Library Hi Tech

A library's information retrieval system (In)effectiveness: case study

Robert Marijan Robert Leskovar

A library's information retrieval system (In)effectiveness: case study

Robert Marijan
Delo Newspaper Corporation, Ljubljana, Slovenia, and
Robert Leskovar
Faculty of Organizational Sciences, University of Maribor, Maribor, Slovenia

Received 12 February 2015
Revised 9 July 2015
Accepted 19 July 2015

Abstract

Purpose – The purpose of this paper is to evaluate the effectiveness of the information retrieval component of a daily newspaper publisher's integrated library system (ILS) in comparison with the open source alternatives and observe the impact of the scale of metadata, generated daily by library administrators, on retrieved result sets.

Design/methodology/approach – In Experiment 1, the authors compared the result sets of the information retrieval system (IRS) component of the publisher's current ILS and the result sets of proposed ones with human-assessed relevance judgment set. In Experiment 2, the authors compared the performance of proposed IRS components with the publisher's current production IRS, using result sets of current IRS classified as relevant. Both experiments were conducted using standard information retrieval (IR) evaluation methods: precision, recall, precision at k, F-measure, mean average precision and 11-point interpolated average precision.

Findings – Results showed that: first, in Experiment 1, the publisher's current production ILS ranked last of all participating IRSs when compared to a relevance document set classified by the senior library administrator; and second, in Experiment 2, the tested IR components' request handlers that used only automatically generated metadata performed slightly better than request handlers that used all of the metadata fields. Therefore, regarding the effectiveness of IR, the daily human effort of generating the publisher's current set of metadata attributes is unjustified.

Research limitations/implications – The experiments' collections contained Slovene language with large number of variations of the forms of nouns, verbs and adjectives. The results could be different if the experiments' collections contained languages with different grammatical properties.

Practical implications – The authors have confirmed, using standard IR methods, that the IR component used in the publisher's current ILS, could be adequately replaced with an open source component. Based on the research, the publisher could incorporate the suggested open source IR components in practice. In the research, the authors have described the methods that can be used by libraries for evaluating the effectiveness of the IR of their ILSs.

Originality/value – The paper provides a framework for the evaluation of an ILS's IR effectiveness for libraries. Based on the evaluation results, the libraries could replace the IR components if their current information system setup allows it.

Keywords Information retrieval, Precision, Open source software, Library systems, Recall, Apache Solr

Paper type Research paper

Introduction

Innovation in assembly is one of the key Web 2.0 principles. The principle refers to an abundance of commodity components (or pre-existing foundations), that one can use to create value by assembling them in novel or effective ways (O'Reilly, 2005; Miller, 2006). Library 2.0 is "a subset of library services designed to meet user needs caused by the direct and peripheral effects of Web 2.0" (Habib, 2006, p. 9). Openness of Library 2.0 extends to "the software and hardware that libraries use, including integrated library

© Emerald Group Publishing Limited
0737-8831
DOI 10.1108/LHT-07-2015-0071

systems (ILS)" (Casey and Savastinuk, 2006). Miller (2006) foresees the end of closed, proprietary, monolithic library software systems and emphasizes the need to "specify and build modular systems from which libraries can select the best components for a given task." The preference toward modifiable and open systems are also noted by Casey and Savastinuk (2006), Nesta and Mi (2011).

Modularity is "a special form of design which intentionally creates a high degree of independence or 'loose coupling' between component designs by standardizing component interface specifications" (Sanchez and Mahoney, 1996). If libraries used the concepts of "loose coupling" (Weick, 1976) in the design and implementation of their software systems, libraries could, "when any element misfires or decays or deteriorates," replace that element with the new one without affecting the operation of other elements. A more recent concept, service-oriented architecture, is "an architecture for building business applications as a set of loosely coupled distributed components linked together to deliver a well-defined level of service. These services communicate with each other, and the communication involves data exchange or service coordination" (Wang and Dawes, 2013). In this research, we explore the concept of "loose coupling" by evaluating the effectiveness of the information retrieval (IR) component of a daily newspaper publisher's ILS in comparison with the open source alternatives, and observe the impact of the scale of metadata attributes, generated daily by library administrators, on retrieved result sets.

We conducted two experiments. In Experiment 1, we compare the result sets of the information retrieval system (IRS) component of the current ILS with the result sets of proposed ones, using standard IR methods. Furthermore, we divide the current archive metadata attribute set into two groups: first, the "all-fields" (AF) group containing all metadata attributes; and second, the "computed-fields" (CF) group containing only automatically generated metadata attributes. For Experiment 2, we compared the performance of various Apache Solr cores relative to the current production IRS. The base core group (AF) configuration included all of the fields (human generated attributes and attributes, automatically generated upon transfer from the publisher's editorial systems to ILS). The second core group included only automatically generated CF attributes.

IR

While Van Rijsbergen (1979) presented the clear difference between data retrieval and IR, Manning *et al.* (2009, p. 1) defined the IR as "finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections (usually stored on computers)."

Herrera-Viedma (2001) defined the main activity of an IRS as "the gathering of the pertinent archived documents that best satisfy the user queries." This author parsed the process of gathering into three components: "(1) *A Database*: which stores the documents and the representation of their information contents (index terms), (2) *A Query Subsystem*: which allows users to formulate their queries by means of a query language and (3) *An Evaluation Subsystem*: which evaluates the documents for a user query obtaining a Retrieval Status Value (RSV) for each document."

Pirkola (2001) presented a morphological classification of languages from the standpoint of IR. He summarized morphology as "a field of linguistics which studies word structure and formation," and split morphology into inflectional morphology and derivational morphology (Pirkola, 2001, p. 331 cited Karlsson, 1998; Bybee, 1985; Matthews, 1991). Pirkola (2001) defined inflection as "the use of morphological methods to create inflectional word forms from a lexeme," with lexeme being defined as "a set of word forms which belong together" (Karlsson, 1983). Pirkola (2001, p. 331) associated derivational morphology with "the derivation of new words from other words using derivational affixes" and also mentioned compounding as another method for forming new words, with a compound word (or a compound) defined as "a word formed from two or more words written together." Pirkola (2001, p. 332) noted that all three main morphological phenomena (inflection, derivation and compound words) have influence on the effectiveness of text retrieval.

Airio (2006) stated that a query word and a word in a relevant document would not match because of inflection (different inflected forms of the same word). To address this problem, various word form normalization methods have been developed. Airio divided word form normalization tools into two classes: stemmers and lemmatizers. "Lemmatizers return a basic form of a word, the lemma, while stemmers return a string which is not inevitably any lexical word" (Airio, 2006, p. 250).

IR effectiveness is language-inflection depended. Airio (2006) stated that in highly inflectional languages like Finnish, German and Slovene, "normalization is without exception advantageous," while in English, "the importance of normalization is not so evident" (Airio, 2006, p. 250 cited Harman, 1991; Popovič and Willet, 1992; Krovetz, 1993).

Research multi-language usability

In our experiments, we used highly inflectional Slovene language. For instance, out of 252 derivatives of the verb "videti [to see]," 40 of them are unique (Jakopin, 1999).

We could use the same approach for any language. Hollink *et al.* (2004) conducted an overview of IR techniques and analyzed their impact on IR effectiveness. Their evaluations were carried out against data, composed of eight European languages. Hollink *et al.* (2004, p. 34 cited Harman, 1995) stated common opinion that basic IR techniques are language-independent, while "auxiliary" techniques, such as stemmers, lemmatizers and other morphological normalization tools, need to be language dependent.

In our research, IR component consisted of Linux Fedora operating system, Apache Tomcat, "an open source software implementation of the Java Servlet and JavaServer Pages technologies" (Apache Software Foundation, 2013) and Apache Solr, enterprise index and search engine (Apache Software Foundation, 2012).

From among the search engines available at the time of research, we selected Apache Solr because there is a Slovenian language-aware lemmatizer module available.

In our experiments, word form normalization tools are cores inside Apache Solr engine. Cores perform normalization methods either in index phase, query phase or in both phases. In our research, we confirmed that Slovenian language-aware cores were more effective in IR than Slovenian language-unaware one, which is consisted with the findings of Airio (2006, p. 250 cited Harman, 1991; Popovič and Willet, 1992; Krovetz, 1993).

Related work

We surveyed two research areas related to our work: first, an evaluation of IR effectiveness (user satisfaction, relevance); and second, open source software alternatives in the field of library automation.

Effectiveness evaluation

Van Rijsbergen (1979) defined effectiveness as "a measure of the ability of the system to satisfy the user in terms of the relevance of documents retrieved," while Bailey and

Pearson (1983) described user satisfaction as "in a given situation the sum of one's feelings or attitudes toward a variety of factors affecting that situation." From the perspective of information seeking Bruce (1998) suggests user satisfaction as "a state of mind which represents the composite of a user's material and emotional responses to the information-seeking context." The subjectivity and (emotional/material) situation dependency is also addressed by Rees (1965).

There have been a number of surveys reviewing the papers on relevance (Saracevic, 1975, 1976; Schamber, 1994; Mizzaro, 1997; Greisdorf, 2000; Hjørland, 2010). Doyle (1963) stated that relevance is too elusive to be a reliable criterion for evaluation (Mizzaro, 1997, p. 815), while Wilson (1968) noted that after user had read the document, he does not judge the same document as still being relevant. Wilson (1968) also presented the case of relevancy of indirect answers; if someone would ask him what he thinks about certain person's honesty, he would answer: "He has served three terms for embezzlement and two for forgery." Can we judge this answer as relevant, even though it is not explicitly answering the question? Rees (1965) mentioned that the IRS cannot perform ideally because "customer will select a different set today then he would tomorrow; if he were in Fargo, North Dakota instead of Washington, D.C., his selection might very well be different; if he examined each document in a different sequence his assessment of relevance would not be the same."

Mizzaro (1997, p. 811) summarized the relevance as a relation between two entities of two groups (Table I).

The relevance could be described as a relation between query and surrogate, in the case of Experiment 1 the "Patria" query and returned documents (surrogates) or received information (about "Patria") in relation to our information need (concerning "Patria").

The relevance is a base concept of precision/recall methods, "typified by the second series of studies conducted in Cranfield, UK (Cleverdon, 1967), which tested the relative effectiveness of 33 indexing languages on retrieval" (Harter, 1996). Manning *et al.* (2009, pp. 153-154) composed a list of the most standard test collections and evaluation series, including the aforementioned Cranfield collection, which was "a pioneering test collection in allowing precise quantitative measures of information retrieval effectiveness," Text Retrieval Conference (TREC) collections, 25 million page GOV2

Table I. Relevance as a relation between two entities of two groups

Group	Entity
A	Document: the physical entity that the user of an Information Retrieval System will obtain after his seeking of information Surrogate: a representation of a document. It may assume different forms and may be made up by one or more of the following: Title, list of keywords, author(s) name(s), bibliographic data (date and place of publication, publisher, pages and so on), abstract, extract (sentences from the document) and so on Information: what the user receives when reading a document
B	Problem: that which a human being is facing and that requires information for being solved Information need: a representation of the problem in the mind of the user. It differs from the problem because the user might not perceive his problem in the correct way Request: a representation of the information need of the user in a "human" language, usually in natural language Query: a representation of the information need in a "system" language, for instance Boolean

Source: Mizzaro (1997, p. 811)

web page collection, NII Test Collections for IR Systems, the Cross Language Evaluation Forum (focussed on European languages and cross-language IR), as well as Reuters-21578 and Reuters-RCV1. Since 1999, TREC conferences have also conducted a question answering track, whose goal is "to foster research on systems that directly return answers, rather than documents containing answers, in response to a natural language question. A factoid question is a fact-based, short answer question such as "How many calories are there in a Big Mac?" (Dang *et al.*, 2008).

Open source replacement alternatives

In contrast to our proposal to replace only a segment of ILS, numerous research studies were conducted in assessing the adequacy of the open source alternatives to replace market-dominated proprietary (commercial) ILSs as a whole.

Müller (2011) analyzed 20 free and open source (FOSS) ILS platforms using a three-stage evaluation method (software licensing, community and functionality). His top ranked ILSs were Koha, Evergreen and PMB. He ranked Koha as the most functionally complete ILS. Evergreen's most noteworthy properties were the quality of feature implementation and robust construction, while PMB included "all necessary basic functions of library automation, especially concerning integration of Web 2.0-oriented features and other patron-based Web services" (Müller, 2011, p. 15).

Koha and Evergreen are the most widely adopted (Palmer and Choi, 2014) and in the Singh's (2013) interviews the most frequently migrated-to ILSs. Singh (2013) also composed the guidelines spanning from the "Evaluation" to the "Go live and after" stages. Randhawa (2013) advised library and information science professionals to monitor the evolution of open source ILSs (Koha, Evergreen and few others) and choose the right product depending upon institution's needs. Randhawa (2013) also noted the need for library professionals to acquire new skills for developing and managing the library by using open source software. In comparative study, Reddy (2013) concluded that Koha (and NewGenLib) were suitable for bigger libraries and functionally superior to E-Granthalaya. Yang *et al.* (2009) made a comparison of staff modules of Koha, Evergreen and proprietary Voyager ILS. Yang and Hofmann (2010) also conducted the comparative study of the online catalogs (OPAC) of the same three ILSs. In addition to extensive community support there is worldwide paid support available for both Koha (Koha Library Software, 2015) and Evergreen (Evergreen, 2015).

Brooke's (2013) examination showed financial, functional and operational benefits of using FOSS software. He noted the trend of more libraries switching to a FOSS ILS, "with 14% using one as of 2012." That percentage could be lower because although the survey was broad, the sample of libraries may not be sufficiently large to be representative of the library population as a whole. Yang *et al.* (2009) stated that in 2009 the 400 libraries were using Koha, 305 Evergreen and 1179 proprietary Voyager ILS. At the same time (April 2009), only about 20 out of 15,000 libraries in Italy used open source library automation systems (Frigimelica, 2009). Some libraries had tried it and then switched back to proprietary ILS (Rapp, 2011) with the explanation that the product was not (yet) being ready for production use. Rapp stated that implementation of open source ILS took a significant amount of work, but gave users more control and concluded that "the effectiveness of an ILS depends in large part on the needs and expectations of the library that uses it." Palmer and Choi (2014) stated that the extent of using open source software in the library community is "still a matter for debate," as in addition to the positive qualities there are also the "concerns among librarians regarding the dependency and sustainability of open source software products." Kinner and Rigda (2009) concluded that there were many articles about open source software implementations, but very few case studies showing success.

Data and methods

Experiment setup

We used a dedicated computer with Intel i7 2600 processor with 32 GB RAM and a Kingston 120 GB SSD drive. The operating system was Linux Fedora Core 19 3.14.4 64bit, database management system was MySQL 15.1; 5.5.37-MariaDB, Apache Tomcat 7.0.47, Apache Solr 4.6.0, PHP 5.5.12. An overview of Experiment 1 is shown in Figure 1.

Data collections

For Experiment 1, we used two data collections (Table II). The "Patria 2006-2013 Dossier" is an ad hoc human-assessed relevance judgment set, used in the process of effectiveness evaluation (Figure 1) as a criterion for a true or false hits.

For Experiment 2, we used query set aggregated from: first, search strings that users of publisher's digital editions inserted in one of the search fields; second, keywords that

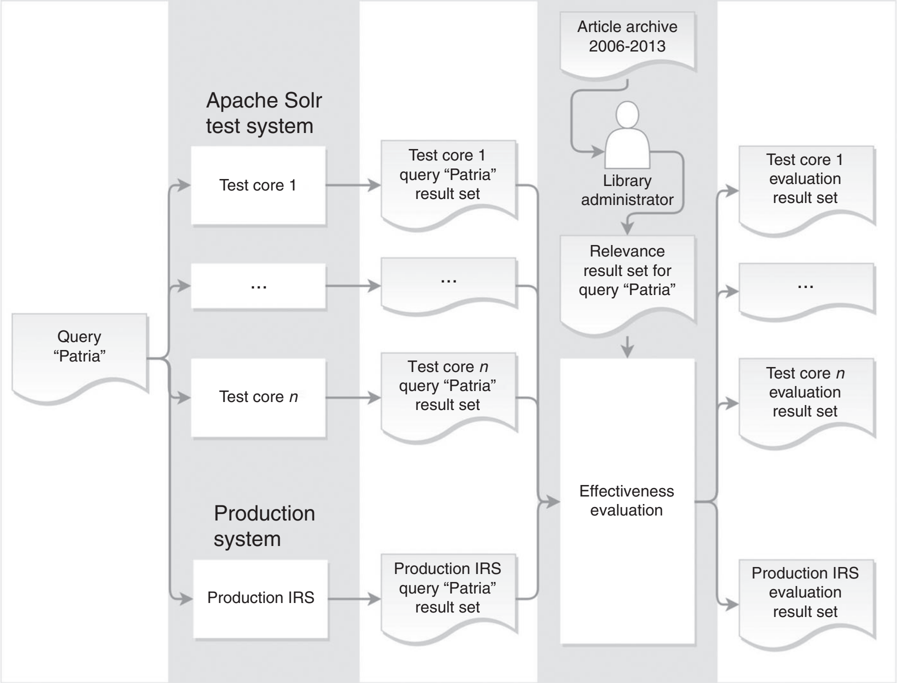

Figure 1. Experiment 1 overview

Table II. Experiment 1 data collection

Experiment 1 data collection	Number of articles
Patria 2006-2013 Dossier	1,439
Daily newspaper article archive 2006-2013	615,831

journalists inserted inside metadata fields of the articles; and third, keywords that library administrators inserted during search for certain topics (Table III).

We performed a three-phase query validation. In Phase 1 of the validation, we removed irregular strings (such as "????????re"); in Phase 2 of the validation, we extracted unique keywords; in Phase 3 of the validation, we discarded keywords with zero returned documents from the production IRS.

After the analysis, we concluded that most of the discarded keywords in Phase 3 were mistyped strings ("aidria airways" instead of "adria airways").

Experiment 2 was conducted using 41,673 validated queries.

For data collection, we used the publisher's article archive from May 1, 1959 to July 31, 2014 (Table IV).

Workflow

In Experiment 1, we imported the daily newspaper's article archive 2006-2013 data collection from publisher's production IRS to a MySQL database, and performed a data import (full-import; clean; commit; optimize) to three different cores in Apache Solr. The 2006-2013 time frame was selected due to its relevance to the "Patria" events.

In Experiment 2, we imported the daily newspaper article archive 1959-2014 data collection from publisher's production IRS to MySQL database, and again performed a data import (full-import; clean; commit; optimize) to three different cores in Apache Solr. Indexing times and sizes, as reported by Apache Solr Admin 4.6.0, are shown in Table V.

Participating cores

Virag (LemmaGen Slovene Lemmatizer module) is LGPLv2 licensed core, programmed by Virag (2013), Domen Grabec and Gašper Žejn, based on the LemmaGen project of the Jozef Stefan Institute (2010). Coding is based on JLemmagen, a Java port of the LemmaGen library by Michal Hlaváč.

	Number of keywords (search strings), entering the phase			
Query collection	Validation phase 1	Validation phase 2	Validation phase 3	Validated query set
Queries July 2014	1,400,000+	666,657	43,084	41,673

Table III. Experiment 2 query collection

Experiment 2 data collection	Number of articles
Daily newspaper article archive 1959-2014	1,328,214

Table IV. Experiment 2 data collection

	Core	Slovenian-language aware	Experiment 1: 615,831 articles Index time	Index size	Experiment 2: 1,328,214 articles Index time	Index size
1	Virag	Yes	00 h 49 m 27 s	8.03 GB	01 h 32 m 59 s	13.97 GB
2	Hunspell	Yes	17 h 09 m 57 s	3.13 GB	31 h 41 m 48 s	5.89 GB
3	Collection_1	No	00 h 06 m 24 s	2.58 GB	00 h 14 m 35 s	4.85 GB

Table V. Apache Solr cores and indexing times and sizes

Hunspell is "a spell checker and morphological analyzer library and program designed for languages with rich morphology and complex word compounding or character encoding" (Németh, 2014). It is an LGPL, GPL, MPL tri-licensed core. Hunspell's code base comes from the MySpell spell checker. We used Slovenian dictionary files from LibreOffice (2013).

Collection_1 is Apache Solr's default core; it is Slovenian language unaware and uses default (English) language rules. It is licensed under the Apache License, Version 2.0 (Apache Software Foundation, 2012).

Apache Solr setup

We were using Apache Solr 4.6.0 with Extended DisMax Query Parser (eDisMax), a "robust parser designed to process advanced user input directly" (Pugh, 2013).

For the indexing phase, we used the cores' default setup with "fields" defined in the same way for all cores; only the special Slovenian language-aware type was defined differently for each core.

For the query phase, we defined two request handlers, "AFs" handler containing all of the fields in "*qf*" parameter, and the "CFs" handler with the fields, computed automatically upon transfer from the publisher's Editorial Information Systems to the publisher's Library Information System every night (for differences see Table VI).

"For each 'word' in query string, eDisMax parser builds a DisjunctionMaxQuery object for that word across all of the fields in the "*qf*" parameter (with the appropriate boost values)" (Pugh, 2013). In our research, only the title was "boosted" with a 1.5 value.

"These DisjunctionMaxQuery objects are then put in a BooleanQuery with the minNumberShouldMatch option set according to the 'mm' parameter" (Pugh, 2013). We set the "mm" parameter to "$2 < -34\%$" meaning that: first, if there are less than three optional clauses, they all must match; and second, if there are three or more optional clauses, then 66 percent must match.

In this iteration, we did not use the "phrase" fields, which "boost" the score of documents in cases in which all of the terms in the "*q*" parameter appear in close proximity (Pugh, 2013).

Evaluation

For set-based measures Manning *et al.* (2009, p. 155) reproduced definitions of precision (p) as the fraction of retrieved documents that are relevant (Equation (1), top left), recall (r) as the fraction of relevant documents that are retrieved (Equation (1), right), and F-measure as a single measure that trades off precision vs recall; a weighted harmonic mean of precision and recall (Equation (1), bottom). In the latter, if β is less than 1, we emphasize precision while in cases in which $\beta = 5$ (in TREC 2003) or $\beta = 3$ (in TREC 2004) (Lin and Demner-Fushman, 2005), we emphasize recall. Elements constituting

Table VI. Difference in included fields by each request handler

Request handler	Fields
Fields included in AF and CF handlers	Supertitle, Title^1.5 (with 1.5 boost value), Subtitle, Source, Section, Authors, Snippet, Signature, Tags, WebCategory, Ref_Title, Ref_Subtitle, Ref_Intro, HtmlContent
AF exclusive fields, inserted daily by library administrators	GeoNames, Genre, Persons, Quoted, Nouns, Udks, UdkCategory

precision/recall equations can be examined in the contingency table below (Table VII):

$$p = \frac{TP}{TP+FP} \quad r = \frac{TP}{TP+FN}$$

$$F_\beta = \left(1+\beta^2\right)\frac{pr}{r+\beta^2 p} = \frac{\left(1+\beta^2\right)TP}{\left(1+\beta^2\right)TP+\beta^2 FN+FP} \tag{1}$$

Equation 1. Precision, recall and *F*-measure (Source: Manning *et al.* (2009, p. 155))

For the evaluation of ranked retrieval results, we used interpolated precision $P_{inter\ p}$ (Manning *et al.*, 2009, pp. 158-159) at a certain recall level *r*, which is defined as the highest precision found for any recall level $r' \geqslant r$ (Equation (2), left). We used interpolated precision to plot an 11-point interpolated average precision graph for both experiments.

Manning *et al.* (2009, pp. 159-160) stated that the "most standard [measure] among the TREC community is *Mean Average Precision* (MAP), which provides a single-figure measure of quality across recall levels. Among evaluation measures, MAP has been shown to have especially good discrimination and stability. For a single information need, average precision is the average of the precision value obtained for the set of top *k* documents existing after each relevant document is retrieved, and this value is then averaged over information needs":

$$p_{inter\ p}(r) = \max_{r' \geqslant r} p(r') \quad mean\ average\ precision = \frac{1}{n}\sum_{q_i} Ave\ Precision(q_i) \tag{2}$$

Equation 2. Interpolated precision $P_{inter\ p}$ and mean average precision (Source: Manning *et al.* (2009, pp. 158-160))

We also measured the precision after *k* documents retrieved (precision @ 10 and precision @ 20) vs every query. Precision at *k* "has the advantage of not requiring any estimate of the size of the set of relevant documents but the disadvantages that it is the least stable of the commonly used evaluation measures and that it does not average well, since the total number of relevant documents for a query has a strong influence on precision at *k*" (Manning *et al.*, 2009, p. 161).

Results and discussion

Experiment 1

For Experiment 1, we had a test collection consisting of: first, a collection of documents; second, a set of information needs or queries; and third, relevance judgments.

In case of using precision and recall evaluation methods, "it is necessary to analyze the entire document collection, and for each query determine the documents that are relevant. This judgment of whether a document is relevant or not must be done by an expert on the field that can understand the need represented by the query" (Middleton and Baeza-Yates, 2007). This classification is referred to "as the gold standard or

	Relevant	Non-relevant
Retrieved	True positives (TP)	False positives (FP)
Not retrieved	False negatives (FN)	True negatives (TN)

Source: Cleverdon (1967)

Table VII. Contingency table

ground truth judgment of relevance" (Manning *et al.*, 2009, p. 152). In our case, it was necessary to analyze all the documents that are relevant to the information need concerning legal process and related events of the Slovenian Department of Defense's procurement and payment of armored vehicles from Finland's state-owned (73.2 percent) corporation Patria, "provider of defense, security and aviation life-cycle support services and technology solutions" (Patria, 2014). The events that took place from 2006 to 2013 (and yet not fully completed) are known in Slovenia as "Patria." Even though the latter set of classified documents were binary defined as relevant by senior library administrator, this is still a sole human judge, and, as such, "idiosyncratic and variable" (Manning *et al.*, 2009, p. 165) in his judgments.

The quality of relevance judgments could be improved by adding more judges. In the case of multiple judges we would use κ-statistics (Manning *et al.*, 2009, p. 165) to measure the agreement between their judgments.

Observing the Summary Statistics table for Experiment 1 (Table VIII), we can see that the IRSs returned between 2,400 and 3,000 documents; only the Hunspell core returned approximately 4,500. The latter AF core returned the most relevant documents (1,256, 87 percent).

The IRS with the largest count of not retrieved relevant documents was the production IRS.

There are situations in which we need high precision; for instance, in web searches, we want relevant documents listed on the first returned result set page that is visible without scrolling down (Lewandowski, 2014). Conversely, when we search for the patents in a certain scientific field, we want the returned result set to include all of the relevant documents (high recall).

The large majority of visitors are using search fields inside publisher's digital editions, so precision is important. As we can see in Table IX, the production IRS ranked last in precision at ten and precision at 20.

The production IRS was also ranked last in all the other measures, precision, recall, *F*-score and average precision.

There is a notable difference in "AFs" vs "CFs" request handler performance. In the case of "Patria," there is the metadata field "Nouns," which explicitly contains the string "Patria," inserted by a human library administrator. Because it is human generated metadata, it is not included in "CFs" request handler, which explains performance superiority of AF cores vs their CF counterparts.

The substandard performance of production IRS could be observed on an 11-point interpolated average precision graphs in Figures 2 and 3.

Table VIII.
Summary statistics for Experiment 1

Core	Retrieved total	Retrieved relevant	% Retrieved relevant	Not retrieved total	Not retrieved relevant	% Not retrieved relevant	Retrieved non-relevant	Not retrieved non-relevant
Production IRS	2,847	681	0.47	612,984	758	0.53	2,166	612,226
Virag AF	2,986	1,231	0.86	612,845	208	0.14	1,755	612,637
Hunspel_1 AF	4,667	1,256	0.87	611,164	183	0.13	3,411	610,981
Collection_1 AF	2,830	1,206	0.84	613,001	233	0.16	1,624	612,768
Virag CF	2,704	996	0.69	613,127	443	0.31	1,708	612,684
Hunspel_1 CF	4,481	1,101	0.77	611,350	338	0.23	3,380	611,012
Collection_1 CF	2,495	931	0.65	613,336	508	0.35	1,564	612,828

Evaluation of Experiment 1 (query "Patria" vs seven cores/IRSs) computed in less than a minute.

Experiment 1 showed that:

- production IRS performance ranked last of all participated cores/IRSs;
- even though there is a human relevance-judged document set (which is a rare and expensive time-consuming effort) a large majority of users visiting publisher's electronic editions do not benefit from it; furthermore, they obtain mediocre results from using production IRS;
- the free, open source Apache Solr cores performed very well (Virag AF); and
- there is a significant difference between AF vs CF request handler performance, so the human effort of generating the current scope of fields is justified in the case of the "Patria" experiment.

Core	Precision @ 10	Precision @ 20	Precision	Recall	F1	Average precision
Production IRS	0	0.1	0.24	0.47	0.32	0.36
Virag AF	0.5	0.65	0.41	0.86	0.56	0.77
Hunspell AF	0.1	0.25	0.27	0.87	0.41	0.67
Collection_1 AF	0.5	0.65	0.43	0.84	0.57	0.76
Virag CF	0.5	0.65	0.37	0.69	0.48	0.67
Hunspell CF	0.5	0.45	0.25	0.77	0.37	0.58
Collection_1 CF	0.5	0.65	0.37	0.65	0.47	0.66

Table IX. Precision @ 10, @ 20, precision, recall, *F*-measure and average precision for Experiment 1

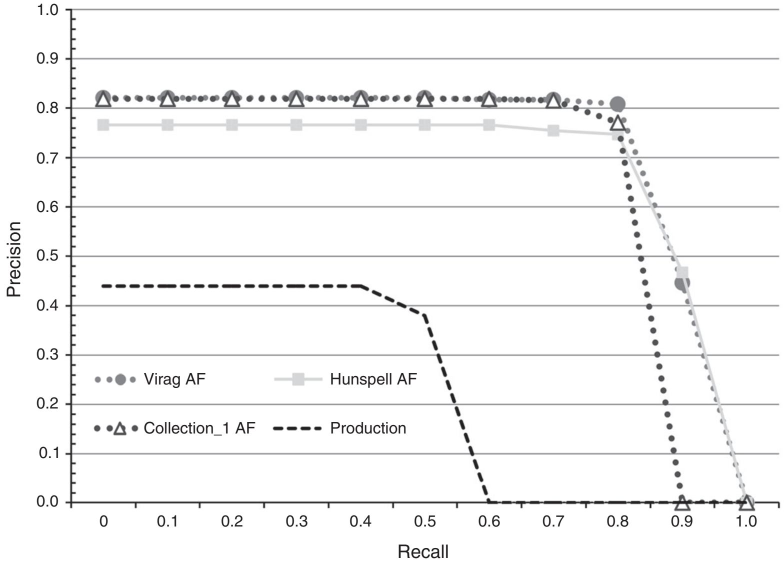

Figure 2. 11-point interpolated average precision of query "Patria", AF cores vs production IRS

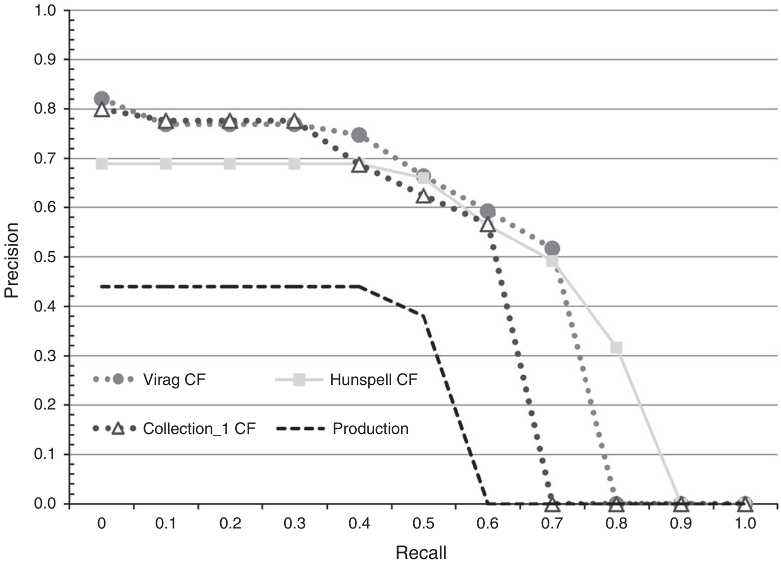

Figure 3.
11-point interpolated average precision of query "Patria", CF cores vs production IRS

Experiment 2

For Experiment 2, unlike Experiment 1, we did not have any human relevance judgment set. Because of the size of a query set and lack of ad hoc dossiers (only 94 available in July 2014), we assumed that the publisher's current IRS always returns the gold standard result set, meaning that all returned documents are classified as relevant.

We run 41,673 queries against six different cores/IRSs to establish the difference in performance relative to current production IRS.

There are known pooling and evaluation methods using incompletely judged sets, but it would still be impossible due to our resources (time, cost) to human judge even a fraction of results returned for every query. In that case, if the examined subset contained a representative sample of the relevant documents, the pooling method would closely approximate the results of judging the entire collection (Soboroff *et al.*, 2001).

In Table X, we can see that the Virag cores returned on average the most relevant documents (and the least relevant left unretrieved), while the Collection_1 performed just the opposite. We can also note that Collection_1 cores returned on average the fewest total documents (approximately 4,000 vs 11,000/18,000).

Table X.
Summary statistics for Experiment 2

Core	Average retrieved total	Average retrieved relevant	Average retrieved non-relevant	Average not retrieved relevant
Virag AF	11,127.40	1,908.80	9,218.60	423.06
Virag CF	11,006.20	1,907.95	9,098.28	423.90
Hunspell AF	18,549.80	1,869.01	16,680.80	462.84
Hunspell CF	18,397.50	1,867.44	16,530.10	464.41
Collection_1 AF	3,992.32	764.61	3,227.70	1,567.24
Collection_1 CF	3,916.61	759.78	3,156.83	1,572.07

Observing Table XI, we see that Collection_1 cores achieved the highest mean average precision score, which is oriented toward favoring early results (Chatzichristofis *et al.*, 2014). If we observe two result sets, first r_set(*q*) = {*d*1, *d*2, *d*3, *d*4, *d*5, *d*6}, in which all the documents are classified as relevant except *d*2 (the second retrieved document), and r_set'(*q*) = {*d*1, *d*2, *d*3, *d*4, …, *d*100}, in which documents from *d*5 to *d*99 are classified as non-relevant, and the fifth relevant document is at the very end of the set (*d*100), the value of the average precision is the same for both queries: 0.81 (Chatzichristofis *et al.*, 2014, p. 21). Precision is defined as the fraction of retrieved documents that are relevant, but in the case of the Collection_1 cores the proportion of averagely not retrieved relevant documents (false negatives) was the highest and the average recall was the lowest.

We compared the results of the 11-point interpolated average precision graph (Figure 4) observing three different recall ranges (NIST, 2002): first, 0 to 0.2 for high precision; second, 0.2 to 0.8 for middle recall; and third, 0.8 to 1 for high recall performance.

The performance ratio between each core pairs (AF and CF) gradually differentiated by every recall point, with the highest distinction point in recall value 1. We can rank the participating cores by precision/recall performance as Virag (both cores), Hunspell (both cores) and Collection_1 (both cores). CF-only request handler primes slightly

Core	Average precision @ 10	Average precision @ 20	Average precision	Average recall	Average F1	Mean average precision
Virag AF	0.59	0.54	0.42	0.88	0.52	0.54
Virag CF	0.66	0.60	0.43	0.88	0.52	0.58
Hunspell AF	0.58	0.52	0.40	0.81	0.46	0.52
Hunspell CF	0.65	0.58	0.40	0.81	0.46	0.55
Collection_1 AF	0.62	0.54	0.46	0.54	0.43	0.60
Collection_1 CF	0.67	0.59	0.47	0.54	0.43	0.63

Table XI. Average precision @ 10, @ 20, average precision, average recall, average *F*-measure and mean average precision for Experiment 2

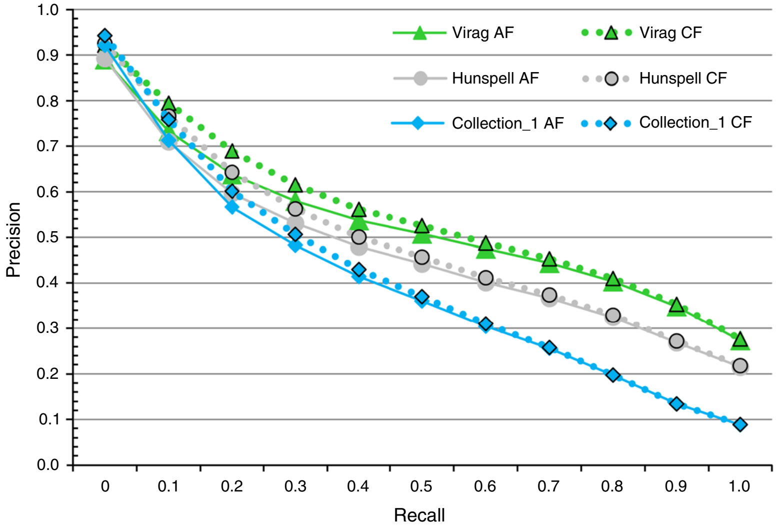

Figure 4. 11-point interpolated average precision for Experiment 2

outperformed their AF counterparts, especially to recall point 0.6; after 0.7 the cores performed almost identically, although CF primes retaining tiny advantage.

Experiment 2 showed that:

- in contrast to Experiment 1 in which there was a significant difference between AF vs CF request handler performances in favor of former, Experiment 2 showed that the request handlers that used only automatically generated metadata performed slightly better; therefore, human effort in generating the current scope of fields is not justified.

Conclusion

Using standard IR methods, we confirmed that the IR component used in the current publisher's ILS could be adequately replaced with an open source component. Based on our research, the publisher could incorporate suggested open source IR components in practice.

In the research, we described the methods that could be used by libraries for evaluating the IR effectiveness of their ILSs. We used available open source software in compliance with licensing for individual software components. Practical implementation depends on libraries' current information system setup and to what extent their information system design incorporates modular paradigm.

In the research, we addressed the technological part of replacing an IR component. Even though the software components that we used in experiments are free, there are still costs associated with open source software implementation. Breeding (2008) raised the question whether the adoption of an open source ILS results in lower or higher cost to the library. The services performed by the commercial firms in the support of open source software might include (Breeding, 2008, pp. 11-12):

- data conversion;
- installation;
- software configuration;
- library staff training;
- hosting; and
- custom development.

We must also take into consideration the legal aspect, in cases in which there is a possibility that the library has no right to modify the system in any way, even if the ILS itself is developed and implemented modularly, and technically suitable for the modification.

Libraries have to focus on their users and "survey, quantify, question and measure anticipated impacts and results before expending limited resources of time, money and people on projects that are not wanted, not needed, or not used" (Nesta and Mi, 2011). In implementation process, libraries should follow the next 12 steps (Dubowski, 2003): "(1) Assemble the project team. (2) Define the goals. (3) Document the important savings or revenue and productivity increases. (4) Draft a requirements document that lists the features needed. (5) Research potential solutions. (6) Submit Requests for Proposal (RFP) to five or six vendors that summarizes requirements and describes preferred methods of response. (7) Analyze the proposals. (8) Vendor's demo. (9) Conduct reference checks. (10) Gather up the strengths and weaknesses of each proposal and make the decision. (11) Negotiate the contract. (12) Write an implementation plan."

Libraries need to exercise prudence in the process of evaluating resources vs needs (Yang *et al.*, 2009) and depending upon libraries' needs choose appropriate technology (Randhawa, 2013). Open source ILSs are not universal answer to every problem, they are viable solution for some libraries, but not for all (Kinner and Rigda, 2009; Rapp, 2011).

References

Airio, E. (2006), "Word normalization and decompounding in mono- and bi-lingual IR", *Information Retrieval*, Vol. 9 No. 3, pp. 249-271.

Apache Software Foundation (2012), "Apache Solr", available at: http://lucene.apache.org/solr/ (accessed November 6, 2013).

Apache Software Foundation (2013), "Apache Tomcat", available at: http://tomcat.apache.org/ (accessed November 7, 2013).

Bailey, J.E. and Pearson, S.W. (1983), "Development of a tool for measuring and analyzing computer user satisfaction", *Management Science*, Vol. 29 No. 5, pp. 530-545.

Breeding, M. (2008), "The commercial angle", *Library Technology Reports*, Vol. 42 No. 8, pp. 11-15.

Brooke, T. (2013), "Open source integrated library systems in public libraries", *SLIS Student Research Journal*, Vol. 3 No. 2, Article 3.

Bruce, H. (1998), "User satisfaction with information seeking on the internet", *Journal of the American Society for Information Science*, Vol. 49 No. 6, pp. 541-556.

Bybee, J.L. (1985), *Morphology: A Study of the Relation between Meaning and Form*, John Benjamins, Amsterdam and Philadelphia, PA.

Casey, M.E. and Savastinuk, L.C. (2006), "Library 2.0: service for the next-generation library", *Library Journal*, Vol. 131 No. 1, pp. 40-42.

Chatzichristofis, S.A., Iakovidou, C., Boutalis, Y.S. and Angelopoulou, E. (2014), "Mean normalized retrieval order (MNRO): a new content-based image retrieval performance measure", *Multimedia Tools and Applications*, Vol. 70 No. 3, pp. 1767-1798.

Cleverdon, C. (1967), "The Cranfield tests on index language devices", *Aslib Proceedings*, Vol. 19 No. 6, pp. 173-194.

Dang, H.T., Kelly, D. and Lin, J. (2008), "Overview of the TREC 2007 question answering track", in Voorhees, E. and Buckland, L.P. (Eds), *TREC2007, Proceedings of the 16th Text Retrieval Conference*, GPO, Washington, DC.

Doyle, L.B. (1963), *Is Relevance an Adequate Criterion in Retrieval System Evaluation?*, System Development Corp., Santa Monica, CA.

Dubowski, S. (2003), "Recovery from IT project madness in a few steps", *Network World Canada*, Vol. 13 No. 10.

Evergreen (2015), "Commercial companies and non-profits that advertise evergreen services", available at: http://evergreen-ils.org/dokuwiki/doku.php?id=faqs:evergreen_companies (accessed June 27, 2015).

Frigimelica, G. (2009), "La diffusione di software open source per la Gestione di Biblioteche in Italia", *Biblioteche oggi*, Vol. 27 No. 6, pp. 37-43.

Greisdorf, H. (2000), "Relevance: an interdisciplinary and information science perspective", *Informing Science*, Vol. 3 No. 2, pp. 67-72.

Habib, M.C. (2006), "Toward academic library 2.0: development and application of a library 2.0 methodology", School of Information and Library Science, University of North Carolina, Chapel Hill, NC, available at: http://dc.lib.unc.edu/cdm/ref/collection/s_papers/id/905 (accessed May 2, 2014).

Harman, D. (1991), "How effective is suffixing?", *Journal of the American Society for Information Science*, Vol. 42 No. 1, pp. 7-15.

Harman, D.K. (1995), "Overview of the third text retrieval conference (TREC-3)", *Proceedings of the Third Text Retrieval Conference (TREC-3), NIST Special Publication*, pp. 1-20.

Harter, S.P. (1996), "Variations in relevance assessments and the measurement of retrieval effectiveness", *Journal of the American Society for Information Science*, Vol. 47 No. 1, pp. 37-49, 38.

Herrera-Viedma, E. (2001), "An information retrieval model with ordinal linguistic weighted queries based on two weighting elements", *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, Vol. 9, Supplement, pp. 77-88.

Hjørland, B. (2010), "The foundation of the concept of relevance", *Journal of the American Society for Information Science and Technology*, Vol. 61 No. 2, pp. 217-237.

Hollink, V., Kamps, J., Monz, C. and de Rijke, M. (2004), "Monolingual document retrieval for European languages", *Information Retrieval*, Vol. 7 Nos 1/2, pp. 33-52.

Jakopin, P. (1999), "Upper bound of entropy in Slovenian literary texts", PhD diss., Faculty of Electrical Engineering, University of Ljubljana, Ljubljana.

Jozef Stefan Institute (2010), "LemmaGen: multilingual open source lemmatisation", available at: http://lemmatise.ijs.si (accessed November 28, 2013).

Karlsson, F. (1983), *Suomen kielen aanne- ja muotorakenne (Phonological and Morphological Structures in Finnish)*, WSOY, Porvoo and Helsinki.

Karlsson, F. (1998), *Yleinen Kielitiede (General Linguistics)*, Helsinki University Press, Helsinki.

Kinner, L. and Rigda, C. (2009), "The integrated library system: from daring to dinosaur?", *Journal of Library Administration*, Vol. 49 No. 4, pp. 401-417.

Koha Library Software (2015), "Paid support", available at: http://koha-community.org/support/paid-support/ (accessed June 27, 2015).

Krovetz, R. (1993), "Viewing morphology as an inference process", *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Seattle, WA*, pp. 191-202.

Lewandowski, D. (2014), "Evaluating the retrieval effectiveness of web search engines using a representative query sample", arXiv preprint arXiv: 1405.2210.

LibreOffice (2013), "LibreOffice the document foundation", available at: www.libreoffice.org/ (accessed November 30, 2013).

Lin, J. and Demner-Fushman, D. (2005), "Evaluating summaries and answers: two sides of the same coin?", *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pp. 41-48.

Manning, C.D., Raghavan, P. and Schütze, H. (2009), *An Introduction to Information Retrieval*, Cambridge University Press, Cambridge.

Matthews, P.H. (1991), *Morphology*, Cambridge University Press, Cambridge.

Middleton, C. and Baeza-Yates, R. (2007), *A Comparison of Open Source Search Engines*, Universitat Pompeu Fabra, Barcelona.

Miller, P. (2006), "Library 2.0: the challenge of disruptive innovation", available at: http://ngl.gcg.ac.uk/pdf/447_Library_2_prf1.pdf (accessed March 28, 2014).

Mizzaro, S. (1997), "Relevance: the whole history", *Journal of the American Society for Information Science*, Vol. 48 No. 9, pp. 810-832, 811.

Müller, T. (2011), "How to choose a free and open source integrated library system", *OCLC Systems & Services: International Digital Library Perspectives*, Vol. 27 No. 1, pp. 57-78.

Németh, L. (2014), "Hunspell", available at: http://hunspell.sourceforge.net/ (accessed November 30, 2014).

Nesta, F. and Mi, J. (2011), "Library 2.0 or library III: returning to leadership", *Library Management*, Vol. 32 Nos 1/2, pp. 85-97.

NIST (2002), "NIST special publication 500-250: the tenth text retrieval conference (TREC 2001), appendices: common evaluation measures", available at: http://trec.nist.gov/pubs/trec10/t10_proceedings.html (accessed May 16, 2014).

O'Reilly, T. (2005), "What is Web 2.0: design patterns and business models for the next generation of software", available at: http://oreilly.com/web2/archive/what-is-web-20.html (accessed April 2, 2014).

Palmer, A. and Choi, N. (2014), "The current state of library open source software research: a descriptive literature review and classification", *Library Hi Tech*, Vol. 32 No. 1, pp. 11-27.

Patria (2014), "About Patria", available at: www.patria.fi/EN/About+Patria/index.html (accessed May 5, 2014).

Pirkola, A. (2001), "Morphological typology of languages for IR", *Journal of Documentation*, Vol. 57 No. 3, pp. 330-348.

Popovič, M. and Willet, P. (1992), "The effectiveness of stemming for natural-language access to Slovene textual data", *Journal of the American Society for Information Science*, Vol. 43 No. 5, pp. 384-390.

Pugh, E. (Ed.) (2013), "ExtendedDisMax", available at: https://wiki.apache.org/solr/ExtendedDisMax (accessed November 7, 2013).

Randhawa, S. (2013), "Open source library management software", *E-Library Science Research Journal*, Vol. 1 No. 7.

Rapp, D. (2011), "Open source reality check", *Library Journal*, Vol. 136 No. 13, pp. 34-36.

Reddy, C.S.V. (2013), "Comparative study of free/open source integrated library management systems (fosilms) with reference to koha, newgenlib and E-granthalaya", *E-Library Science Research Journal*, Vol. 1 No. 12.

Rees, A.M. (1965), "The evaluation of retrieval systems", Comparative systems laboratory technical Report No. 5, Center for Documentation and Communication Research, School of Library Science, Western Reserve University, Cleveland.

Sanchez, R. and Mahoney, J.T. (1996), "Modularity, flexibility, and knowledge management in product and organization design", *Strategic Management Journal*, Vol. 17, Special Issue, pp. 63-76.

Saracevic, T. (1975), "Relevance: a review of and a framework for the thinking on the notion in information science", *Journal of the American Society for Information Science*, Vol. 26 No. 6, pp. 321-343.

Saracevic, T. (1976), "Relevance: a review of the literature and a framework for thinking on the notion in information science", *Advances in Librarianship*, Vol. 6, pp. 79-138.

Schamber, L. (1994), "Relevance and information behavior", *Annual Review of Information Science and Technology*, Vol. 29, pp. 3-48.

Singh, V. (2013), "Experiences of migrating to open source integrated library systems", *Information Technology and Libraries*, Vol. 32 No. 1, pp. 36-53.

Soboroff, I., Nicholas, C. and Cahan, P. (2001), "Ranking retrieval systems without relevance judgments", *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 66-73.

Van Rijsbergen, C.J. (1979), *Information Retrieval*, 2nd ed., Butterworth, London.

Virag, J. (2013), "Slovene lemmatization in Solr", available at: www.virag.si/2013/07/slovene-lemmatization-in-solr/ (accessed November 28, 2013).

Wang, Y. and Dawes, T.A. (2013), "The next generation integrated library system: a promise fulfilled", *Information Technology and Libraries*, Vol. 31 No. 3, pp. 76-84, 79.

Weick, K.E. (1976), "Educational organizations as loosely coupled systems", *Administrative Science Quarterly*, Vol. 21 No. 1, pp. 1-19.

Wilson, P. (1968), *Two Kinds of Power: An Essay on Bibliographical Control*, University of California Press, Oakland, CA.

Yang, S.Q. and Hofmann, M.A. (2010), "The next generation library catalog: a comparative study of the OPACs of koha, evergreen, and voyager", *Information Technology and Libraries*, Vol. 29 No. 3, pp. 141-150.

Yang, S., Hofmann, M.A. and Weeks, M. (2009), "Koha, evergreen, and voyager: a comparison of their staff modules", *Ten Years of Experience, A Future of Possibilities, VALE/NJ ACRL/NJLA CUS Tenth Annual Users' Conference, Rutgers University*, *Piscataway, NJ*.

Corresponding author
Robert Marijan can be contacted at: robert.marijan@delo.si

This article has been cited by:

1. Wei Yu Jiangsu Key Laboratory of Data Engineering and Knowledge Service, Information Management School, Nanjing University, Nanjing, China Junpeng Chen School of Information Engineering, Nanjing University of Finance and Economics, Nanjing, China . 2016. Constructing linkage between libraries and up-to-date news. *Library Hi Tech* **34**:2, 301-313. [Abstract] [Full Text] [PDF]