Visual Query Formulation for Linked Open Data: The Norwegian Entity Registry Case

Simen Heggestøyl

simenheg@ifi.uio.no

Guillermo Vega-Gorgojo

guiveg@ifi.uio.no

Martin Giese

martingi@ifi.uio.no

Abstract

The cloud of Linked Open Data is steadily growing, yet it remains largely inaccessible to the general public, due to the technical barrier posed by the requirement to know a formal query language such as SPARQL. In this article, we present a new approach for visual query formulation toward arbitrary SPARQL endpoints, and we show a working prototype developed to improve the accessibility to Linked Open Data sources for non-computer experts. We show how the prototype can be employed to simplify access to governmental Linked Open Data with a case study from the Norwegian Entity Registry.

1 Introduction

Imagine the following situation: a journalist is in the process of writing an article about tiny, but successful, fishing businesses in Norway. For his research, he would like to know the answer to the following question:

Which Norwegian fishing companies have ever had a yearly revenue of more than 100,000,000 (one hundred million) Norwegian kroner, but no more than 10 employees?

To obtain this information, our journalist has some options:

1. He may go ahead and do some research. After a few basic web searches, it seems that the answer to this question is not readily available on the web.

   For instance, a web search at Google with the query string "small fishing companies in Norway" turns up a list of fishing companies in Norway, but information on their size or revenue is lacking. The journalist will have to look up this additional information from other sources, and manually filter out the companies that don't fit the problem description. Unfortunately, this is a very demanding and time consuming task. Furthermore, such manual information gathering is prone to errors and incomplete results, especially if he doesn't find any authoritative information sources. That is, how can he be sure that the list of fishing companies he found through Google is complete?

---

2. In this particular case, an authoritative source for information about companies in Norway is the Norwegian Entity Registry (Norwegian: Enhetsregisteret), as published by The Brønnøysund Register Centre[1] (Norwegian: Brønnøysundregistrene). Using this data source gives the journalist increased confidence in the completeness and correctness of the data, compared to the web search approach.

   But there is one major shortcoming with this approach as well, in that it only supports look-up based on the company *name* or *organization number*, so he cannot formulate his intended query.

3. A third option would be to query a version of the Norwegian Entity Registry that has been published as *Linked Open Data* (LOD)[2]. A detailed explanation of the term Linked Open Data will be given in Section 2. In short, it means that the data is freely available for use and reuse, and published according to the *Linked Data* principles [Ber06], providing a generic and flexible mechanism for accessing, discovering and integrating data from different sources.

   This possibility exists due to the efforts of Semicolon II[3], a research project which was led by Karde AS, with support from a number of large Norwegian actors, including Computas, The Brønnøysund Register Centre, and the University of Oslo.

   The query code the journalist would need to write in order to query the mentioned dataset would look something like this:

```
SELECT DISTINCT ?companyName WHERE {
  ?company a org:Enhet ; rdfs:label ?companyName ;
           org:nacekode inace:A3.1 ;
           org:antAnsattePåDato [ a org:AntAnsattePåDato ;
                                  org:antAnsatte ?employees ] .
  [ a reg:Regnskap ;
    ?enhet ?company ;
    reg:årsresultat ?revenue ]
  FILTER (?employees <= 10 && xsd:integer(?revenue) > 100000000)
}
```

   This approach is good because he will get an answer within seconds. He can also be confident in the data source, as it is an authoritative source for information about Norwegian companies. Furthermore, the answer will be *precise*, as the above snippet is written in a formal query language called SPARQL [HS13]. A formal language has formally defined syntax and semantics. In this case, it also means that running the same code over again would produce the same result, given that the data source hasn't changed. This is what is meant by a *precise* answer.

   But it also means that the journalist has to know the SPARQL language to have any use of the data source. The journalist is not a computer expert, so he realizes he will have to hire somebody to write these codes for him. This process may not only be time consuming, but possibly expensive as well.

Similar scenarios involving other Linked Open Data sources can be imagined:

---

[1]http://brreg.no/
[2]http://data.computas.com/
[3]http://www.semicolon.no/

- A historian exploring Project Gutenberg's[4] vast collection of books.

- A music journalist collecting music facts from DBTune[5].

- A student browsing DBpedia[6], which is a Linked Open Data collection of facts from Wikipedia[7].

These use cases are only a few examples of how people can benefit from access to the cloud of Linked Open Data that has been growing rapidly since 2007 [HB11, Ch. 3.1]. Governments around the world are publishing their public data as Linked Open Data, and the Norwegian Entity Registry is an example of such a governmental data source. Governments are in fact the greatest contributors to the Linked Open Data cloud in terms of sheer data size [BJC11].

How would the situation look if we were able to eliminate the need for the computer expert in these situations? That is, if the "man on the street" was able to exploit Linked Open Data sources, without having to learn a formal query language?

We can see that people would benefit from having a flexible and powerful, yet simple and unassuming method of exploiting Linked Open Data. Appropriate tools and interfaces are needed to fill the gap between the competence of the users and the formal query formulation process, as requested in [Sha+12] and [HDS06]:

> "Extrapolating from the principle that users should not need knowledge of HTML to use the web, we argue that no knowledge of RDF or ontologies should be required to use Semantic Web applications. However this does raise the issue of how to harness the power of the Semantic Web in the interface (e.g. rich information modeling, enabling new kinds of queries) in such a way that the complexity of the underlying implementation is hidden. This is an area that would seem to require considerable further research." [HDS06]

Several natural language interfaces have been developed for this purpose, such as Querix [KBZ06], FREyA [DAC11], and AutoSPARQL [SN12]. They allow users to formulate queries using natural language. However, such tools inherit one of the big problems of natural language: ambiguity [Fre+12]. The natural language interfaces can be made more accurate by elaborating their design, for instance by having users clarify their queries whenever ambiguity arises. Developing accurate natural languages interfaces has however been found to be a very complex and time-consuming task [KB10].

Graphical user interfaces for querying Linked Open Data sources are sparser in the literature. One such interface, Semantic Crystal, was compared with three natural language interfaces in [KB10]. It was, however, found to perform worst of them all, both with respect to query formulation time, result accuracy, and user likability. The study concluded that the graphical user interface was too complex for the target users.

At the group for Logic and Intelligent Data[8] (LogID) at the Department of Informatics, University of Oslo, we wanted to have another try at this problem. We are researching on graphical, interactive interfaces to enable the general public to exploit Linked Open

---

[4] http://wifo5-03.informatik.uni-mannheim.de/gutendata/
[5] http://dbtune.org/
[6] http://dbpedia.org/
[7] https://www.wikipedia.org/
[8] http://www.mn.uio.no/ifi/forskning/grupper/logid/

Data sources, and in particular a Linked Open Data source published by the Norwegian government.

The rest of this article is structured as follows: in Section 2, we define the term *Linked Open Data*. We also present the dataset used for our main case study: a Linked Open Data version of the Norwegian Entity Registry. In Section 3, we present a search tool prototype that we have developed to improve the accessibility to Linked Open Data sources for non-computer experts. We show how the prototype can be used by non-computer experts – such as the journalist from the introduction – to query the Norwegian Entity Registry. Lastly, in Section 4 we summarize our results and discuss the contributions of our work.

2 Linked Open Data

The term *Linked Open Data* (or LOD for short) is used to describe data that can be characterized as both *Linked Data* and *open data*. Following the definition by the Advisory Council of the Open Definition, the definition of "open data" can be summarized as follows:

> "A piece of data or content is open if anyone is free to use, reuse, and redistribute it — subject only, at most, to the requirement to attribute and/or share-alike." [Cou]

Many people feel that data from the public sector should be licensed as open data. Håkon Wium Lie[9] is one such proponent. He argues on the grounds of "fairness"; when we, as taxpayers, collectively fund the production of some data, it is simply fair that this data is given back to us. Wium Lie draws The Norwegian Mapping Authority's nautical charts as an example. The production of the charts is largely funded by tax money. Still, private persons have to pay approximately 10,000 Norwegian kroner to acquire electronic copies of these charts. Wium Lie suggests that the citizens are charged twice – once through tax, and once through direct payment [Lie09; HR13] .

Linked Data denotes a way of publishing data so that it becomes possible to interlink the data with other data sources [HB11, Ch. 2]. This puts the data in a greater context, increasing the value of all the interlinked data. It also enables machines to explore the data and extract meaning through the links – much like we humans explore the World Wide Web through hyperlinks.

The potential value of this can for instance be seen in the case of the BBC; by linking their BBC Music and BBC Programmes services with existing Linked Open Data stores, they were able to create a richer web experience for their users [Kob+09]. Likewise, we want to show that having governmental data – such as the Norwegian Entity Registry – published as Linked Open Data can be beneficial for the people.

The Norwegian Entity Registry

The Norwegian Entity Registry is an example of an governmental dataset that has been published according to the Linked Open Data principles. It contains information about every Norwegian registered "entity". An "entity" can – for instance – be a company or a foundation [Brø]. The registry includes over 350,000 such entities. Every entity is linked to relevant information, such as its organization number, date of founding, number of employees, or daughter entities. Linking together data from the Entity Registry and the

---

[9]Norwegian politician and inventor of Cascading Style Sheets (CSS)

Accounting Registry makes it possible to retrieve historical accounting information for Norwegian entities as well.

As seen in Section 1, accessing this data has so far been inconvenient. The Brønnøysund Register Centre maintains a website[10] which supports per-company look-up, when the name of the company is already known. Some limited subsets of the company registry have also been published as structured data[11]. After being published as Linked Open Data, the data can be explored through a web interface[12], or queried through a SPARQL endpoint[13]. But as previously mentioned, this may still not be very useful to the average citizen.

We will use this dataset as our prime case study while presenting our proposed approach for simplifying exploration and exploitation of Linked Open Data sets for Norwegian citizens.

3   A Query Formulation Tool for Linked Open Data

We have see that there is a large (and growing!) amount of data being published as Linked Open Data, yet it is still inaccessible to the general public due to the lack of suitable user interfaces. Several graphical interfaces already exist for this, but in our view, none of the existing approaches has shown a high enough flexibility and usability to suit general public usage.

The prototype we are developing to solve this is called PepeSearch[14], whose main purpose is to ease the access to Linked Open Data stores for the general public. It is a graphical web application for querying Linked Open Data, geared toward the "man on the street". With PepeSearch, we try to strike a balance between usability and expressivity, while trying to minimize the manual work needed to adapt it to new datasets.

Our approach draws inspiration from faceted search mechanisms, while aiming to be more expressive than existing faceted-based search approaches for Linked Open Data. Faceted search lets the user search datasets by putting restrictions on predefined categories called *facets* [Hea09, Ch. 8.6]. This approach is often applied for product exploration in online shopping sites on the Web, such as Amazon[15] or the Norwegian FINN.no[16] web site. For instance, when searching for books at an online store, relevant facets could be price range, author, book format, text language and whether international shipping is available. By this example, we see that facet restriction can both be number ranges (e.g. price), concept refinements (e.g. book format), or Boolean selections (e.g. international shipping).

Virtuoso Facets[17] is an existing faceted search tool for Linked Open Data, but it works with refinement of *single concepts*, while in Linked Open Data queries we often want to link different concepts together. Unlike pure faceted search tools, our tool allows selection of multiple linked classes, each filtered by their own facets. Other existing tools try to remedy this in different ways; tFacet [BH11] presents related concepts in a hierarchically

---

folded menu, while SemFacet [Are+14] collects facets of related concepts into a list. How useful these tools are to the general public is uncertain, as no studies assessing their usability could be found.
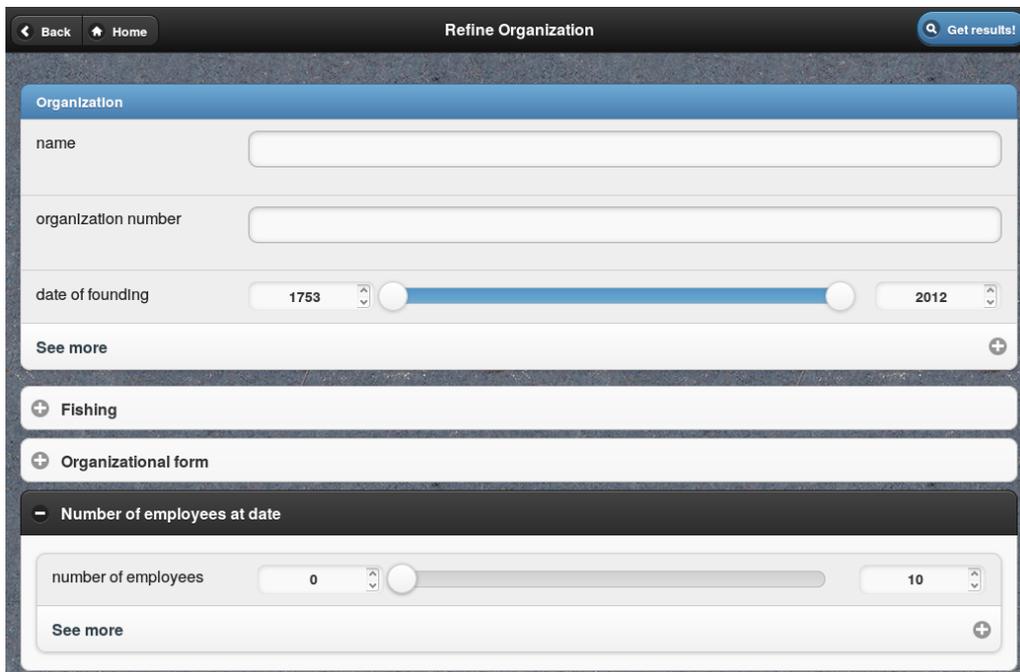
Querying the Norwegian Entity Registry

Figure 1: Using PepeSearch to search for small but successful fishing companies

An example of PepeSearch in use is illustrated in Figure 1, where the tool has been configured to search the Linked Open Data version of the Norwegian Entity Registry. This facet page was generated automatically because the class "Organization" was found in the dataset. The facets "name", "organization number", and "date of founding" are among the datatype properties belonging to the class. Note also that the numerical limits for the "date of founding" facet correspond to the minimum and maximum values found in the dataset. "Sector", "Organizational form", and so on, are classes related to "Organization" by object properties.

If the "Get results!" button is clicked without any refinements to the facets, the user will retrieve a list of every organization found in the dataset (this is not entirely true, since the tool will put a hard limit on the total number of results to retrieve). Optionally, the user may refine the results by putting restrictions on the facets. For instance, if the "date of founding" facet is adjusted, only organizations founded within the given timespan is retrieved.

When the "Get results!" button is pressed, PepeSearch will create a SPARQL query corresponding to the facets and attribute restrictions set by the user. Next, the SPARQL query is sent to a server, where the results are collected, and sent back to PepeSearch. The results are then presented to the user in a table, like the one shown in Figure 2. Note that the concepts are links; they can be clicked for further exploration of their associated properties.

Some limitations has been put on the expressiveness in order to simplify the user interface. For instance, there is no way to express OR in PepeSearch; AND is used by

Organization	organization number	date of founding	Fishing	label	Number of employees at date	Accounting	net income
HANSEN DAHL FISKERI AS	879910692	1998-01-05	Hav- og kystfiske	03.111	5	2008	150891748.00
VARTDAL FISKERISELSKAP AS	920767532	1922-10-20	Hav- og kystfiske	03.111	1	2008	153239748.00
YTTERSTAD FISKERISELSKAP AS	866057222	1993-02-01	Hav- og kystfiske	03.111	6	2009	137764604.00

Figure 2: Small but successful fishing companies in Norway found by PepeSearch

default for every refinement in the search.

A live demo of PepeSearch configured for searching the Norwegian Entity Registry is available at `http://sws.ifi.uio.no/project/semicolon/search/`. A video showing the whole query formulation process of a user (maybe a journalist) using PepeSearch to answer the question posed in the introduction of this article can be viewed at `http://folk.uio.no/simenheg/pepesearch.webm`.
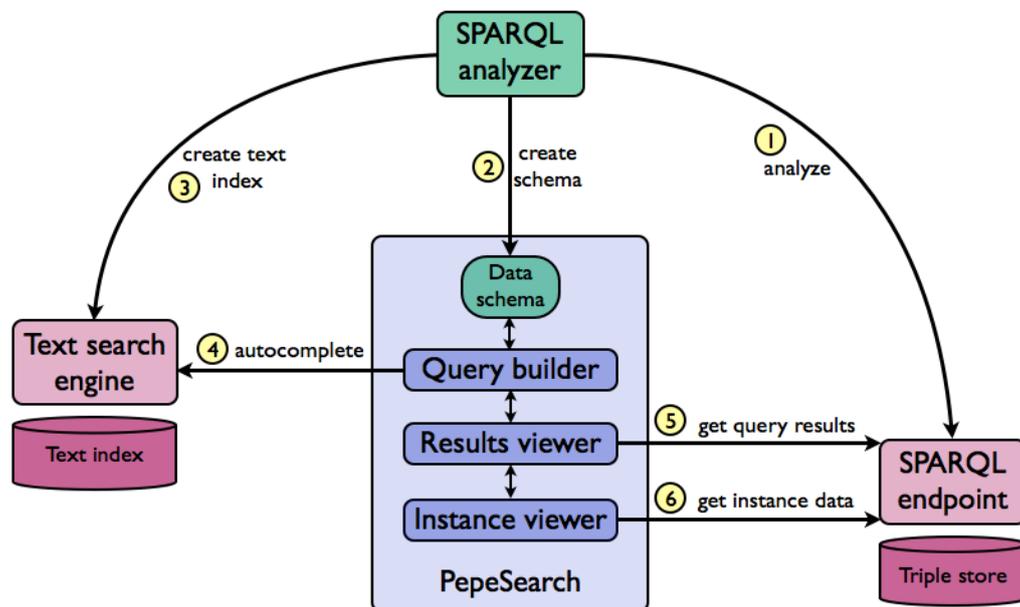
Implementation of PepeSearch

Figure 3: Architecture of PepeSearch

Figure 3 shows an overview of the architecture of PepeSearch. The core component of PepeSearch is programmed in the JavaScript programming language, while the jQuery Mobile[18] framework is employed to make a user interface that is portable across desktop

---

[18] `http://jquerymobile.com/`

computers, smartphones and tablets.

Optionally, PepeSearch can be configured to facilitate completion in text fields with the Apache Solr[19] platform. This enables PepeSearch to suggest search terms dynamically as the user types. For instance, if the user starts to write "nor" in the organization name field, PepeSearch will suggest "NOR-ATLANTIC AS", "GABBRO NOR AS", and "NOR-SAFE DA", among others.

The SPARQL analyzer is a standalone component whose job is to retrieve and analyze data from a given SPARQL endpoint, in order to build a custom user interface for any dataset. This process is fully automated, so a custom user interface can be built without any domain knowledge, but some manual tuning can sometimes be necessary to achieve a nicer presentation. A data graph is mapped to user interface components by the following rules:

Classes are mapped to user interface (UI) classes.

Subclasses of every class are mapped to UI class facets, and organized in a hierarchical class menu.

Object properties between classes are used to link the corresponding UI classes together. We define an *object property* as an RDF property between two RDF resources that both have a type.

Datatype properties of every class are mapped to UI class facets. We define a *datatype property* as an RDF property linking the class to a literal value. Literal types are retrieved as well in order to support different displaying schemes, such as text-fields for strings, or slider bars for numerals. When the literal type is numeric, the minimum and maximum values of that property are recorded to adjust the slider bar accordingly.

Both PepeSearch and the SPARQL analyzer component are released as free and open source software. Their source code can be retrieved from `https://github.com/guiveg/pepesearch` and `https://github.com/simenheg/sparql-endpoint-analyzer` respectively.

Subclass User Interface

Class hierarchies appear naturally in many contexts. In the case of the Norwegian Entity Registry, one such hierarchy is the classification of companies into sectors. These sectors are arranged by the European Union (EU) into a five level deep hierarchy called the NACE hierarchy [Com10]. At each level in the hierarchy, a sector is further refined from its parent sector. This results in broad sectors at the top level, such as "Manufacturing" or "Mining and quarrying", and in highly specialized sectors at the bottom level, such as "Manufacture of bricks, tiles and construction products, in baked clay" or "Quarrying of ornamental and building stone, limestone, gypsum, chalk and slate".

To automatically and correctly detect class membership for SPARQL queries, we employ a technique called *subclass reasoning*. By performing subclasses reasoning, PepeSearch should be able to infer that a company belonging to a sector should also belong to every superclass of that sector. PepeSearch knows when to perform subclass reasoning based on previous configuration by the SPARQL analyzer.
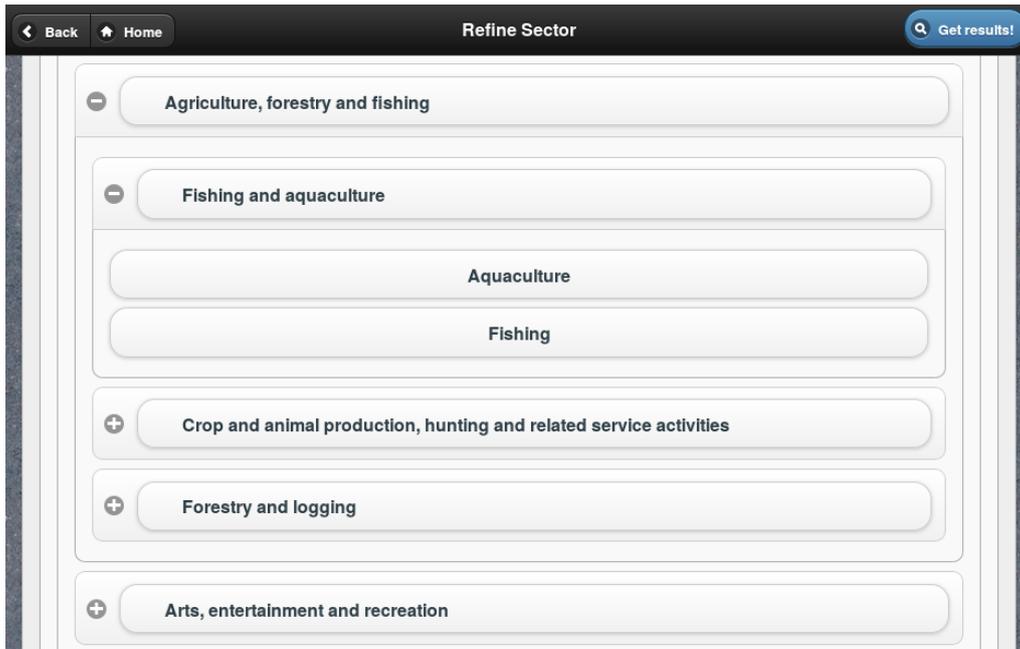
---

[19]`https://lucene.apache.org/solr/`

Figure 4: Browsing subclasses of the "Agriculture, forestry and fishing" sector

To support class selection, PepeSearch presents class hierarchies found in the dataset in a foldable menu. Figure 4 shows the interface in use, where a user is refining the sector class. Upon expanding the "Fishing and aquaculture" sector, the subclasses "Aquaculture" and "Fishing" are displayed and selectable. So for instance, if the class "Fishing and aquaculture" is selected, PepeSearch will perform subclass reasoning to infer that the search should include instances of the classes "Aquaculture" and "Fishing" as well.

It is also possible to search the subclass hierarchy through a free-text search bar. When the user enters a search string there, the interface will show a flat list of every class with a name matching the given string.

Note that this interface supports only single-class selection. While multi-class selection may be a desirable feature, we had to consider the trade-off of increasing the user interface complexity.

Adapting to Other Linked Open Data Stores

While our main case study has been support query formulation for the Norwegian Entity Registry dataset, we intend PepeSearch to be adaptable to arbitrary Linked Open Data stores. To put our tool to the test, we successfully employed PepeSearch to additional SPARQL endpoints, including the Semantic Web Dog Food Corpus[20], the Nobel Prizes dataset[21], and a Linked Open Data version of the FactPages[22] from the Norwegian Petroleum Directorate[23]. There is also ongoing research on using PepeSearch for medical guidelines access.

Figure 5 shows PepeSearch configured to search the FactPages from the Norwegian Petroleum Directorate, which contain public information regarding petroleum activities in Norway. Among the concepts recorded here are oil fields, involved companies, wells,

---

[20] http://data.semanticweb.org/
[21] http://data.nobelprize.org/
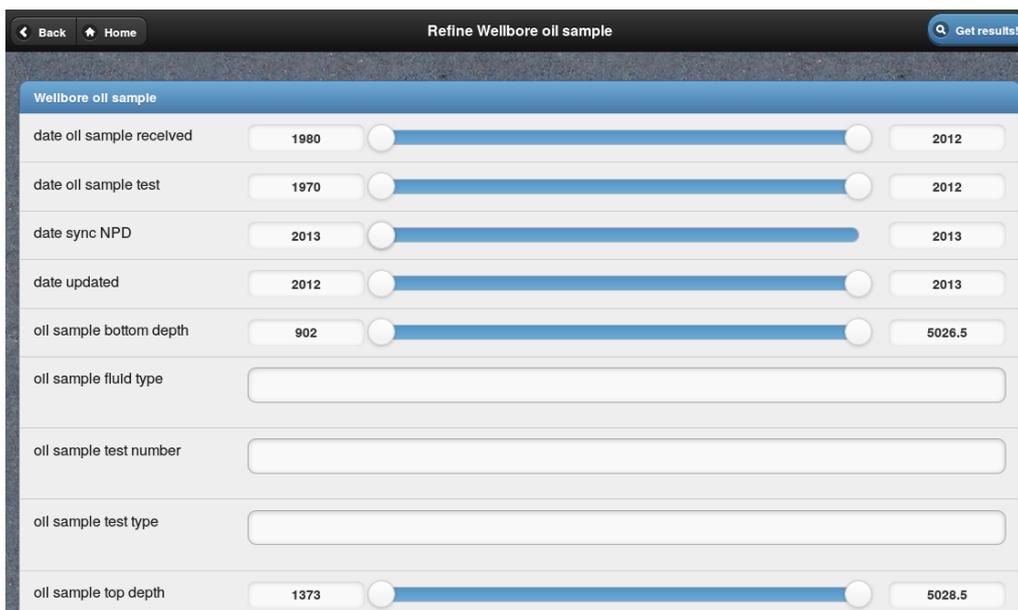[22] http://factpages.npd.no/
[23] http://www.npd.no/

Figure 5: Using PepeSearch to search for wellbore oil samples

and wellbores. Configuring PepeSearch for the FactPages with the SPARQL endpoint analyzer was successful, but with over 170 classes, over 50 possible links between some of them, and equally many datatype properties, the interface approaches the brink of what it can handle. For the heaviest classes, the interface spends several seconds to load.

PepeSearch in Practice

In order to assess the usability and utility of PepeSearch, we carried out an evaluation experiment with 15 Norwegian citizens. The participants were asked to use PepeSearch to attempt to answer six different questions related to Norwegian companies, similar to the question posed in the introduction of this article, crafted by us in attempt to cover as broad a range of use cases as possible. The participants received no prior training with the tool, except a visual tutorial explaining the different components of the user interface. We measured the *retrieval performance* of a user for one task as the *F-measure* of the submitted answers relative to the correct result set defined by us.

We found that the users were able to achieve a high retrieval performance using PepeSearch to answer the questions. Moreover, the confidence of the users and the perceived usefulness of PepeSearch was highly rated by the participants in the study. For details about the experimental setup and the results from this study, see [Veg+14].

4 Conclusions

Let us come back to the introductory example of the ambitious journalist who wanted to answer the following question:

Which Norwegian fishing companies have ever had a yearly revenue of more than 100,000,000 (one hundred million) Norwegian kroner, but no more than 10 employees?

Recall that to answer this question, the only real option the journalist had was to have a computer expert query the Norwegian Entity Registry by means of SPARQL, a formal query language.

With PepeSearch, the situation has changed. By using the graphical user interface of

PepeSearch to formulate his query, we believe that the journalist would be able to answer the question on his own. The results from our user experiment reinforces our confidence in the usability of PepeSearch for non-computer experts to visually query Linked Open Data stores.

Limitations and Future Work

Our current tools only work with a single data endpoint. To utilize the full potential of Linked Open Data, we would like to research the possibilities of performing federated queries across multiple linked datasets.

During the user experiment, we received several useful suggestions with respect to improving the usability of PepeSearch. Specific suggestions include having a more useful initial form, and improving the usefulness of the slider bars.

Fifteen people participated in our user study, all of which were associated with the fields of either informatics or library science. PepeSearch turned out to perform well with this particular group of people, but even if this pilot experiment gave indications that PepeSearch works well for querying Linked Open Data, more comprehensive user studies are needed to investigate how well it really works for the "man on the street".

Acknowledgements

References

[Are+14]   Marcelo Arenas et al. "SemFacet: Semantic Faceted Search over Yago". In: *Proceedings of the Companion Publication of the 23rd International Conference on World Wide Web Companion*. Seoul, Korea: International World Wide Web Conferences Steering Committee, 2014, pp. 123–126.

[Ber06]   Tim Berners-Lee. *Linked Data*. July 27, 2006. URL: `http://www.w3.org/DesignIssues/LinkedData.html` (visited on 07/14/2014).

[BH11]   Sören Brunk and Philipp Heim. "tFacet: Hierarchical Faceted Exploration of Semantic Data Using Well-Known Interaction Concepts". In: *Proceedings of the International Workshop on Data-Centric Interactions on the Web (DCI 2011)*. Vol. 817. 2011, pp. 31–36.

[BJC11]   Chris Bizer, Anja Jentzsch, and Richard Cyganiak. *State of the LOD Cloud*. Sept. 19, 2011. URL: `http://lod-cloud.net/state/` (visited on 07/14/2014).

[Brø]   Brønnøysundregistrene. *Enhetsregisteret*. URL: `http://www.brreg.no/registrene/enhet/` (visited on 07/14/2014).

[Com10]   The European Commission. *List of NACE codes*. Mar. 25, 2010. URL: `http://ec.europa.eu/competition/mergers/cases/index/nace_all.html` (visited on 07/14/2014).

[Cou]   The Open Definition Advisory Council. *Open Definition*. URL: `http://opendefinition.org/` (visited on 07/14/2014).

[DAC11]   Danica Damljanovic, Milan Agatonovic, and Hamish Cunningham. "FREyA: an Interactive Way of Querying Linked Data using Natural Language". In: *8th International Semantic Web Conference*. (Westfields Conference Center, near Washington, DC). 2011, pp. 125–138.

[Fre+12]   André Freitas et al. "Querying Heterogeneous Datasets on the Linked Data Web: Challenges, Approaches, and Trends". In: *IEEE Internet Computing* 16.1 (2012), pp. 24–33.

[HB11]   Tom Heath and Christian Bizer. *Linked Data: Evolving the Web into a Global Data Space*. 1st ed. Morgan & Claypool, 2011.

[HDS06]   T. Heath, J. Domingue, and P. Shabajee. "User Interaction and Uptake Challenges to Successfully Deploying Semantic Web Technologies". In: *Third International Semantic Web User Interaction Workshop (SWUI 2006), Athens, GA, USA*. 2006.

[Hea09]   Marti A. Hearst. *Search User Interfaces*. Cambridge University Press, 2009.

[HR13]   Simen Heggestøyl and Daniel Rødskog. "Openness of Norwegian Public Data". In: *Open Source, Open Collaboration and Innovation*. Ed. by Wolfgang Leister and Nils Christophersen. Norsk Regnesentral, Aug. 22, 2013. Chap. 7, pp. 135–146.

[HS13]   Steve Harris and Andy Seaborne, eds. *SPARQL 1.1 Query Language*. Mar. 21, 2013. URL: http://www.w3.org/TR/2013/REC-sparql11-query-20130321/ (visited on 07/14/2014).

[KB10]   Esther Kaufmann and Abraham Bernstein. "Evaluating the usability of natural language query languages and interfaces to Semantic Web knowledge bases". In: *J. Web Sem.* 8.4 (2010), pp. 377–393.

[KBZ06]   Esther Kaufmann, Abraham Bernstein, and Renato Zumstein. "Querix: A Natural Language Interface to Query Ontologies Based on Clarification Dialogs". In: *5th International Semantic Web Conference*. (Athens, GA, USA). Springer, 2006, pp. 980–981.

[Kob+09]   Georgi Kobilarov et al. "Media Meets Semantic Web - How the BBC Uses DBpedia and Linked Data to Make Connections". In: *ESWC*. Ed. by Lora Aroyo et al. Vol. 5554. Lecture Notes in Computer Science. Springer, 2009, pp. 723–737.

[Lie09]   Håkon Wium Lie. *Offentlig informasjon skal være offentlig*. May 5, 2009. URL: http://voxpublica.no/2009/05/offentlig-informasjon-skal-v%C3%A6re-offentlig/ (visited on 07/14/2014).

[Sha+12]   Nigel Shadbolt et al. "Linked Open Government Data: Lessons from Data.gov.uk". In: *Intelligent Systems, IEEE* (2012), pp. 16–24.

[SN12]   Nurfadhlina Mohd Sharef and Shahrul Azman Mohd Noah. "Semantic Search Processing in Natural Language Interface". In: *Computing and Convergence Technology (ICCCT), 2012 7th International Conference on* (2012), pp. 1436–1442.

[Veg+14]   Guillermo Vega-Gorgojo et al. "Visual query interfaces for semantic datasets: an evaluation study". In: *Journal of Web Semantics* (2014). Submitted for publication.