users be supported in navigating the contents of a book effectively? How can the table of contents, back-of-book index, and search functions aid different purposes of reading? What techniques provide better support for allowing users to quickly assess the relevance or usefulness of a book? What role does annotation, review, summary, etc. play in understanding and learning? Which forms of annotation are suitable for which tasks? How can active reading techniques increase recall in memory tasks?

---

[1] http://ebooks.strath.ac.uk/eboni/

7   How to submit your runs

Runs should be uploaded to http://booksearch.org.uk by the **30th of September 2010**. Login using your INEX credentials and go to the Upload Area. Please note that no online validation of your runs is offered, so please make sure that your runs conform to the appropriate DTD and that all XPaths (see Appendix A) are valid.

Appendix A: XPaths and Passages

XPath

XML element and book page paths should be given in XPath syntax[2]. To be more precise, only fully specified paths are allowed, as described by the following grammar:

```
Path ::= '/' ElementNode Path | '/' ElementNode | '/' AttributeNode
ElementNode ::= ElementName Index
AttributeNode ::= '@' AttributeName
Index ::= '[' integer ']'
```

Example:
        <path>/document[1]/page[4]/section[2]</path>

This path identifies the XML element, which can be found if we start at the document root, select the first "document" element, then within that, select the fourth "page" element, within which we select the second "section" element.

Please note that XPath counts element nodes **starting with 1** and takes into account the element type. For example, if a "page" element has a title and two sections then both the title and the first section elements would be indexed with 1 (since they are different element types). Their XPaths would be given as:

/document[1]/page[1]/title[1],
/document[1]/page[1]/section[1], and
//document[1]/page[1]/section[2].

XML and whitespace

XML is very flexible in its handling of whitespace, i.e., the following two documents are usually regarded as identical.

```
<a>
    <b />
</a>
```

```
<a><b/></a>
```

However, strictly speaking the document on the left contains whitespace content (newlines, tabs, spaces) which is not present in the document on the right. That is, the element <a> in the document on the left contains first a newline and some spaces, then an empty <b> element, and then again a newline.

When constructing passages, any whitespace that represents the **only** textual content of a text node should be ignored.

---

[2] Clark, J. and DeRose, S. 1999. XML Path Language (XPath) version 1.0. W3C Recommendation. http://www.w3.org/TR/xpath.