



A Model for Content Enrichment of Institutional Repositories Using Linked Data

Vinit Kumar

To cite this article: Vinit Kumar (2018) A Model for Content Enrichment of Institutional Repositories Using Linked Data, Journal of Web Librarianship, 12:1, 46-62, DOI: [10.1080/19322909.2017.1392271](https://doi.org/10.1080/19322909.2017.1392271)

To link to this article: <https://doi.org/10.1080/19322909.2017.1392271>



Published online: 06 Dec 2017.



Submit your article to this journal [↗](#)



Article views: 186



View related articles [↗](#)



View Crossmark data [↗](#)



A Model for Content Enrichment of Institutional Repositories Using Linked Data

Vinit Kumar 

School of Library and Information Science, Central University of Gujarat, Gandhinagar, Gujarat, India

ABSTRACT

Institutional repositories have positioned themselves as an essential service for many libraries. Content-enriched metadata in library records is reported as being helpful to library users in identifying and selecting information objects for their needs. The presence of this extra-enriched content helps users to decide on the relevance of the item without the need to access the full text. Through this paper, we report content enrichment of records in an Institutional Repository using Linked Open Data datasets. In particular, this is done by application of a linked dataset in an institutional repository. The design, implementation, configuration, and workflow of the application is discussed along with implications and potential future work.

ARTICLE HISTORY

Received 20 June 2017
Accepted 11 October 2017

KEYWORDS

Institutional Repository;
Linked Open Data; Content
Enrichment; Library Online
Services; Linked Data

Introduction

Content-enriched metadata in bibliographic records is considered helpful for library users in identifying and selecting library materials for their needs. Recent user studies published in the library literature have shown that library users today are influenced by Internet search engines, online bookstores, and seamless access to full-text resources, and as a result are more than ever demanding enhanced content and functionality in library catalogues to assist their discovery of relevant search results and resources (Connaway and Dickey 2010; Tosaka and Weng 2011). Content-enriched metadata contains topics related to the subject that enhance the retrievability of relevant items. The presence of this extra-enriched content helps the users to decide an item's relevance to their needs without needing to examine the item physically. Similarly, Calcagno (2000) pointed out the three important benefits libraries will have when they follow content enrichment. According to Calcagno, content enrichment improves the user's ability to locate and evaluate specific titles of interest, enhances the precision of resource sharing, and improves access to underutilised portions of the collection.

CONTACT Vinit Kumar  mailvinitkumar@gmail.com  School of Library and Information Science, Central University of Gujarat, Gandhinagar, Gujarat, India

Color versions of one or more of the figures in the article can be found online at www.tandfonline.com/wjwl.

© 2018 Vinit Kumar

Legacy library online services have relied solely on internal databases to store data in the form of records such as bibliographic data, holdings information, funds, and membership data. The amount of information that can be placed within the record structure is limited to the available information in the internal database of records.

In recent years, there have been huge developments in the presentation and acceptance of data on the Web about interactivity and making data meaningful. Most of the time, this data is generated by either the institution or sometimes by a community of users. This contrasts with the previous approach where only webmasters used to generate data. The ever-growing content and knowledge released online as open datasets or social web APIs not only provides an opportunity but also unlocks possibilities for libraries to complement their data. Conceptually related information, such as places mentioned in the title or the subject of the document, can be semantically linked to disparate and distributed destinations over the web. Eventually, users may benefit with richer data collections and new search possibilities. Thus, enrichment will further establish inherent relationships between media, metadata, and external information sources.

However, there is a need to enrich the metadata available so that extra information can be linked to the bibliographic metadata. For instance, the authors of a book can be linked to their Wikipedia article or their authority file entry stored in another authority file dataset (Nandzik et al. 2013). Also, the extracted entities, such as places, persons, and events, can be disambiguated (e.g., to automatically discern Java as a computing language from Java as an island). Similarly, recent developments like the announcement of Google's Knowledge Graph project involving the display of descriptions from Wikipedia with the search results help expand our understanding of the possibilities of content-enriched metadata (Google 2012, 2016). The launch of Facebook's Open Graph protocol and Smart Internet Research Initiative that enables the semantic description of resources can be of interest to libraries too in providing semantic description of bibliographic items for better discoverability ("Open Graph Protocol" 2016; Ng 2010). We suggest that these kinds of approaches need to be followed in the field of web librarianship to improve the quality of library services.

Approaches to content enrichment

Content enrichment deals with moving content from its current state of programmatic and semantic capability to a higher level through the application of an enhanced structure and additional metadata. The content enrichment of metadata in a bibliographic record is reported as having value to library patrons in identifying and selecting library resources (Tosaka and Weng 2011). The extra-enriched content helps users to determine the relevance of a

specific item for their information needs without the necessity of accessing the full text. There are several methods for accessing content to display such as retrieving related content via a Web Services Application Programming Interface (API), Really Simple Syndication (RSS) feeds, site scraping, and the use of linked open datasets. Each of these approaches will be described to help clarify why we ultimately chose the method we did.

Due to its simplicity, content retrieval via API has gained more popularity, but this approach has some limitations. One of the limitations is the high dependency on a limited number of available APIs. Also, the content available from each API is hidden from search engines, thereby making it difficult to find appropriate API for the requirement at hand. There are some other limitations too, such as mandatory display of branding from the content provider, and the need for continuous efforts on the part of the developer's end to learn each API independently.

Another method for getting content is through RSS feeds, a web feed format for publishing frequently updated information. RSS feeds benefit publishers by letting them syndicate content automatically. As the RSS feed format is XML based, it has gained popularity in aggregating services.

Site scraping is also a method for content extraction from websites. This approach involves the application of computer programs, also known as scrapers, to automatically extract information from various locations on the Internet. Web scraping focuses more on the transformation of unstructured content into structured data. Once the important parts of a website are available, it could be used for content enrichment purposes. There are many data-rich, multiple-record, document-oriented applications using site scraping techniques, such as sites collecting content for advertisements, movie reviews, weather reports, travel information, sports summaries, financial statements, obituaries, and others (Embley et al. 1999). Zheng, Gu, and Li (2012) applied these techniques to detect false drug advertisements on the Web.

In web librarianship, website scraping and HTML heuristics can be effectively used to extract rich metadata from the seemingly unstructured table of contents pages of e-journals. Similarly, Bergmark et al. (2001) scraped the whole ACM digital library to automatically link the references in the documents in the ACM Digital Library. Bergmark and Lagoze (2001) used scraping techniques for "Automatic Reference Linking" of papers in the Cornell Digital Library. Another research-based application for scraping was done by CiteSeer, a popular site for researchers, which used site scraping to develop a Web-based information agent. This agent can find papers that are similar to a given paper using word information and by analyzing common citations made in research papers (Bollacker, Lawrence, and Giles 1998). It also includes a personalised recommendation system that uses browsing behavior and automatic learning to adapt to individual research interests, even as they change over time (Bollacker, Lawrence, and Giles 1998).

Although the website scraping approach has advanced over the years and sophisticated scrapers are available in prominent programming languages, this approach involves a high probability of copyright infringement. Some of the websites restrict the scraping of their content explicitly through their “terms of use” policies, while some do not have explicit “terms of use” policy. In the latter case, it becomes difficult for a developer to decide whether scraping is allowed or not, hence leading to the high probability of copyright infringement.

Observing the above limitations of RSS, APIs, and Web scraping, we propose the use of Linked Data for the enrichment of the metadata of records of library online services. The Linking Open Data project is a project started by W3C based on the idea of the Semantic Web. The term Linked Data or Web of Data refers to a set of best practices for publishing and connecting structured data on the Web (Bizer et al. 2009). The glue that holds together the traditional document Web interface is the hypertext links between HTML pages. Likewise, the glue of the data web is RDF links. An RDF link simply states that one piece of data has a relationship to another piece of data. Once the datasets are published as Linked Data, it becomes easy to find related data linked with another piece of data. Just like on the web, one can get hyperlinked HTML documents from another HTML document.

To consume Linked Data, queries are formatted as per SPARQL protocol to SPARQL endpoints of content providers that in response send the related data in JSON, HTML, and other formats. As this method involves standard procedures for querying content and content representation, high-quality structured content can be retrieved following this approach. As stated by Johnson and Estlund (2014), “by relying on a statement-centric model, using URIs for controlled vocabularies, and welcoming external sources of information, libraries can improve both data quality and collection utility”.

In this paper, we propose a framework for integrating the available online Linked Open Datasets and social applications to enrich the metadata of library online services such as the library WebPAC, digitized archival collections and subject guides. Through rich linkages with complementary data from trusted sources, libraries can increase the value of their data beyond the sum of their sources taken individually. The intricacies of the model are explained with a brief overview of the technology and tools involved. Further, we discuss the development and implementation of a prototype designed for the content-enrichment of a scholarly institutional repository using a Linked Data compliant knowledge base.

SELOS model

The prototype proposed in this paper is based on the SELOS model (Semantic and Social Enrichment of Library Online Services) (Kumar 2014). The model follows a

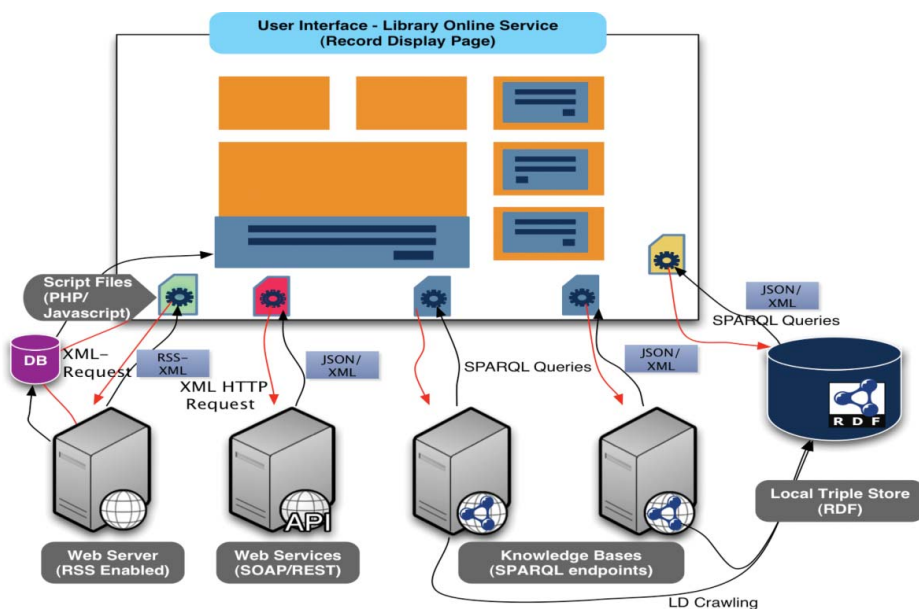


Fig: Semantic and Social Enrichment of Library Online Services (SELOS)

Figure 1. Semantic and Social Enrichment of Library Online Services (SELOS Model).

multilayered approach. The four layers are the Content Layer, the Storage and Request Layer, the Wrapper Layer, and the Presentation Layer (see Figure 1). In the following sections, we discuss the tools and technological aspects applicable in the respective layers.

Content layer

The content layer primarily deals with handling the mechanisms to fetch content from distributed sources with varied architectural builds. This layer plays a major role in the consolidation and compilation of content. The layer comprises content providers, content formats, and technology. The content could be brought in from the following categories of content providers:

- **RSS enabled web servers:** Most of current websites are RSS compliant and make their content available in the form of RSS feed formats, but if the servers are not RSS compliant, they can be easily obtained by using server-side scripting or using XSLT to produce standard RSS feeds.
- **Web Services:** A Web service is a service offered by a Web server to programmatically access data on a web server in a similar manner as is done by humans. Most of the current popular websites such as Twitter or Facebook have opened their datasets for developers in the form of APIs with documentation to access them.

- **Linked Open Datasets:** Databases power nearly every site on the Web and these sites include hyperlinks to other websites, but their databases and the data stored inside them are not linked to each other (Sandhaus 2010). Linked data uses W3C standards for representing the data, known as the Resource Description Format (RDF) and the mechanism for querying the dataset, known as SPARQL Protocol and RDF Query Language (SPARQL).

By following the Linked Data principles, several organisations and services have opened up their data. Apart from opening up the datasets, they also provide a mechanism for handling SPARQL queries, popularly known as SPARQL endpoints.

Storage and request layer

As the content providers discussed in the content layer have different mechanisms to deliver content, the request mechanism for obtaining content must also change accordingly. The storage and request layer deals with the various kinds of requests and storage mechanisms.

The request to the distributed datasets is made with the help of AJAX and CURL requests.

- **AJAX Request:** Asynchronous JavaScript and XML (AJAX) specify a protocol to format requests to any AJAX compliant system. The model used the AJAX method to send encoded SPARQL queries to the distributed datasets. The desired SPARQL Queries are framed following the SPARQL 1.1 Query Language specification (Arenas and Pérez 2011; “SPARQL 1.1 QUERY LANGUAGE” 2017).
- **cURL Requests:** The requests to download RSS feeds from the distributed Web servers is done by using cURL, a utility to download files from the command line. The cURL commands are automatically executed using PHP scripts.

The result obtained in response to the request is stored using storage mechanisms like RDBMS and Triple Store. For quicker access to content, the content collected through RSS feeds, Web APIs, and Site Scraping is stored in a Relational Database Management System (RDBMS). The RDBMS is queried using SQL queries and embedded in the presentation layer with the help of wrappers. Similarly, the content from the Linked Data compatible datasets can be crawled using specific purpose programs known as Linked Data crawlers or LDSpider (Isele et al. 2010). Some datasets also provide their dataset as RDF dumps in the form of downloadable gunzipped files. The downloaded RDF dump files can be indexed and stored in a triple store. Several open source triple store software are available such as Virtuoso. Once the data is ingested in the triple store, it can be queried using SPARQL, a query language for RDF. RDF triples stored in a local

triple-store are readily available, and retrieval of interlinked data becomes possible.

Wrapper layer

This layer comprises server side includes (scripts) to capture relevant metadata and prepare relevant queries to be sent to distributed SPARQL endpoints or local RDBMS or Triple Store. The scripts are written using JavaScript or PHP, or any other scripting language. The server should be configured to support the scripting language in which the wrapper is written.

Presentation layer

The final layer is the user interface layer, which is involved with the presentation of relevant content at the desired location on the webpage of the library online service. This layer provides mechanisms to capture the Uniform Resource Identifier (URI). The presentation layer processes the resulting data from content providers serialised into formats such as JavaScript Object Notation (JSON), RDF/XML and XML and presents this data using boxes, tabs, menus and tables. We propose using JQuery, JavaScript and CSS for presenting the content. JavaScript is a client-side scripting language, and jQuery¹ is a JavaScript library that was selected because using it involves writing less code. It is a multi-browser supported JavaScript library designed to simplify the client-side scripting of HTML. jQuery is also CSS compliant and lightweight. The jQuery UI is a curated set of user interface interactions, effects, widgets, and themes built on top of the jQuery JavaScript Library (jQuery Foundation 2017).

Prototype

In this section, we report on a prototype based on the SELOS model, which was developed to demonstrate the enrichment of records of an Institutional Repository (IR) built using DSpace. The content is brought in from Linked Open Dataset accessed using SPARQL. The content obtained is contextually presented to the user at the record display page of the institutional repository.

Development of a prototype involves several decisions regarding the software stack, customizability, and applicability. We considered our decisions based on two broad criteria: the first was that the prototype should demonstrate the content enrichment of records in an existing service, and second, the prototype should be built on open source tools so that other libraries could adopt the model.

Although other library online services such as Web Public Access Catalogue (WebPAC) and library website were also potential candidates for selection, we chose IR for two major reasons. First, the article level display of an IR needs more

¹<http://jquery.com/>

description, and second, there have already been some efforts from the library sector to enrich WebPACs, but no such efforts have been made for the IR.

Currently, three main software packages are being used in libraries for building institutional repositories. These are, GNU EPrints,² Greenstone Digital Library Software,³ and DSpace.⁴ Among these, DSpace was chosen because of our high level of familiarity with the underlying architecture of DSpace, and because DSpace has the most installations among all the three packages listed above (“DSpace User Registry | DuraSpace” 2017; Kumar 2010).

As the prototype demonstrates the enrichment of content from the open datasets available outside the library, we decided to choose a linked dataset with the broadest coverage, so that it could include all the possible subject topics. For this, we selected DBpedia⁵ as a knowledge base. The DBpedia knowledge base contains a significant amount of general-purpose knowledge and can thus be used to answer queries about a wide range of topics (Bizer et al. 2009). The about page⁶ of DBpedia reads “Today, most knowledge bases cover only specific domains, are created by relatively small groups of knowledge engineers and are very cost intensive to keep up-to-date as domain changes.” Seeing the potential of Wikipedia, it is now referred to as a central hub of knowledge about anything contained in articles written by contributors across the globe. The DBpedia project pulls this enormous source of knowledge by extracting structured information from Wikipedia articles and makes this information accessible on the Web (Auer et al. 2007).

The stats about DBpedia available on the about page reads “The English version of the DBpedia knowledge base describes 4.58 million things, out of which 4.22 million are classified in a consistent ontology, including 1,445,000 persons, 735,000 places (including 478,000 populated places), 411,000 creative works (including 123,000 music albums, 87,000 films and 19,000 video games), 241,000 organizations (including 58,000 companies and 49,000 educational institutions), 251,000 species and 6,000 diseases. The knowledge base consists of 10.3 million pieces of information (RDF triples). It features labels and short abstracts in 111 different languages; 8.0 million links to images and 24.4 million links to external web pages; 27.2 million external links into other RDF datasets, 55.8 million links to Wikipedia categories, and 8.2 million YAGO categories” (Lehmann et al. 2015, 16–18).

The entities are classified in four concept hierarchies: The DBpedia ontology, the YAGO ontology (Suchanek, Kasneci, and Weikum 2007), the UMBEL⁷ ontology and a SKOS (Miles et al. 2005) representation of the Wikipedia category system. In comparison to the existing knowledge bases, the DBpedia knowledge base has several unique characteristics such as varied coverage of domains; community

²www.eprints.org/software/

³www.greenstone.org

⁴www.dspace.org

⁵<http://dbpedia.org>

⁶www.wiki.dbpedia.org

⁷<http://www.umbel.org/>

recognized, automatized mechanisms for updates, such as when there is a change in a Wikipedia article, and it supports multilingual information (Wikipedia Contributors 2015). The DBpedia knowledge base allows queries against Wikipedia, for instance, “Give me all cities in New Jersey with more than 10,000 inhabitants” or “Give me all Italian musicians from the 18th century” (Auer et al. 2007). DBpedia project hosts all its datasets at <https://github.com/dbpedia/extraction-framework/wiki/Datasets2014>

The DBpedia knowledge base provides access through four mechanisms (Auer et al. 2007; Kobilarov et al. 2009; Lehmann et al. 2015):

Through Content Negotiation: DBpedia URIs can be dereferenced over the Web according to the Linked Data principles.

Querying DBpedia resource identifiers (such as <http://dbpedia.org/resource/Bangalore>) returns (a) RDF descriptions when accessed by Semantic Web agents (such as data browsers or crawlers of Semantic Web search engines), and (b) a simple HTML format of the same information for traditional Web browsers. The HTTP content negotiation method is used to deliver the appropriate format.

SPARQL Endpoint: The DBpedia knowledge base can also be accessed through a SPARQL endpoint. Client programs can request queries formatted in SPARQL to the endpoint available at <http://dbpedia.org/sparql>.

RDF Dumps: N-Triple serialisations of the datasets are available for download at the DBpedia website at <http://wiki.dbpedia.org/Downloads38>. These dumps can be downloaded and locally stored using triple-stores and other RDF servers for providing services on the top of it.

Lookup Index: As the consumption of linked data starts with a URI, the URI of the resource for which the query must be made is required to search for URIs for the resources that exist in the knowledge base. To make it easy for the client applications to locate the DBpedia URIs of a DBpedia resource, it provides a lookup service that makes available DBpedia URIs for a text string. This Web service is accessible at <http://lookup.dbpedia.org/api/search.asmx>.

From a sustainability viewpoint, another advantage of using DBpedia is that it dynamically incorporates new concepts as it mirrors new additions in Wikipedia.

How does the prototype work?

The prototype uses data from the DBpedia and displays it on the item display page of DSpace. In terms of the SELOS model described in the Model Section, DBpedia knowledge base provides the support for the Content Layer, whereas DSpace handles the Storage and Request Layer as well as the Presentation Layer. Similarly, JQuery and PHP scripts support the Wrapper layer. The prototype fetches the DBpedia URIs using the “Lookup Index” service and then queries the DBpedia SPARQL endpoint for abstract and related keywords.

The design of the prototype involves the following workflow:

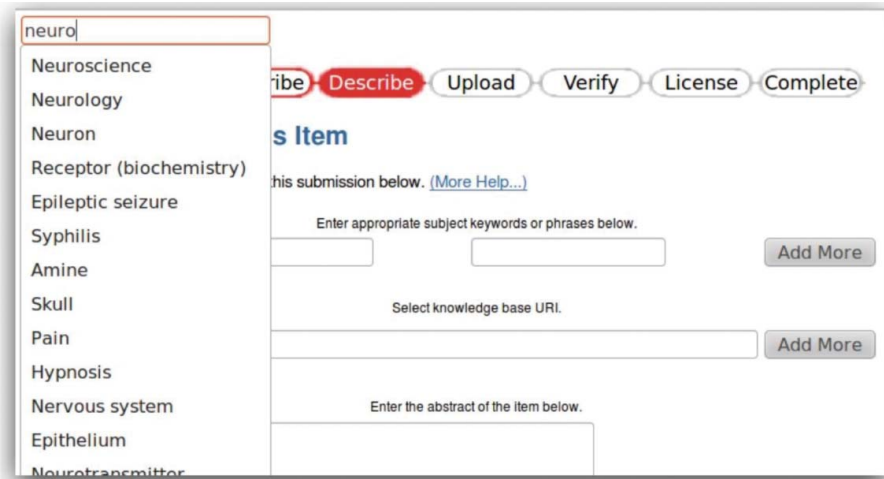


Figure 2. Autosuggestion of DBpedia topics.

- During the submission workflow, the DSpace submitter is presented with an autocomplete field suggesting the topics fetched from the DBpedia as autosuggestions.
- The submitter selects the appropriate topic based on the labels from the autocomplete field, once the topic is selected, the application then queries the DBpedia ID of the selected label and stores the identifier as the value of a metadata element defined in the underlying metadata model of DSpace.
- The identifier stored in the previous step is later used to query the DBpedia SPARQL endpoint for a language-specific abstract and related keywords.
- When a user opens the record display page of the item, the retrieved abstract and related keywords are displayed to the user along with the local metadata.

Configuration and workflow

In this section, we explain the steps taken to configure the prototype. The configuration involved making changes in the files handling the user interface of DSpace. We deployed the prototype on a Linux Machine with the Ubuntu 16.10 operating system.

DSpace provides two user interfaces for Web access, the Java Server Pages-User Interface (JSPUI) written in JSP and eXtensible Markup Language-User Interface (XMLUI). We have implemented the prototype in JSPUI. We preferred JSPUI as it is easier to customise JSPUI because of clear and segregated layout files of this

Table 1. Modification in DSpace metadata format.

Element	Qualifier	Scope Note
subject	dbpedia	'Captures the DBpedia URI'

```
<!-- A new metadata field should be added before making any changes to input-forms -->
<field>
  <dc-schema> dc </dc-schema>
  <dc-element> subject </dc-element>
  <dc-qualifier> dbpedia </dc-qualifier>
  <repeatable> true </repeatable>
  <label> Knowledge Base URI </label>
  <input-type> onebox </input-type>
  <hint> Select knowledge base URI . </hint>
  <required> </required>
  <vocabulary> </vocabulary>
</field>
```

Figure 3. Modified input-forms.xml.

interface as compared to XMLUI, which involves full-fledged theme development for any customisation.

The DSpace application is served through an Apache Tomcat servlet engine configured locally. The JSPUI folder is the main web application folder for a DSpace instance. The layout folder contains files header-default.jsp, footer-default.jsp, location-bar.jsp and other files that make the basic template for the DSpace Web interface. The “static” folder is primarily used to serve static files such as JavaScript files or any other file whose content is not required to be dynamically changed.

The consumption of any knowledge base starts with de-referenceable URIs. For capturing the URI of the concepts, we followed the auto-completion mechanism. The target was to provide the submitter with a textbox, in which the moment the submitter starts entering a topic, the auto-completion starts querying DBpedia and auto suggests related topics as a drop-down list (See Figure 2).

To implement autosuggestion, we used jQuery UI Autocomplete Widget⁸. The jQuery script captures the terms entered by the submitter and prepares an AJAX request for the JSON datatype and forwards the request to a wrapper auto-complete.js written in JavaScript scripting language.

For security reasons like other Web servers, DBpedia also implements the “Same origin policy”⁹ The restriction at DBpedia resulted in an access denied response for

⁸<http://jqueryui.com/autocomplete/>

⁹The same origin policy is an important security concept for a number of browser-side programming languages, such as JavaScript. The policy permits scripts running on pages originating from the same site - a combination of scheme, hostname, and port number - to access each other's methods and properties with no specific restrictions, but prevents access to most methods and properties across pages on different sites. The same origin policy also applies to XMLHttpRequest and to robots.txt (see https://en.wikipedia.org/wiki/Same-origin_policy).

```

PREFIX dcterms: < http://purl.org/dc/terms/>
PREFIX ontology: < http://dbpedia.org/ontology/>
PREFIX foaf: < http://xmlns.com/foaf/0.1/>
SELECT
DISTINCT ?subjectLabel ?Abstract ?primaryTopic ?
broader ?title
WHERE { <" + keyword + "> dcterms:subject ?subject ;
ontology:abstract ?Abstract ;
foaf:isPrimaryTopicOf ?primaryTopic ;
a ?type ;
rdfs:label ?title .
?subject rdfs:label ?subjectLabel .
?type rdfs:label ?broader
FILTER ( langMatches ( lang (? Abstract ), ' EN ') && langMatches (
lang (? broader ), ' EN ') && langMatches ( lang (? title ), ' EN '))}

```

Figure 4. SPARQL query.

the wrapper to fetch results from lookup.dbpedia.org. We resolved the issue of “Same origin policy” by writing another PHP script wrapper `auto.php`.

The DBpedia URI captured in the last step needs to be stored with the metadata of the item. This DBpedia URI will be used to query the abstract and other enrichments at the record display page. To achieve this, we modified the Dublin Core Metadata schema of DSpace using the administrator interface of DSpace. A new field `dc.subject.dbpedia` was created (See [Table 1](#)).

Adding a new metadata field to the metadata schema is not enough to enable the submitter to enter the metadata values. The corresponding modifications in the input form of DSpace are also required. The data entry worksheet of DSpace is pulled from a file `input-forms.xml` available in the config

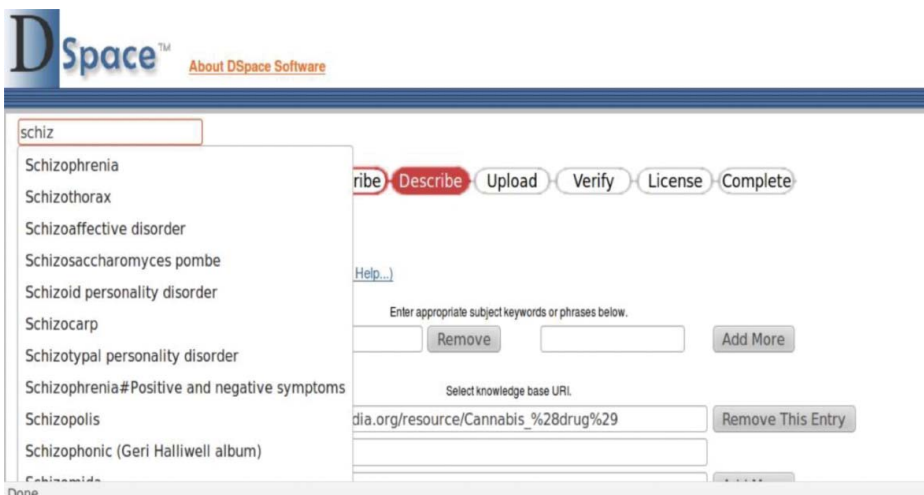


Figure 5. DBpedia topic autosuggestions for prompt ‘schiz’.

Please use this identifier to cite or link to

Title: Assessing the impact of cannabis use on trends in diagnosed schiz

Authors: [Frisher, Martin](#)
[Crome, Ilana](#)
[Martino, Orsolina](#)
[Croft, Peter](#)

Keywords: Cannabis

Issue Date: 2009

Abstract: A recent systematic review concluded that cannabis use increases a model of the association between cannabis use and schizophrenia. This model is based on three factors: a) increased relative risk of psychotic outcomes, b) a substantial rise in UK cannabis use from the mid-1970s onwards in the UK by examining trends in the annual prevalence and incidence analysis of the General Practice Research Database (GPRD) was conducted on almost 600,000 patients each year, representing approximately 2.3% of the population. Schizophrenia and psychoses were either stable or declining. Exploratory conclusion, this study did not find any evidence of increasing schizophrenia reserved.

URI: <http://localhost:8080/xmlui/handle/1/11>

Appears in Collections: [Articles](#)

Files in This Item:

Assessing-the-impact-of-cannabis-use-on-trend-in-diaagn	File

Schizophrenia & Schizoaffective

About Schizophrenia

Schizophrenia is a mental disorder characterized by a breakdown of thought processes and by poor emotional responsiveness, and it is accompanied by significant social or occupational dysfunction. The onset of symptoms typically occurs in late adolescence or early adulthood, and it is often associated with a patient's reported experiences. Genetics, early environment, neurobiology, and psychological and social processes are all factors that influence the disorder, although no single isolated organic cause has been found. The disorder is often confused with other conditions, such as multiple personality disorder or "split personality"—a condition with which it is often confused in public perception. Psychotherapy and vocational and social rehabilitation are also important in treatment. In more serious cases—where the disorder is thought mainly to affect cognition, but it also usually contributes to depression and anxiety disorders; the lifetime occurrence of substance abuse is almost 50%. Social problems, such as homelessness, are common, and the disorder is associated with a 15-year life expectancy that is 15 years less than those without, the result of increased physical health problems and a higher suicide rate (about 10%).

Related Terms : [Greek loanwords](#), [Schizophrenia](#), [Psychosis](#)

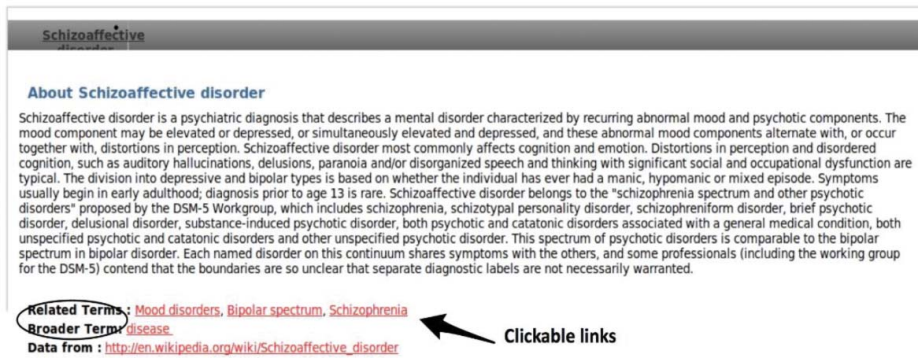
Broader Term: [disease](#)

Data from : <http://en.wikipedia.org/wiki/Schizophrenia>

Figure 6. Enriched interface with external content.

folder in DSpace installation directory. We added the following XML elements and sub-elements according to the documentation guidelines in input-forms.xml (See Figure 3).

The identifier stored in the last step is used to query the DBpedia SPARQL endpoint for the abstract and related keywords. This involved two problems: to find out the value of the DBpedia resource URI from the item display page and to prepare a SPARQL query to be sent to the DBpedia SPARQL endpoint. A JavaScript script was written to find the content of all the elements with the name DC.subject. The results are then saved in a variable.



Schizoaffective

About Schizoaffective disorder

Schizoaffective disorder is a psychiatric diagnosis that describes a mental disorder characterized by recurring abnormal mood and psychotic components. The mood component may be elevated or depressed, or simultaneously elevated and depressed, and these abnormal mood components alternate with, or occur together with, distortions in perception. Schizoaffective disorder most commonly affects cognition and emotion. Distortions in perception and disordered cognition, such as auditory hallucinations, delusions, paranoia and/or disorganized speech and thinking with significant social and occupational dysfunction are typical. The division into depressive and bipolar types is based on whether the individual has ever had a manic, hypomanic or mixed episode. Symptoms usually begin in early adulthood; diagnosis prior to age 13 is rare. Schizoaffective disorder belongs to the "schizophrenia spectrum and other psychotic disorders" proposed by the DSM-5 Workgroup, which includes schizophrenia, schizotypal personality disorder, schizophreniform disorder, brief psychotic disorder, delusional disorder, substance-induced psychotic disorder, both psychotic and catatonic disorders associated with a general medical condition, both unspecified psychotic and catatonic disorders and other unspecified psychotic disorder. This spectrum of psychotic disorders is comparable to the bipolar spectrum in bipolar disorder. Each named disorder on this continuum shares symptoms with the others, and some professionals (including the working group for the DSM-5) contend that the boundaries are so unclear that separate diagnostic labels are not necessarily warranted.

Related Terms: [Mood disorders](#), [Bipolar spectrum](#), [Schizophrenia](#)

Broader Terms: [disease](#)

Data from: http://en.wikipedia.org/wiki/Schizoaffective_disorder

Clickable links

Figure 7. Displaying abstract and related terms.

After saving the URI in a variable, we were able to frame the query. We used the AJAX method to send the query. [Figure 4](#) shows the query sent to DBpedia.

Once the content is received from the DBpedia, the retrieved content from the knowledge base is displayed using CSS and jQuery.

Demonstration

This subsection demonstrates the enrichments made by implementing the prototype in DSpace. For demonstration, we submitted a sample article to the DSpace instance. The title of the article is "Assessing the impact of cannabis use on trends in diagnosed schizophrenia in the United Kingdom from 1996 to 2005," written by Martin Fisher, Ilana Crome, Orsolina Martino, and Peter Croft.

Submission

While submitting, at the Describe step, the autosuggest box (see [Figure 5](#)) popped up and we selected the label Schizophrenia.

Display

At the item display page, the abstract along with the broader terms (keywords) and narrower terms are retrieved from DBpedia and shown in a CSS box (see [Figure 6](#)).

The Related terms and Broader Terms are clickable. Clicking on them causes a local search in the repository (see [Figure 7](#)). This sequence of searching events should suggest to users that they can search with related keywords. This functionality was not available in DSpace before the enrichment of records.

Conclusion

Enriching the content of electronic information is coming of age as it transitions from the traditional bibliographic representation of data to book

previews, book-cover display, and article-level search, which go far beyond capturing as much data as possible from electronic resources to facilitating contextual and semantic information retrieval from library online services. There are two broad approaches for content enrichment that are being followed: on-the-fly enrichment and pre-loading enrichment. The on-the-fly approach involves layering the additional content only as a display feature. Following this method, the enrichment takes place only upon the display and does not affect the search process. In contrast, the pre-loading enrichment approach involves retrieving the enriched content in advance and allowing the local search engine to index the enriched content too. The model followed in this paper proposes an approach to use existing semantic and social web technologies for enriching the content of records of libraries' online services. The content is not only retrieved from the parent institution but also reuses the already available content at distributed locations in the form of knowledge bases.

Knowledge bases have the potential for improving a library's online services, if libraries are ready to embrace them. The SELOS model brings lightweight enrichments to the existing systems; it does not require a full overhaul or major changes in the existing system and services. Further, this ease of implementation increases the chances of adaptability of this model for libraries. The model causes minimal intrusion in the existing services, as there is no need to create many new fields in the existing database. The model has the potential for high adoptability for different systems, such as WebPACs and library websites. Also, there is no need to change the data model of pre-existing library online services. The main purpose of this paper was to demonstrate an approach exposing the potential of semantic knowledge bases for improving the library online services. Based on this model, hopefully libraries can create novel and innovative services.

This paper underlines the importance and need of content enrichment and discusses the various approaches for content enrichment. Further the paper proposes a model, SELOS model architecture for integrating externally available content for the enrichment of library online services. Finally, the paper reports the design and development of a prototype based on the SELOS model for the content enrichment of item record's metadata in DSPace, using the most popular knowledge base of Linked Open Data cloud, the DBpedia.

In the future, we will strive to ingest data retrieved from a variety of knowledge bases to develop a more intuitive display. As most datasets are not currently published as Linked Data, a similar model could be developed for publishing these datasets as Linked Data. Similarly, Natural Language Processing techniques can be used with Linked Open Data for automatic annotations of scholarly content. Automatic interlinking of content is also an area where some further research can be done. We seek to implement the prototype in a

production institutional repository and evaluate the real user impact through statistical methods.

Acknowledgments

The author is thankful to Prof. (Ms.) Devika P. Madalli, Indian Statistical Institute, Bangalore for her guidance and the Indian Statistical Institute, Bangalore for providing a fellowship to complete this work.

ORCID

Vinit Kumar  <http://orcid.org/0000-0001-8306-2087>

About the author

Vinit Kumar is an Assistant Professor at the School of Library and Information Science, Central University of Gujarat, Gandhinagar, Gujarat, India. He is having more than 7 years of experience in teaching and research in Library and Information Science. He has successfully guided three students leading to M.Phil. (LIS) degree. His research interests are library online services, Open data, application of Semantic Web in library services and digital libraries.

References

- Arenas, M., and J. Pérez. 2011. Querying Semantic Web Data with SPARQL. In *Proceedings of the Thirtieth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, 305–16. PODS '11. New York, NY, USA: ACM.
- Auer, S., C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives. 2007. Dbpedia: A Nucleus for a Web of Open Data. In Aberer K. et al. (eds) *The Semantic Web. Lecture Notes in Computer Science*, 4825:722–735. doi:10.1007/978-3-540-76298-0_52.
- Bergmark, D., and C. Lagoze. 2001. An Architecture for Automatic Reference Linking. *IN PROCEEDINGS OF ECDL 2001* 9:115–126.
- Bergmark, D., P. Phempoonpanich, and S. Zhao. 2001. Scraping the ACM Digital Library. *ACM SIGIR Forum* 35 (2):1. doi:10.1145/511144.511146.
- Bizer, C., J. Lehmann, G. Kobilarov, S. Auer, C. Becker, R. Cyganiak, and S. Hellmann. 2009. DBpedia-A Crystallization Point for the Web of Data. *Web Semantics: Science, Services and Agents on the World Wide Web* 7 (3):154–165. doi:10.1016/j.websem.2009.07.002.
- Bollacker, K. D., S. Lawrence, and C. L. Giles. 1998. CiteSeer: An Autonomous Web Agent for Automatic Retrieval and Identification of Interesting Publications. In *Proceedings of the Second International Conference on Autonomous Agents*, 116–123. AGENTS '98. New York, NY, USA: ACM.
- Calcagno, J. 2000. Catalog Enrichment Services, Syndetic Solutions, Inc. In *Proceedings of the Bicentennial Conference on Bibliographic Control for the New Millennium (2001)*. Washington, DC: Library of Congress. http://www.loc.gov/catdir/bibcontrol/calcagno_paper.html.
- Connaway, L. S., and T. J. Dickey. 2010. The Digital Information Seeker: Report of the Findings from Selected OCLC, RIN and JISC User Behaviour Projects. Report. Joint Information Systems Committee (JISC), 27. <http://www.jisc.ac.uk/media/documents/publications/reports/2010/digitalinformationseekerreport.pdf>.
- DSpace User Registry | DuraSpace. 2017. <http://registry.duraspace.org/registry/dspace>.

- Embley, D. W., D. M. Campbell, Y. S. Jiang, S. W. Liddle, D. W. Lonsdale, Y. K. Ng, and R. D. Smith. 1999. Conceptual-Model-Based Data Extraction from Multiple-Record Web Pages. *Data & Knowledge Engineering* 31 (3):227–251. doi:10.1016/S0169-023X(99)00027-0.
- Google. 2012. Rich Snippets (Microdata, Microformats, and RDFa) – Webmaster Tools Help. <http://support.google.com/webmasters/bin/answer.py?hl=en&answer=99170>.
- Google. 2016. How Search Works – The Story – Inside Search – Google. <https://www.google.co.in/insidesearch/howsearchworks/thestory/>.
- Isele, R., J. Umbrich, C. Bizer, and A. Harth. 2010. LDspider: An Open-Source Crawling Framework for the Web of Linked Data. In *Proceedings of the 2010 International Conference on Posters & Demonstrations Track-Volume* 658:29–32. CEUR-WS.org. <http://dl.acm.org/citation.cfm?id=2878407>.
- Johnson, T., and K. Estlund. 2014. Recipes for Enhancing Digital Collections with Linked Data. *The Code4Lib Journal* 23 (January). <http://journal.code4lib.org/articles/9214>.
- jQuery Foundation. 2017. About JQuery UI | JQuery UI. <https://jqueryui.com/about/>.
- Kobilarov, G., C. Bizer, S. Auer, and J. Lehmann. 2009. Dbpedia-a Linked Data Hub and Data Source for Web and Enterprise Applications. *Programme Chairs* 4.
- Kumar, V. 2010. Comparative Evaluation of Open Source Digital Library Packages. In *Open Source Library Solutions OSLS*, edited by A. Tripathi, H.N. Prasad, and R. Mishra, 71–91. New Delhi: Ess Ess Publications.
- Kumar, V.. 2014. *A Model for Enrichment of Library Online Services Using Semantic and Social Web Technologies*, 87–95. Kolkata: University of Calcutta.
- Lehmann, Jens, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, N. Pablo Mendes, Sebastian Hellmann, et al. 2015. DBpedia—a Large-Scale, Multilingual Knowledge Base Extracted from Wikipedia. *Semantic Web* 6 (2):167–195.
- Miles, Alistair, Brian Matthews, Michael Wilson, and Dan Brickley. 2005. SKOS Core: Simple Knowledge Organisation for the Web. In *International Conference on Dublin Core and Metadata Applications*, 3–10. <http://dcpapers.dublincore.org/pubs/article/view/798>.
- Nandzik, Jan, Berenike Litz, Nicolas Flores-Herr, Aenne Löhden, Iuliu Konya, Doris Baum, André Bergholz, et al. 2013. CONTENTUS—technologies for Next Generation Multimedia Libraries. *Multimedia Tools Appl.* 63 (2): 287–329. doi:10.1007/s11042-011-0971-2.
- Ng, Joanna. 2010. The Personal Web: Smart Internet for Me. In *Proceedings of the 2010 Conference of the Center for Advanced Studies on Collaborative Research*, 330–344. CASCON '10. Riverton, NJ, USA: IBM Corp.
- Open Graph Protocol. 2016. The Open Graph Protocol. <http://ogp.me/>.
- Sandhaus, Evan. 2010. Build Your Own NYT Linked Data Application. *The New York Times*. March 30. https://open.blogs.nytimes.com/2010/03/30/build-your-own-nyt-linked-data-application/?_r=0.
- SPARQL 1.1 Query Language. 2017. w3.org. <https://www.w3.org/TR/sparql11-query/>.
- Suchanek, Fabian M., Gjergji Kasneci, and Gerhard Weikum. 2007. Yago: A Core of Semantic Knowledge. In *Proceedings of the 16th International Conference on World Wide Web*, 697–706. ACM. <http://dl.acm.org/citation.cfm?id=1242667>.
- Tosaka, Yuji, and Weng Cathy. 2011. Reexamining Content-Enriched Access: Its Effect on Usage and Discovery. *College & Research Libraries* 72 (5): 412–427. doi:10.5860/crl-137.
- Wikipedia Contributors. 2015. About | DBpedia. <http://wiki.dbpedia.org/about>.
- Xiaoqing, Zheng, Gu Yiling, and Li Yinsheng. 2012. Data Extraction from Web Pages Based on Structural-Semantic Entropy. In *Proceedings of the 21st International Conference on World Wide Web*, 93. WWW '12 Companion. Lyon, France: ACM Press.