Contents

DBpedia Mashups

Mihály Héder and Illés Solt

1 Summary

Abstract If you see Wikipedia as a main place where the knowledge of mankind is concentrated, then DBpedia – which is extracted from Wikipedia – is the best place to find machine representation of that knowledge. DBpedia constitutes a major part of the semantic data on the web. Its sheer size and wide coverage enables you to use it in many kind of mashups: it contains biographical, geographical, bibliographical data; as well as discographies, movie meta-data, technical specifications, and links to social media profiles and much more. Just like Wikipedia, DBpedia is a truly cross-language effort, e.g., it provides descriptions and other information in various languages. In this chapter we introduce its structure, contents, its connections to outside resources. We describe how the structured information in DBpedia is gathered, what you can expect from it and what are its characteristics and limitations. We analyze how other mashups exploit DBpedia and present best practices of its usage. In particular, we describe how Sztakipedia – an intelligent writing aid based on DBpedia – can help Wikipedia contributors to improve the quality and integrity of articles. DBpedia offers a myriad of ways to accessing the information it contains, ranging from SPARQL to bulk download. We compare the pros and cons of these methods. We conclude that DBpedia is an un-avoidable resource for applications dealing with commonly known entities like notable persons, places; and for others looking for a rich hub connecting other semantic resources.

Mihály Héder
Institute for Computer Science and Control (SZTAKI),
Hungarian Academy of Sciences (MTA), e-mail: `mihaly.heder@sztaki.mta.hu`

Illés Solt
Department of Telecommunications and Media Informatics (TMIT),
Budapest University of Technology and Economics (BME), e-mail: `solt@tmit.bme.hu`

2 Introduction

In this section, we take a closer look at Wikipedia itself, then we examine the process by which DBpedia extracts information from it.

2.1 Wikipedia

By now, Wikipedia is a big ubiquitous collaborative encyclopdia counting over 10 million articles in over 200 languages. Readers are very active: Wikipedia receives over 10 billion page views per month and over 200 thousand edits per day. However, growth in article count and number of contributions no longer seems to be exponential for the largest English language edition.[1]

For our purposes, contrasting Wikipedia to traditional printed works is not essential, but it allows us to draw attention to some of its key characteristics. Wikipedia is not governed by a formal editorial board, but instead by the community and its self-imposed guidelines, decision making and escalation processes. Unavoidably, the coverage of articles in a given language edition is biased towards public interest of the Wikipedians speaking the language. The English language Wikipedia has been found to be on par in accuracy with Encyclopædia Britannica [13], and with peer reviewed medical journals [30]. Furthermore, Wikipedia has the unmatched ability to cover current events and incorporate changes in near real time.

Also, Wikipedia is free to download and hack for everyone. As all digital documents, it has structural elements, like lists and tables. Like encyclopædias, it also has a category system. Furthermore, it contains many infoboxes – structured schemas that communicate facts about the subject of the article,for instance of a city. Users can find the infoboxes at the top right part of certain articles. It is not easy to access the infoxbox data programmatically. Many parsing related issues originate from the rather complicated and less standardized nature of wikitext. In spite of these problems the DBpedia project was started in order to extract and structure Wikipedia information.

2.2 DBpedia

As its name suggests, DBpedia[2], aims to provide a structured view of user contributed Wikipedia content [3, 6]. The structuring of the vast amount of data in Wikipedia allows new and innovative uses including querying, navigation, association and aggregation. While the consistency of DBpedia may not keep up with some domain-specific knowledge bases painstakingly crafted by domain experts, how-

[1] Since 2007, see `http://stats.wikimedia.org/EN/#see_also`

[2] `http://dbpedia.org`

ever, its broad coverage and almost real-time updates are key advantages to many applications.

2.2.1 The DBpedia Ontology

DBpedia normalizes information extracted from Wikipedia infoboxes at various levels. First, the infobox type is mapped to an Ontology type, e. g., the Wikipedia article having an "Infobox Austrian district" is classified as dbpedia-owl:Administrative-Region. Ontology types have a fixed set of properties, that are populated based on infobox property mappings, e. g., the "twin1" property of the Austrian district infobox is mapped to the ontology property dbpprop:twinCity which has the range dbpedia-owl:Settlement.

The mapping of infobox templates and properties to ontology types and properties ensures that users of DBpedia data need not be concerned about the peculiarities of the organically evolving Wikipedia infoboxes and can focus on remixing the data instead of creating it.

As you might expect, crafting mapping rules is straightforward in some cases, but overly complex in others. One source of complication is the fact that Wikipedia is not like a database: Infobox properties may be (and often are) filled out in unexpected ways due to the non-triviality of the information entered, but also due to to the insufficient guidance on how infoboxes should (not) be filled out. To cope with this complex and also dynamically changing landscape across the dozens of Wikipedia editions DBpedia has opted for crowdsourcing by the enaction of the DBpedia Mappings Wiki. This meta-wiki allows DBpedia contributors to define and adjust mapping rules for the Wikipedia edition they are most familiar with.

Property values are normalized based on their specified or expected range, e. g., a number entered may be interpreted as a year or a distance in miles/kilometers, depending on the property type and the presence of explicit range specifiers (such as Wikipedia date and conversion templates). Example property ranges contained in DBpedia:

- numeric: integer, float, double
- metric: length, area, volume
- geographical: latitude, longitude, elevation, region
- temporal: date, time, interval

Normalizing plain text property values to the appropriate ranges requires localized parsers to be added to DBpedia's codebase that cope well with incomplete, non-standard, mistyped or even inappropriate user input. For example, expect even correctly spelled dates to be represented as a native string instead of a normalized xsd:date for non-English Wikipedia editions.

Now that we have an overview of the structure of the database we should not forget to ask what is the connection of the data to the real word - what is the semantics of the data? DBpedia is a result of an empirical experiment, and as it often happens with these kind of enterprises, the theoretical framework is running late in explaining its nature and widespread success. As long as we don't want to dig deep, we can say that Wikipedia describes all kinds of things, like persons, groups, locations, events, activities, concepts, etc. These things might be fictional or real, or thought to be real in the past or maybe expected to become real in the future. What we can say about them is that they are mostly backed by a consensus. At this point of course all kinds of exceptions come in mind, like articles about politicians, etc - but that is not the majority, and also the debates are rather about evaluations and normative statement and much rarely about things like the birth date of a certain politician. Second, we can safely assume that the things on Wikipedia are notable enough - there are guidelines to ensure that.[3] Therefore DBpedia data represents machine-readable facts from all the kinds of things mentioned above.

This resonates nicely with the original Semantic Web project, but that project also included a heavy mathematical toolbox to define ontologies to allow machine inference: the Web Ongology Language - abbreviated as OWL - was created.

OWL by design has Description Logic semantics. The older OIL language was designed to implement a description logic called SHIQ and a software called FaCT[4] is used to carry out so-called T-box and A-box reasoning on it. OIL was submitted together with another language called DAML to W3C and there it became OWL[25][5].

Terminology-box (T-box) and Assertion-box (A-box) are terms from description logic[4]. The terminology of a system is defined in a T-Box, and its statements are usually about concepts (sets of objects) and roles (binary relations). A-Boxes are about individuals and contain two kinds of different statements: C(a) and R(a,b). C means "concept assertion", where R is a "role assertion". Examples look like Man(tom) and Parent(tom,jenny).

After this short introduction we can see why some DBpedia developers themselves often characterize their data set aptly as a large A-Box. Although they map infobox properties to OWL properties, the development of the original infoboxes and therefore their terminology (The T-box) is outside the scope of DBpedia. One can think of editors on Wikipedia as the ones who develop both T-boxes (categories, infobox templates and textual descriptions of them) and also A-boxes (actual infobox data about individuals). DBpedia can only retrieve the data because it is machine readable, but not the meaning of text (unlike human readers) in which most of the knowledge relies.

Naturally, this is only one interpretation of DBpedia semantics. There are many others and in general there is a lack of consensus about this question (for a good criticism on Linked Open Data and its usefulness in general see [22]). We should not think that this question is only the businesses of theorists however. To use DBpedia data in any kind of intelligent application an interpretation is needed to be found, one way or the other. What is usually happening is that the developer of a mashup examines what is available at DBpedia; she either already knows or investigates

[3] http://en.wikipedia.org/wiki/Wikipedia:Notability

[4] http://owl.man.ac.uk/factplusplus/

[5] http://www.w3.org/TR/owl-features/

what kind of articles was the data of her interest extracted from; finally she builds an application according to her own somewhat custom interpretation. This means that all the various applications of DBpedia we will discuss later in this chapter should have implied their own idea of the semantics of the data. There is no problem with that - this is just how it works.

2.2.2 DBpedia in numbers

The DBpedia dataset is a collection of information on about 3,77 million things[6], half of which are classified into the unique *DBpedia Ontology*. The distribution of things roughly matches public interest: 500+ k places, 400+ k persons, 180+ k species, 160+ k organizations, 100+ k music albums, 60+ k movies. Altogether 1 billion pieces of information are extracted from the various Wikipedia language editions, though 40 % come from the largest English edition. Names and abstracts are thus available in multiple languages, together with links to images and websites.

2.2.3 DBpedia's connections to other resources

One of the ways DBpedia goes beyond being just a large, isolated database is its rich connections to other projects. Such links explicitly state the equality (owl:sameAs) of a DBpedia entity and a third party concept, allowing creative mashups. DBpedia currently explicitly interlinks with 18 other knowledge bases available as RDF, including:

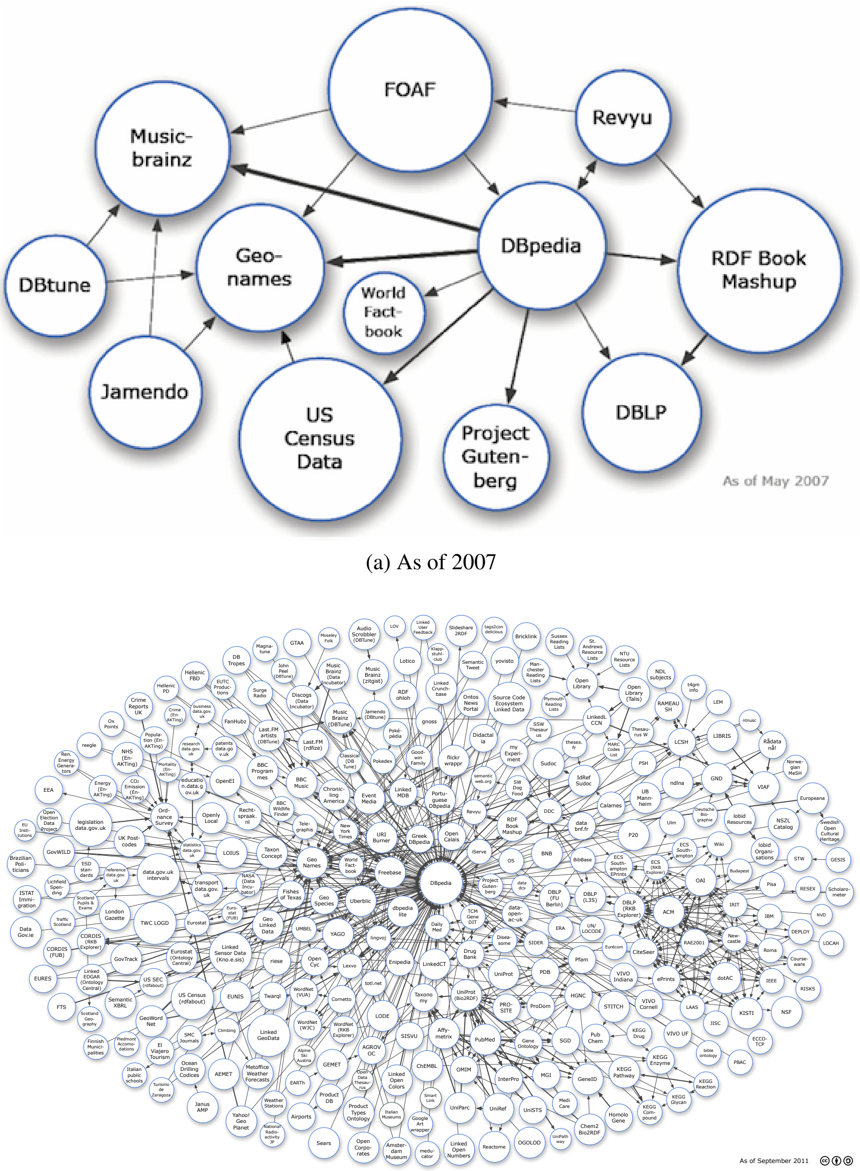
- specialized KBs, e. g. GeoNames, MusicBrainz, Project Gutenberg, Drugbank
- statistical KBs, e. g. US Census, EuroStat, World Factbook
- ontologies, e. g. WordNet, OpenCyc, New York Times

As DBpedia exposes external links found on Wikipedia articles, it may be used to associate things to other repositories, e. g.:

- images via Wikipedia uploads, Wikimedia Commons and flickr wrappr
- videos via YouTube links
- movies via IMDb links
- social media profiles via Facebook and Twitter links

DBpedia has become an integral part and flagship of the Linking Open Data initiative under the umbrella of the Semantic Web. Key to the advancement and adoption of these agendas is to have meaningful and diverse data available to be linked to. In this respect, DBpedia is portrayed as a *nucleus* [3] due to its aforementioned coverage and outlinks to over a dozen other knowledge bases. Furthermore, many other datasets link to DBpedia, effectively making it a hub in the cloud of Open Data (Figure 1).

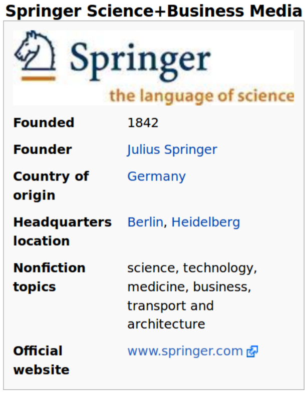
[6] as of August, 2012. For more details, see: http://wiki.dbpedia.org/Datasets



(a) As of 2007

(b) As of 2011

Fig. 1: The Linking Open Data cloud diagram [9] depicting the increase in the number of knowledge bases also available in RDF, and the central role DBpedia plays in their cross-linking.



Springer Science+Business Media

Springer
the language of science

Founded	1842
Founder	Julius Springer
Country of origin	Germany
Headquarters location	Berlin, Heidelberg
Nonfiction topics	science, technology, medicine, business, transport and architecture
Official website	www.springer.com

Property	Value
dbpprop:founded	1842 (xsd:integer)
dbpprop:founder	dbpedia:Julius Springer
dbpprop:country	dbpedia:Germany
dbpprop:headquarters	dbpedia:Berlin dbpedia:Heidelberg
dbpedia-owl:abstract	Springer Science+Business Media S.A., mit Sitz . . . Springer Science+Business Media or Springer is . . .
dbpedia-owl:wikiPageExternalLink	http://www.springer.com
dcterms:subject	category:Pan-European media companies category:Academic publishing category:Commercial digital libraries category:Publishing companies of Germany
foaf:depiction	http://upload.wikimedia.org/. . ./Springer.jpg
is dbpprop:publisher of	dbpedia:Society (journal) dbpedia:European Physical Journal

Fig. 2: Wikipedia infobox and DBpedia data for Springer. http://dbpedia.org/page/Springer_Science%2BBusiness_Media The data on the right-hand side is derived from the infobox, the category links on the page, the abstract of the article and other features.

2.3 Freebase

Freebase[7], a collaborative knowledge base backed by a for-profit organization[8], has taken a different approach to extract and expose structured information from Wikipedia articles [7]. Instead of being a “read-only” repository, it allows and socially encourages its users to edit and extend its contents on a database-editor-like user interface (compare to Wikipedia’s single textbox edit interface with cumbersome wikitext syntax). Thanks to this approach, Freebase can and indeed does grow independently from Wikipedia, housing data on things that would not meet Wikipedia guidelines[9] such as amateur artists, local businesses and offices. In Freebase, also the type system (the analogue of the DBpedia Ontology) is dynamic and can be edited by users, however, only in limited ways to maintain consistency and avoid vandalism.

Freebase regularly crawls Wikipedia for new information, updating and creating entries as necessary while also paying attention to preserve any Freebase user edits that may have taken place between two such cycles. Freebase’s extraction framework[10] (WEX) transforms Wikipedia articles available as wikitext into a well-formed and structured XML dump, opening it up for other projects without the need to deal with peculiarities of wikitext. Both DBpedia and Freebase data are free as in ‘free speech’, they are made available under a Creative Commons Attribution li-

[7] http://www.freebase.com/

[8] Metaweb Technologies, Inc. It has been acquired by Google in 2010.

[9] Especially the notability test: https://en.wikipedia.org/wiki/Wikipedia:Notability

[10] http://wiki.freebase.com/wiki/WEX

cense that allows both derivative works and commercial use as long as the source is acknowledged. DBpedia and Freebase are easily mashed up as both interlink with each other.

2.4 Wikidata

Finally, it is appropriate to shortly mention here a third project called *Wikidata*[11], which aims to create a semantic database that is curated with similar principles as Wikipedia itself. At the time of writing this chapter (mid-2012), the system is under development, so it's too early to evaluate. But the situation can change quickly, so you should check on the project at the time of reading this.

Table 1: Comparison of DBpedia and Freebase

	DBpedia	**Freebase**
Entities	3.77 million	22 million
Data access	read only	read–write
Ontology modification	maintainer only	limited
Download	RDF (N3)	TSV
Query language	SPARQL	Metaweb Query Language
HTTP API	Structured query, keyword and prefix search results in JSON	
Content license	CC-BY-SA (Attribution-ShareAlike)	CC-BY (Attribution)
Source code license	GPL	proprietary
Hosted by	University of Leipzig, Freie Universität Berlin, OpenLink Software	Metaweb Inc. (acquired by Google Inc.)

3 Mashups of the domain

DBpedia interfaces well with other applications due to its broad coverage across topics, languages and geographical regions. DBpedia makes it easy and free to internationalize and localize some applications, features that could be economically unfeasible to license or implement for oneself. Due to the steady growth of Wikipedias across the globe and continued development of DBpedia mapping and extraction frameworks, DBpedia further improves in coverage, consistency and interoperability. In this section, we present a few mashups that use DBpedia, and give an overview of the various ways to access DBpedia data.

[11] `http://www.wikidata.org`

3.1 Mashups that are already using DBpedia

One straightforward way of using DBpedia data is to create custom visualizations. The most spectacular things can be made with the geographical data. One can put a specific subset of the data on a google map or an alternative map technology. An other nice looking and useful thing is to put events or life spans on a time scale. Of course, data can be cumulated by country or filtered by certain conditions before visualization - the options are endless. The following three mashups are nice examples of this kind of application.

3.1.1 Maps and visualization

3.1.1.1 *DBpedia Mobile*

DBpedia Mobile[12] [5] is a location enabled augmented map viewer mashup targeting mobile devices. As you might expect, the user can navigate on a Map enriched with hundreds of thousands of geo-referenced DBpedia entities, including of course points of interest and geographical features. Of course, all these geographical entities are not shown all at once - the whole point of this application is that the user can specify what exactly she wants to see using the semantic features of the data. This way the maps won't become over-populated. The fact that the DBpedia dataset is interlinked with GeoNames, US Census, the CIA factbook, and Eurostat datasets provides a rich user experience. What is more, for certain entities photos can be viewed with the help of flick wrappr[13] or even reviews can be read from Revyu (see later in this section).

Other distinguishing features of DBpedia Mobile are the ability to switch the language in which labels are displayed (independently from the region viewed), and SPARQL integration for selecting entities to be shown. DBpedia data offers some elegant ways to construct the interactive summaries for entities, including text summaries, native name, official website, or hierarchical navigation.

3.1.1.2 *Vispedia*

Vispedia[14] is a visualization interface that is tuned for the visualization of the results using DBpedia. The main idea behind Vispedia is that the best way to consume the semi-structured data of Wikipedia is by interactive data exploration facilitated by a nice interface. In other words, the goal of the system is to bring down the cost of finding and accessing relevant data. [8]

[12] `http://wiki.dbpedia.org/DBpediaMobile`

[13] `http://www4.wiwiss.fu-berlin.de/flickrwrappr/`

[14] `http://graphics.stanford.edu/projects/vispedia/`

“Data” here means every kind of tables on Wikipedia that can be put on a map, timeline or scatterplot. As the tables often don’t contain all the relevant features, semantics from DBpedia is involved. The starting point of a visualization is a Wikipedia article that contains a table. The rows of the table are converted into a graph, in which one node corresponds to each row. At this point, data integration with DBpedia data is carried out interactively by the user, who enters keywords to indicate what kind of information she needs. The keywords are compared to graph edge labels, a similarity measure is calculated and the similar graph edges are included in the result set. Using the similarity measure as path cost, a time-limited A* (A-star) graph search is performed to find relevant entities. By letting the to refine her search keywords, an interactive sensemaking loop can be established.

3.1.2 Search

DBpedia data, along with the many options for visualization also facilitates search. Using semantics in search enables to find entities not only by literal occurrences in text but also by inference - that has been a long-standing goal of the Semantic Web project. The navigation and presentation can also be enhanced by structured data, like in the following two mashups.

3.1.2.1 *Contentus*

Contentus[15] is a Semantic Search Engine with nice user interface that carries out multi-modal search over web documents and linked data like DBpedia. It is capable of marking up web documents with semantic data and present the result to the user. [16] [29].

The main motivation behind the project is that ever-growing open data sets and available content could create novel ways for libraries or other cultural institutes to present their collections on the internet. In this scenario, the existing collections are most of the time already annotatated by librarians or information scientists manually. This metadata has to be intergated with other sources. At the same time, the manual annotation of new digital content is becoming an ever-growing problem because of the increasing pace of collection growth and the lack of human resources. Besides of integrating internet resources (e.g Wikipedia or GeoNames items) with the metadata about a whole document, in the case of digital documents the indexing and linking of entities *within* the content is also possible and could be facilitated. This is what is attempted by Contentus developers. Besides of giving tools to the curators, a nice end user interface is provided. On this interface the users can search in the contents of various multimedia libraries. For instance, a newspaper article might be presented in its original scanned format, complemented by the text extracted with

[15] `http://www.iais.fraunhofer.de/contentus.html`

[16] For a screencast see: http://www.yovisto.com/labs/vissw2011/

OCR. In the document presentation, the semantic entities are highlighted. These entities and their relations are provided by the German National Library's[17] person database[18], but these are mapped to DBpedia thus linked to the Open Data cloud.

3.1.2.2 *SWSE*

SWSE [19] is another Semantic Search Engine that crawles the web for semantic data. Among its results DBpedia entities are among the top ones. [20] SWSE is designed for the users of the general web. It provides search by keywords just like google, yahoo, bing and others, but instead of giving back links to the documents that contain the keywords provided, it returns a ranked list of semantic descriptions about real-world entites. Along the relations of the found entities, users can navigate and discover other entities.

Just like any search engine, SWSE has its own indexes that are built by crawlers and indexers. DBpedia and freebase are only two of the many RDF sources the system relies on. Among the usual crawler and indexer components there are some that are specifically designed for RDF: there is a consolidation component that tries to merge the duplicate entities; also there is a reasoner that generates new rdf based on the existing data.

3.1.3 Recommendations and reviews

The graph that is constituted by the data of the DBpedia makes it possible to for to reasonings about how "close" some things are to others by some measure. This allows for recommending things for the users based on what we already know about their preferences as in the following mashup:

3.1.3.1 *dbrec*

dbrec [20] [27] is a music artist and band recommendation mashup based on DBpedia data. By correlating genres, joint performances and album releases of artists, recommendations are made and also explained to the user. The underlying distance algorithm is implemented via SPARQL queries, and is precomputed over the dataset. User experience is made more attractive by including DBpedia supplied images, descriptions and YouTube videos.

dbrec can provide recommendations for almost 40 thousand artists and bands. These recommendations are based on the so-called Linked Data Semantic Distance

[17] Deutsche Nationalbibliothek - DNB

[18] Personennamendatei - PND

[19] `http://swse.deri.org/`

[20] `http://dbrec.net/`

(LDSD) algorithm. This algoritm is tailored for the characteristics of LOD data: it only relies on links, not on label distance, it does not use a general ontology, only instance data, and it exploits the fact that most of the LOD URIs are dereferencable - meaning that the URI can be fetched by a HTTP GET. The result of the computation with LDSD is a measure normalized to the [0,1] interval between two LOD resources.

This algorithm was applied to the more than 39 thousand artists and bands that could be found (after some data cleaning) on DBpedia at the time of creating the system. This means that basically all the distances between particular artists/bands and all the rest were computed (again, with some optimizations) that took several days. Using the distance database users can simply find similar artists/bands to their favorite ones. Moreover, the database also provides information on what properties shaped the disctance measure. This is called “explanation” and turned out to be a popular feature for the users.

3.1.3.2 *Revyu*

Revyu [21] is a portal where the users can review anything they want. To render a better presentation of the reviews to the users it uses data from DBpedia [15]. At the same time, the site not only consumes but generates RDF as well. The reviews written in the system are processed and e.g. in case of movies, queries against the DBpedia endpoint are executed. In simpler cases this results in the DBpedia resource for the given film. Similar heuristics are applied in case of books, using the RDF book mashup. This is called “retroactive linking” by the developers. They also use “proactive linking”, meaning that they generate “skeleton” (empty) reviews for things users might want to review based on LOD data. The limitation of this approach is that there are too many potential entities to cope with. Besides of linking to DBpedia and the LOD in general, revyu is nicely linkable: the reviews have their own URIs that are dereferenceable.

3.1.4 Plain text enrichment

Recommendations can not only be based on a distance measure within a graph, they can be based on co-occurrence and even on natural language processing (NLP) techniques.

Zemanta Is a blogging assistant that helps bloggers to enhance their content by links to Wiki articles, other blogs, amazon, IMDB entries and such. With the enhancements the users content tends to reach a better place in the search results and to get more links back when linking to other blogs.

BBC Content Link Tool uses DBpedia to help editors in properly tagging any BBC urls with appropriate semantic metadata. [23]

[21] `http://revyu.com/`

Apache Stanbol[22] is an OSGi based Semantic Enhancement Engine. This means that one can send content to the system through an API, and Stanbol responds with enhancements in RDF. The system is integrated with DBpedia Spotlight (see later in this section) and many other annotation sources, e.g. Zemanta or OpenCalais[23].

3.1.5 Identifying

DBpedia is good for identifying things that people put on your website or portal.

For example, *flick wrappr* itself is a mashup, it maps photos to things by correlating flickr tags and geotags with DBpedia labels and geo-references.

Not only places, but also persons can be identified. The *White House Visitor Log*[24] is a demo mashup made at Rensselaer Polythechnic Institute which shows how different sources of data can be mashed up in a single application. On this website users can search for the visitors to the White House – the data is taken from data.gov – and the search results are enriched with DBpedia data, as many of the visitors are prominent politicians with their own Wiki pages [10]. Similarly, *Academia Europea*[25] has many members whose profile page was created with the help of DBpedia and DBLP data. [24]

Finally, you can rely on the entities in DBpedia when populating a new portal with labels and categories. After all, why start with an empty category set or label set when you can have a sensible one right away? *Faviki*[26] is a social bookmarking platform that has chosen to use DBpedia entities (that is Wikipedia articles) as tags. When it comes to tagging, the traditional choices have been using a fixed tag set (taxonomy; e. g., DMOZ Open Directory Project) or allowing any tags to be entered by the user (folksonomy; e. g., flickr). Faviki's approach benefits from both sides: Users select tags from a large but also mostly unambiguous tag space, while the tag space itself is kept current by Wikipedia contributors. Additional benefits that come without extra user or maintainer effort are the support for multiple languages and the structured information associated to tags, notably hierarchical generalization–specialization relations.

This was only a sample of the DBpedia mashups out there that we have found mostly by browsing the many hundreds of citations the initial DBpedia white papers[27]. This means that lots of applications are probably excluded. Also a large number of mashups still existed on paper and in the form of screen shots but the URLs were unaccessible by the time we checked. This indicates how fastly the sce-

[22] http://stanbol.apache.org/

[23] http://www.opencalais.com/

[24] http://logd.tw.rpi.edu/demo/white-house-visit/search

[25] http://www.ae-info.org/

[26] http://faviki.com/

[27] If you want to do your own research, use Google Scholar and search for "Dbpedia: A nucleus for a web of open data" and "DBpedia-A crystallization point for the Web of Data". The two articles together received a remarkable 1300 citations to date

nario changes - we can only hope that you could really find most of the mashups above.

3.2 Accessing DBpedia

DBpedia dataset is accessible in many ways making it easy for both humans and computers to tap into its wealth of information. This section aims to give an overview of the most commonly used methods to access DBpedia dataset.

3.2.1 Download

If you like it raw, DBpedia offers regularly updated bulk downloads of its dataset and ontology at `http://wiki.dbpedia.org/Downloads`.

The DBpedia ontology itself is made in OWL and serialized in RDF/XML for download, while the dataset files are hosted in the less verbose N-Triple and N-Quad format. The latter RDF serialization format is more storage-friendly and also less resource-hungry to process.

3.2.2 SPARQL endpoint

DBpedia also provides an interactive query interface and a RESTful web service at `http://dbpedia.org/sparql` providing a variety of output formats including *de facto* standard JSON. The endpoint interprets the popular RDF query language SPARQL, for a hands-on introduction to SPARQL[28] refer to [1]. Here is an example DBpedia query using only DBpedia and FOAF ontology properties:[29]

```
PREFIX dbo: <http://dbpedia.org/ontology/>
SELECT ?name ?birth ?person WHERE {
     ?person dbo:birthPlace :Berlin .
     ?person dbo:birthDate ?birth .
     ?person foaf:name ?name .
     FILTER (?birth < "1900-01-01"^^xsd:date) .
}
ORDER BY ?name
```

Which will result in an output containing:

```
{"name":  { "type":"literal", "xml:lang":"en",
            "value":"\"Helene\" Ellen Franz"        },
 "birth": { "type":"typed-literal",
            "datatype":"http://www.w3.org/2001/XMLSchema#date",
            "value":"1839-05-30" },
 "person":{ "type":"uri",
            "value":"http://dbpedia.org/resource/Ellen_Franz" }}
```

[28] `http://www.w3.org/TR/rdf-sparql-query/`

[29] http://wiki.dbpedia.org/OnlineAccess

The above query will return the name and date of birth of individuals born in Berlin before 1900. Note that `Berlin` here unambiguously refers to the English Wikipedia article with title ‘Berlin’, thus the German capital; and that the property `name` obtained from the FOAF ontology is in English.

Be aware that processing many and/or complex queries puts a significant burden on the DBpedia’s backend servers, which are operated in a non-profit fashion. In such situations, or when dealing with sensitive information, you might consider setting up DBpedia appliances for private use.

3.2.3 Virtual appliance

Online DBpedia interfaces are driven by the Virtuoso[30] server platform. Virtuoso is also available for free[31] and comes with instructions on how to import the latest DBpedia datasets, thus offering users a way to set up DBpedia endpoints for private use. To make life easier, the developers of Virtuoso offer downloadable appliances preinstalled, preconfigured and populated with DBpedia data for Amazon’s EC2 virtualization platform.[32]

Amazon’s Public Data Set also includes DBpedia, which facilitates integration to other Amazon Web Services applications.

3.2.4 DBpedia Spotlight

We have seen that the coverage of DBpedia offers some unique ways to semantically enrich structured data. However, finding the pieces of data to be enriched in content like free text can be a major challenge.[33] DBpedia Spotlight[34][26] aims to overcome this semantic gap by analyzing plain text and automatically suggesting linkages between DBpedia entities and text spans, very much like Wikipedia internal links. Thus DBpedia Spotlight extends the scope of applications that can benefit from DBpedia, adding e. g., blogs, libraries, feed aggregators, or other applications dealing with user generated text content. Sztakipedia uses DBpedia Spotlight as a source for link recommendation and there is another small application that helps in discoverying Google Summer of Code projects. [35] This application leverages relationships between concepts in DBpedia in order to suggest ”related topics” for students searching for a project. For example, if a student searches for ”Cloud Computing”, the mashup is able to suggest other concepts such as ”Platform as a service” and ”Scalability”.

[30] `http://virtuoso.openlinksw.com/`

[31] More precisely it has a free version, besides the enterprise plan

[32] `http://www.openlinksw.com/dataspace/dav/wiki/Main/VirtAWSPublicDataSets`

[33] See literature on *information extraction*.

[34] `http://dbpedia.org/spotlight`

[35] `http://spotlight.dbpedia.org/gsoc/`

4 Sztakipedia project

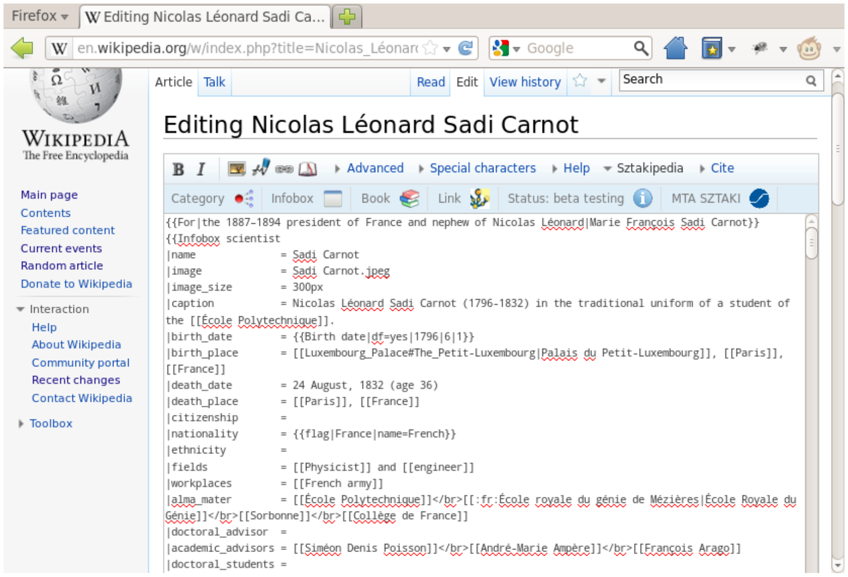
Sztakipedia builds upon structured data from DBpedia and from many other different sources and intends to be an intelligent assistant for Wikipedia editors. We are completely sure that every reader of this sentence has read at least one Wiki article in her/his life. However the huge majority of the readers of Wikipedia have never written a single article. They probably do not suspect what exactly writing an article involves. Let's suppose someone wants to write an article on a more-or-less known historical person. Probably the author is deeply interested or is an expert in the topic, so she has some kind of draft in her head or even text portions ready to copy-paste and revise. She gets a standard html textarea in which she can compile the plain text. The next step is to format the text. Right now it is mainly done by wikitext markup, although a new visual editor is being developed by the Wiki developers, that will most probably become really popular. [36] But our focus right now is on what happens after the formatting: inserting links, infoboxes, categories, and the necessary citations to sources. In our case the author should at least use the `person` infobox, or a more specific one, like `philosopher`. Proper category labels should also be added, as well as source citations, which are required by the Wikipedia editing policies. And, naturally one should link the more important concepts in the text to their corresponding articles. If this is not done, either because the author is new to the system and is not familiar with infoboxes, the category system, etc., or because she does not have enough time to learn the details of the syntax, the newly written article will be reverted, or labeled as "stub" by someone with more administrative power, and for a good reason: source citations, pagelinks, infoboxes, and categories are crucial for the quality of the content. . So they cannot be omitted, but the creation of them could be assisted by a recommender system - this is the idea behind Sztakipedia.

The system design is based on a requirement survey that was conducted among more than 1450 Hungarian wiki editors [16]. Sztakipedia uses DBpedia data, among other sources, for making suggestions of different kinds, offering pertinent content for editors to add to the documents, e.g. Wikipedia infoboxes, categories and page links. Among the information sources used, we highlight Web search, library catalogs, as well as tf-idf[37] database and co-occurrence data extracted from Wikipedia. Through the use of Sztakipedia, Wikipedia users can unknowingly reuse DBpedia data when editing articles, through a toolbar from the standard wiki editor interface (Figure 3). The assisted editing of articles can increase the level of interconnection of existing knowledge and potentially enhance the quality of articles on Wikipedia.

Our broader vision is that a virtuous cycle of semantic enhancement can be created by assisting knowledge creation. The more authors are using a recommender system to create machine-readable annotations in the content, the more training data is created for enhancing the recommenders, this creates a positive feedback loop. An

[36] In the early stage of Sztakipedia project our team also developed a TinyMCE based editor, but that is discontinued now.

[37] tf-idf is a videly used statistical relevance measure. For details, see [28]



```
{{For|the 1887–1894 president of France and nephew of Nicolas Léonard|Marie François Sadi Carnot}}
{{Infobox scientist
|name              = Sadi Carnot
|image             = Sadi Carnot.jpeg
|image_size        = 300px
|caption           = Nicolas Léonard Sadi Carnot (1796-1832) in the traditional uniform of a student of
the [[École Polytechnique]].
|birth_date        = {{Birth date|df=yes|1796|6|1}}
|birth_place       = [[Luxembourg_Palace#The_Petit-Luxembourg|Palais du Petit-Luxembourg]], [[Paris]],
[[France]]
|death_date        = 24 August, 1832 (age 36)
|death_place       = [[Paris]], [[France]]
|citizenship       =
|nationality       = {{flag|France|name=French}}
|ethnicity         =
|fields            = [[Physicist]] and [[engineer]]
|workplaces        = [[French army]]
|alma_mater        = [[École Polytechnique]]</br>[[:fr:École royale du génie de Mézières|École Royale du
Génie]]</br>[[Sorbonne]]</br>[[Collège de France]]
|doctoral_advisor  =
|academic_advisors = [[Siméon Denis Poisson]]</br>[[André-Marie Ampère]]</br>[[François Arago]]
|doctoral_students =
```

Fig. 3: A screenshot from Sztakipedia toolbar

important element of this vision is that the user should make most of the decisions about the suggestions. As a consequence the IS must be present online in the editor interface of the user, e.g. as a plugin.

Leaving the Wiki context, one can find more practical reasons for enriching an article with links and other enhancements. The point of writing a wiki article, a blog post or a forum entry is to convey a message to others. On the web, it all works asynchronous: we find these messages by searching for keywords in a search engine or by clicking on interesting links on one of the very few a web pages we regularly visit, which are usually news portals or social platforms. This is why the writer has to put the new document in context by labeling or categorizing, and enriching it with links and metadata - to make it accessible and *related*. The more related the content is, the more visitors will find it and the more revenue will come from advertisments. If you plan to become a professional blogger who is creating content frequently and effectively, you can already use a tool like Zemanta (which also uses DBpedia data) to streamline the document creation process.

4.1 Features of Sztakipedia

Here we present Sztakipedia-toolbar [38] [17] – which consists of a MediaWiki user script and a modular server in java –, that can be easily enabled by any Wikipedia

[38] http://pedia.sztaki.hu/

user – currently fully functional only for the English and Hungarian Wikipedia. The toolbar provides access to four main functions.

4.1.0.1 *Link recommendation*

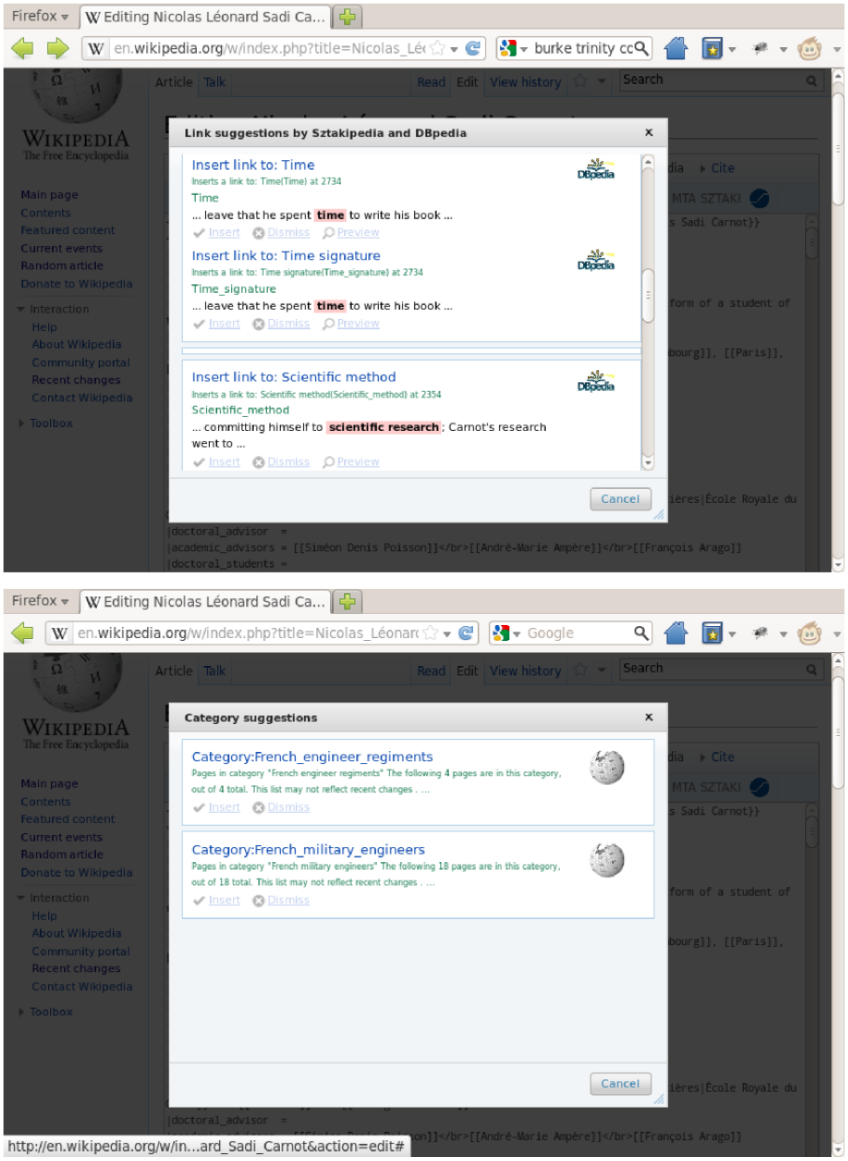
Good links are essential in a document, and so is link recommendation in an intelligent assistant application. In Sztakipedia, this function is partly based on DBpedia's "Links to Wikipedia Article" dataset. This file contains every link that points from one wiki page to an other one. Also, one can count the frequency of linking to the articles. In the English case, given that most of the words or phrases the system encounters are likely to have a wiki page, one can supplement or even replace tf-idf calculation. This is very useful if there is no initial corpus on which a statistical relevance system might be trained. In Sztakipedia we use a phrase weight measure which is based on the product of tf-idf and DBpedia frequency. This measure forms the basis of pagelink recommendations and their ordering. However, this mechanism is complemented with DBpedia Spotlight, DBpedia's own link recommendation feature (see earlier in this chapter). DBpedia Spotlight relies on a number of name-URI associations extracted from titles, redirects and disambiguates, as well as TF*ICF (Inverse Candidate Frequency) scoring [26] of the target text to choose between possible disambiguation options. When the author requests pagelink recommendations, the plain text document derived from wikitext is processed by an UIMA[39] engine, that finds all the words and phrases, which are also page titles, and calculates a weight for them for ranking. Parallel to this, DBpedia Spotlight also processes the text on the DBpedia server. The results are merged and presented as link recommendations to the user. See the top-right corner of the figure 4. for a screenshot.

4.1.0.2 *Infobox recommendation*

The implementation of this function is based on document similarity, calculated by the Lucene[40] framework. The articles in a Wikipedia dump are transformed into plain text and indexed by a Lucene instance. The currently edited document is also converted to plain text and used to search similar articles. If the resulted articles have infoboxes on them - a fact provided by DBpedia -, the hypothesis is that they will be applicable to this document as well. We have tried machine learning techniques to recommend infoboxes and categories but the results were unsatisfactory and we also had to face serious technical problems – concerning mainly memory usage and speed – with a corpus this large. We could not conduct strict numeric measurements on the applicability of the infoboxes recommended by Lucene. However user feedback indicates that in certain topics like settlements and biographies,

[39] UIMA Stands for Unstructured Information Management Architecture. It is a modular framework for annotating content. For more details, see http://uima.apache.org/

[40] http://lucene.apache.org



Link suggestions by Sztakipedia and DBpedia

Insert link to: Time
Inserts a link to: Time(Time) at 2734
Time
... leave that he spent **time** to write his book ...
✓ Insert ⊗ Dismiss Preview

Insert link to: Time signature
Inserts a link to: Time signature(Time_signature) at 2734
Time_signature
... leave that he spent **time** to write his book ...
✓ Insert ⊗ Dismiss Preview

Insert link to: Scientific method
Inserts a link to: Scientific method(Scientific_method) at 2354
Scientific_method
... committing himself to **scientific research**; Carnot's research went to ...
✓ Insert ⊗ Dismiss Preview

Cancel

```
|doctoral_advisor  =
|academic_advisors = [[Siméon Denis Poisson]]</br>[[André-Marie Ampère]]</br>[[François Arago]]
|doctoral_students =
```

Firefox
Editing Nicolas Léonard Sadi Ca...
en.wikipedia.org/w/index.php?title=Nicolas_Léonar
Google
Article
Talk
Read
Edit
View history
Search

Category suggestions

Category:French_engineer_regiments
Pages in category "French engineer regiments" The following 4 pages are in this category, out of 4 total. This list may not reflect recent changes . ...
✓ Insert ⊗ Dismiss

Category:French_military_engineers
Pages in category "French military engineers" The following 18 pages are in this category, out of 18 total. This list may not reflect recent changes . ...
✓ Insert ⊗ Dismiss

Cancel

http://en.wikipedia.org/w/in...ard_Sadi_Carnot&action=edit#

Fig. 4: Link and category Recommendation in Sztakipedia

the Lucene recommendation works quite well that is, the proper infobox is mostly in the top 3-5 recommendations. The recommendation of infrequent infoboxes is less robust in general, but many times it provides infoboxes previously unknown to the users which they usually consider as an added value. Infobox recommendation also has fill support that relies on DBpedia infobox data. For screenshots of the recommendations and the fill helper, see the middle row on figure 5.

4.1.0.3 *Category recommendations*

In general, categorization could be done in a very similar way to infobox recommendation. In the case of a Wikipedia of almost any given language however, this method did not prove precise enough. So to provide category recommendations we use another search engine, Yahoo's Build your Own Search Service (aka. Yahoo BOSS). By searching for the most important phrases we gathered from our weight measure with the following query 'Category <important phrase1>, <important phrase2>. . . ' we usually get good enough category recommendations.

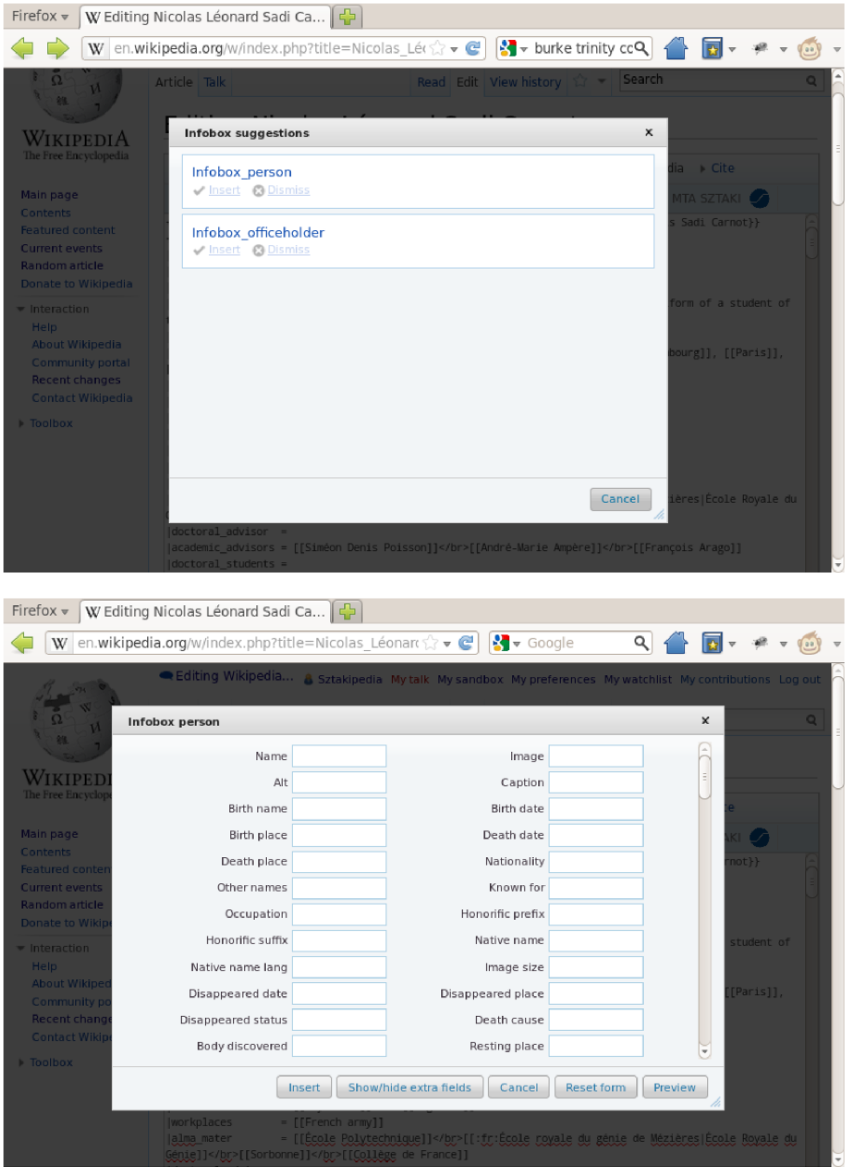
4.1.0.4 *Source citation recommendation*

Finally, there is a fourth kind of recommendation that is based on Linked Data: related literature. This feature is enabled by the fact that both British National Library(BNB) and OpenLibrary(OL) offers their data for download. BNB data is in RDF that one can use directly in an rdf store, while OL data is in JSON, which is also easy to process for a machine. We loaded both in a Lucene instance to make it searchable. Also it is good to know that many libraries offer a Z39.50 or Open Archives Initiative (OAI) interface[41]. The good thing with libraries is that the records they have are usually categorized and have keywords that are created by the enduring labor of generations of librarians. In more specific topics that are covered only a few dozens of books this makes it possible to offer a set of books that contains the one which is just in the article writer's mind.

4.1.0.5 *System architecture*

The architecture of Sztakipedia is depicted on Figure 7. The user interface of the tool communicates with only one server-side endpoint. The mashup of the different sources happens at the server side. The main reason of this is that the system has to collaborate with many different interfaces, some of them not quite accessible from browsers, like the ages old Z39.50 library interface. But an equally important issue is that we need optimized performance for our application.

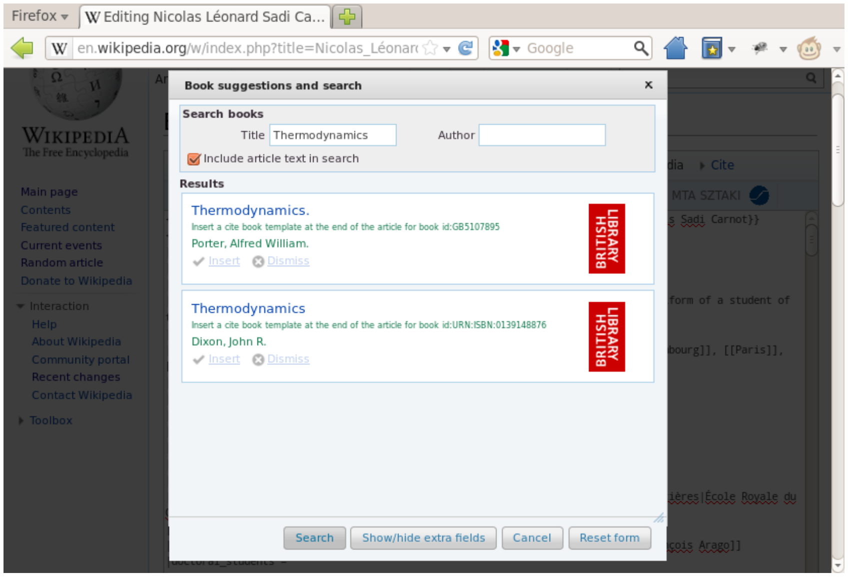
[41] these are the old and new machine interface standards supported by most library systems



```
s Sadi Carnot}}
form of a student of
bourg]], [[Paris]],
ières|École Royale du
|doctoral_advisor  =
|academic_advisors = [[Siméon Denis Poisson]]</br>[[André-Marie Ampère]]</br>[[François Arago]]
|doctoral_students =
```

```
|workplaces        = [[French army]]
|alma_mater        = [[École Polytechnique]]</br>[[:fr:École royale du génie de Mézières|École Royale du
Génie]]</br>[[Sorbonne]]</br>[[Collège de France]]
```

Fig. 5: Infobox recommendation and filling assistant in Sztakipedia.



Book suggestions and search

Search books

Title Thermodynamics Author

☑ Include article text in search

Results

Thermodynamics.
Insert a cite book template at the end of the article for book id:GB5107895
Porter, Alfred William.
✓ Insert ⊗ Dismiss

Thermodynamics
Insert a cite book template at the end of the article for book id:URN:ISBN:0139148876
Dixon, John R.
✓ Insert ⊗ Dismiss

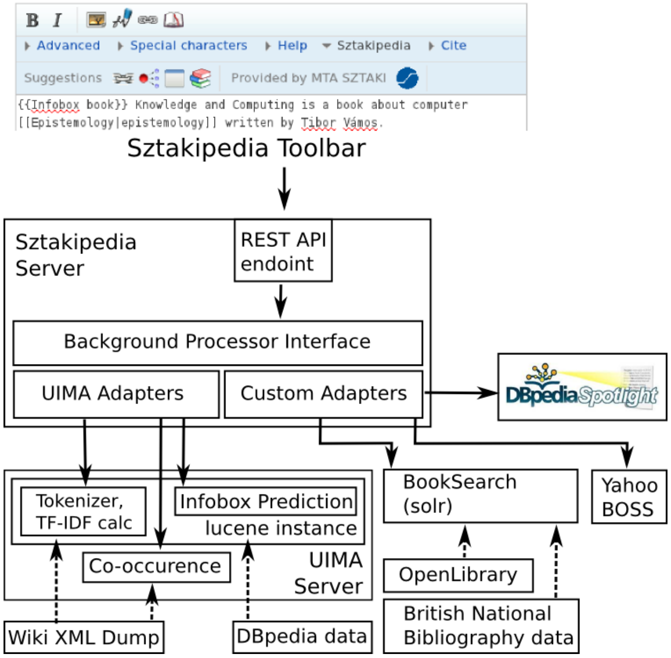
Fig. 6: Book recommendation and search in Sztakipedia.

Behind Sztakipedia there is an Apache UIMA [11] server that can annotate any given text very quickly. Annotation means word importance detection for all words, finding the corresponding DBpedia entities, and also finding similar documents. The data needed for this processing is stored in a Lucene index that can be updated with new documents any time. This system is fed by the plain text version of all wiki articles, plus DBpedia pagelink data.

This also explains why we decided to download some of DBpedia data and not use the API directly. Our application requires the lookup of many thousand words and their linking frequencies preferably under a second. Putting this load to an unoptimized endpoint would overload the system the first time of trying. Here lurks one of the greatest dilemmas of creating mashups.

4.2 *Lessons learned*

It is very convenient when a semantic data source is maintained by others; one does not have to care about the data updates, technical issues with the data sources, etc. On the other hand, the problem of quality of service is not quite solved in the Linked Data paradigm. And, of course, one does not have service quality that is enforceable by a contract when using a free service. On can perceive a tension in the minds of software developers and integrators when it comes to relying on an external service that is available for free: what if it goes down? In these situations it does not really



```
{{Infobox book}} Knowledge and Computing is a book about computer
[[Epistemology|epistemology]] written by Tibor Vámos.
```

Fig. 7: The Architecture of Sztakipedia

matter that e.g. DBpedia has a great uptime. What matters is, that when our team has contracted a customer to provide a service (e.g. Sztakipedia for a Company's internal wiki) under certain quality conditions, we should be able to rely on equally strict or stricter conditions from the services we use to provide ours. Otherwise, we will feel that our back is not covered, and there is a fear in everyone who maintains a system that if the 3rd party service is down, we won't be able to do anything about it. This is why the downloadable DBpedia virtual machine is so important: one can use it locally, but still partly benefit from the service provider.

Furthermore, there are cases like the word importance measure combined from tf-idf and DBpedia link frequency, where the standard interfaces of linked data access just don't fit. In general, it is much more likely that a custom solution will be much faster than a SPARQL query for instance. However, the situation can very easily change with technology and QoS advancements.

Downloading and loading in the data in a customized system is problematic in a different way. In this case a regular nurturing of data is needed that can be very difficult. Consider for example how often Wikipedia changes. DBpedia Live[42] is able to follow these changes in almost real time, while one can only update one's own database when a new dump is created – once in every couple of weeks.

[42] http://live.dbpedia.org/

This sequence introduced all the problems of the “download and use our linked data” approach. Some of the problems were solved when DBpedia Spotlight was introduced, but in general, we still have to do regular data maintenance.

5 Conclusion and future prospects

In this chapter we tried to introduce DBpedia and its richness to the reader. The famous Linked Open Data map rightly puts DBpedia at the center of the picture. We presented many applications and mashups that are using DBpedia, but these are only a small fraction of a large set of projects.

Many different projects explore similar areas like Named Entity Recognition, creating recommendations, semantic search, facilitated editing. However, we are sure that these are only the forerunners of more creative ways of using DBpedia yet unknown. Consider, for example the idea of finding the right and so-often missing column names for tables gathered from the web pages by a search engine [2].

We also presented Sztakipedia, a mashup application in detail, that is – together with DBpedia Spotlight – trying to give something back to Wikipedia editors in return for great value they created by writing articles and thus enabling the DBpedia project.

Acknowledgements We are deeply indebted for Domonkos Tikk and Pablo Mendes for their valuable comments on the text. The authors were partially supported by the grant TÁMOP-4.2.2.B-10/1–2010-0009 of the Hungarian National Development Agency (NFÜ).

References

1. Dean Allemang and James A. Hendler. *Semantic web for the working ontologist: effective modeling in RDF and OWL.* Morgan Kaufmann, 2008.
2. M.S. AMIN and H. JAMIL. An efficient web-based wrapper and annotator for tabular data. *International Journal of Software Engineering and Knowledge Engineering*, 20(2):215, 2010.
3. Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, and Zachary Ives. DBpedia: A Nucleus for a Web of Open Data. In *6th Int’l Semantic Web Conference, Busan, Korea*, pages 11–15. Springer, 2007.
4. F. Baader. *The description logic handbook: theory, implementation, and applications*. Cambridge: Cambridge University Press, 2003.
5. Christian Becker and Christian Bizer. DBpedia Mobile: A Location-Enabled Linked Data Browser. In *Linked Data on the Web (LDOW)*, 2008.
6. Christian Bizer, Jens Lehmann, Georgi Kobilarov, Sören Auer, Christian Becker, Richard Cyganiak, and Sebastian Hellmann. DBpedia – A Crystallization Point for the Web of Data. *Journal of Web Semantics: Science, Services and Agents on the World Wide Web*, pages 154–165, 2009.
7. Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *SIGMOD’08: Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, pages 1247–1250, New York, NY, USA, 2008. ACM.

8. B. Chan, J. Talbot, L. Wu, N. Sakunkoo, M. Cammarano, and P. Hanrahan. Vispedia: on-demand data integration for interactive visualization and exploration. In *Proceedings of the 35th SIGMOD international conference on Management of data*, pages 1139–1142. ACM, 2009.
9. Richard Cyganiak and Anja Jentzsch. Linking Open Data cloud diagram. `http://lod-cloud.net/`.
10. D. DiFranzo, A. Graves, J.S. Erickson, L. Ding, J. Michaelis, T. Lebo, E. Patton, G.T. Williams, X. Li, J.G. Zheng, et al. The web is my back-end: Creating mashups with linked open government data. *Linking Government Data*, pages 205–219, 2011.
11. D. Ferrucci and A. Lally. UIMA: an architectural approach to unstructured information processing in the corporate research environment. *Natural Language Engineering*, 10(3-4):327–348, 2004.
12. David Ferrucci and Adam Lally. Uima: an architectural approach to unstructured information processing in the corporate research environment. *Nat. Lang. Eng.*, 10(3-4):327–348, 2004.
13. Jim Giles. Internet encyclopaedias go head to head. *Nature*, 438(7070):900–901, December 2005.
14. Michael Hausenblas. Exploiting Linked Data to Build Web Applications. *IEEE Internet Computing*, 13:68–73, 2009.
15. T. Heath and E. Motta. Revyu: Linking reviews and ratings into the web of data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 6(4):266–273, 2008.
16. M. Héder. Integrating artificial intelligence solutions into interfaces of online knowledge production. *ICIC Express Letters*, 5(12):4395–4401, 2011.
17. Mihály Héder, Mihály Farkas, Tibor Oláh, and Illés Solt. Sztakipedia: Mashing Up Natural Language Processing, Recommender Systems and Search Engines to Support Wiki Article Editing. In *AI Mashup Challenge 2011 at Extended Semantic Web Conference (ESWC) 2011*, Iraklion, Crete, Greece, 2011.
18. R. Heese, M. Luczak-Rösch, R. Oldakowski, O. Streibel, and A. Paschke. One click annotation. In *Proceedings of the Workshop on Collaborative Construction, Management and Linking of Structured Knowledge (CK2009)*, number 514, 2010.
19. C. Hirsch, J. Hosking, and J. Grundy. Interactive visualization tools for exploring the semantic graph of large knowledge spaces. In *Workshop on Visual Interfaces to the Social and the Semantic Web (VISSW2009)*, volume 443, 2009.
20. A. Hogan, A. Harth, J. Umbrich, S. Kinsella, A. Polleres, and S. Decker. Searching and browsing linked data with swse: the semantic web search engine. *Web Semantics: Science, Services and Agents on the World Wide Web*, 2011.
21. I. Horrocks, U. Sattler, and S. Tobies. Reasoning with individuals for the description logic shiq\ mathcal {SHIQ}. *Automated Deduction-CADE-17*, pages 482–496, 2000.
22. P. Jain, P. Hitzler, P.Z. Yeh, K. Verma, and A.P. Sheth. Linked data is merely more data. *Linked Data Meets Artificial Intelligence*, pages 82–86, 2010.
23. G. Kobilarov, T. Scott, Y. Raimond, S. Oliver, C. Sizemore, M. Smethurst, C. Bizer, and R. Lee. Media meets semantic web–how the bbc uses dbpedia and linked data to make connections. *The Semantic Web: Research and Applications*, pages 723–737, 2009.
24. P. Korica-Pehserl and A. Latif. Meshing semantic web and web2. 0 technologies to construct profiles: Case study of academia europea members. *Networked Digital Technologies*, pages 334–344, 2011.
25. D.L. Mcguinness, R. Fikes, J. Hendler, and L.A. Stein. Daml+oil: an ontology language for the semantic web. *Intelligent Systems, IEEE*, 17(5):72 – 80, sep/oct 2002.
26. Pablo N. Mendes, Max Jakob, Andrés García-Silva, and Christian Bizer. Dbpedia spotlight: Shedding light on the web of documents. In *Proceedings of the 7th International Conference on Semantic Systems (I-Semantics)*, 2011.
27. Alexandre Passant. dbrec: music recommendations using DBpedia. In *Proc. of the 9th International Semantic Web Conference on The semantic web*, volume Part II of *ISWC'10*, pages 209–224, Berlin, Heidelberg, 2010. Springer-Verlag.
28. G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval* 1. *Information processing & management*, 24(5):513–523, 1988.

29. J. Waitelonis, J. Osterhoff, and H. Sack. More than the sum of its parts: Contentus–a semantic multimodal search user interface. In *Proc of workshop on visual interfaces to the social and semantic Web (VISSW), co-located with ACM IUI*, volume 13, 2011.
30. Andrew Wood and Kate Struthers. Pathology education, Wikipedia and the Net generation. *Medical Teacher*, 32(7):618, 2010.

Index