Social Semantic Search: A Case Study on Web 2.0 for Science

Laurens De Vocht[1], Selver Softic[2],
Ruben Verborgh[1], Erik Mannens[1], and Martin Ebner[2]

[1] imec - IDLab, Ghent University,
Technologiepark-Zwijnaarde 15 9052 Ghent, Belgium.
{firstname.lastname}@ugent.be
[2] Institute for Information Systems and Computer Media, Graz University of Technology,
Inffeldgasse 16c 8010 Graz, Austria.
{firstname.lastname}@tugraz.at

Abstract. When researchers formulate search queries to find relevant content on the Web, those queries typically consist of keywords that can only be matched in the content or its metadata. The Web of Data extends this functionality by bringing structure and giving well-defined meaning to the content and it enables humans and machines to work together using controlled vocabularies. Due the high degree of mismatches between the structure of the content and the vocabularies in different sources, searching over multiple heterogeneous repositories of structured data is considered challenging. Therefore, we present a semantic search engine for researchers facilitating search in research related Linked Data. To facilitate high-precision interactive search, we annotated and interlinked structured research data with ontologies from various repositories in an effective semantic model. Furthermore, our system is adaptive as researchers can synchronize using new social media accounts and efficiently explore new datasets.

Keywords: Web of Data, Linked Data, Social Media, Semantic Search, Digital Libraries, Web 2.0, Research 2.0

1 Introduction

The evolution of Web 2.0 enabled many users via wikis, blogs and other content publishing platforms to become the main content providers on the web. The Web 2.0 for Science, also known as *Science 2.0* or *Research 2.0* aims to adapt the Web 2.0 for researchers. It entails a set of tools and services which researchers use to discover resources, such as academic publications or events they might be interested in, as an alternative to traditional search engines (De Vocht et al., 2011). The tools and services are typically API's, publishing feeds, search and discovery services and interfaces designed based on social profiles (Parra & Duval, 2010; Ullmann et al., 2010). Research 2.0 comprises interacting with information published on Social Media, online collaboration platforms and other Web 2.0 tools. These platforms find more and more uptake (Van Noorden, 2014). The data is available under the form of posts, threads, tags and user information is transferable into semantic form, since widely used and accepted vocabularies for these domains exist. Weaving microblogs into the Web of Data is interesting from a researcher centric semantic search perspective. Twitter[1], as exemplary microblog Social Media platform, can help resolving scientific citations (Weller et al., 2011).

Studies on the use of microblogs like Twitter during conferences within the science community showed that researchers were using Twitter to discuss and asynchronously communicated on topics during conferences (Ebner et al., 2010) and in their everyday work (Reinhardt et al., 2009). A survey on Twitter use for scientific purposes (Letierce et al., 2010) showed that Twitter is not only a communication medium but also reliable source of data for scientific analysis, profiling tasks and trends detection (Tao et al., 2011; Mathioudakis & Koudas, 2010; Softic, Ebner et al., 2010). Twitter hashtags have an influence on the structuring of communication within Twitter as well as for community building (Laniado & Mika, 2010; Bakshy et al., 2011).

However, the mass produced data remains in so-called 'data silos' bound to a specific platform or somewhere within databases. The access to these data sources is associated with specialized application interfaces (API's) which requires specialized technical knowledge to retrieve the data in a desirable form. Many information public interest sources remain captured behind a so-called 'walled garden'. Combining information resources over the walls leads to a high degree of mismatches between vocabulary and data structure of the different sources (Herzig & Tran, 2012). When users formulate a (Web) search in a certain context across multiple data sources, it often includes keywords. In many cases the semantic importance and meaning of the keyword is not considered. The keyword order and combination in a query affects the context, the precise goal of the search and thus the results.

Mostly direct querying approaches were tried and applications were often built around a limited set of supported query patterns. Furthermore, queries are still hard to construct for end users or even developers, despite GUIs and advanced query builders. Vocabularies are getting more streamlined and linked data is maturing. This leads to much more possibilities compared to traditional keyword search. Therefore, we propose a semantic model that drives the search engine, and is optimized for this use case. The key variables that are important in this regard are the *efficiency* (performance and complexity) and the *effectiveness* (search precision) of the proposed engine and thus indirectly the model it implements.

---

[1] http://www.twitter.com/

1.1 Research Questions

Social semantic search combines concepts aimed at personalized information retrieval with well-defined services resolving case specific Web user needs. Understanding semantic search in the scope of information retrieval (IR) differs from the one in the Semantic Web community (Tran, Herzig & Ladwig, 2011). However, common to many semantic search approaches is using a ‘Semantic Model’ which includes (heterogeneous) data sources, a query mechanism and a matching framework. We investigate how researchers find the results by implementing an engine that enables them to interact with relevant data sources. It is relevant to measure if and how well the semantic model proves to be useful in tackling these issues. The following questions address a set of research questions by applying social semantic search to Research 2.0:

1. *How does the semantic (search) model reveal relations between resources interlinked in a scientific research context?*. Our approach and evaluation illustrates how to apply these paradigms for semantic search within Research 2.0.
2. *How well does the implementation of a semantic model enable researchers to find people (researchers), documents (papers) and events (conferences)?*, as well as some other related entities relevant for the context;
3. *How do researchers effectively search give a certain search context?*, for instance detect conferences, based on their earlier activities on social media;
4. *How does the proposed engine perform compared to a relevant semantic baseline?*

1.2 Methodology

The methodology focuses on the performance (efficiency), the user perception and information retrieval quality (effectiveness). Thereby we are testing (task-oriented) user experience and information retrieval aspects of each approach (such as search precision). We deduct as much as possible information out of the use case setting to address the research questions. Therefore, we implemented the semantic model specifically for the social semantic web search use case of research 2.0.

We considered using SPARQL[2] benchmarks[3], but it would not cover all implemented search functionality aspects. In the experience report (Uren et al., 2010) the authors: reflected about their experience over years on semantic search systems evaluation (i); concluded that such evaluations are generally small scale due the lack of appropriate resources and test collection, agreed performance criteria and independent performance judgment (ii); and proposed for future evaluation work: “developing extensible evaluation benchmarks and using logging parameters for evaluating individual components of search systems” (iii). Led by these findings and absence of adequate benchmarks that cover all facets of our approach we necessitated to define own user-centered methodology and benchmark for social semantic search (De Vocht et al., 2015).

The goal of the methodology and benchmark is to design a search engine and evaluate it with certain queries and datasets against a relevant baseline. It aggregates above mentioned and other related approaches and optimizes aspects for use with interactive exploration, social data, Linked Open Data and end-user involvement. Therefore, the benchmark requires test-user input to define the queries and measure the engine’s efficiency and effectiveness.

---

2 https://www.w3.org/TR/rdf-sparql-query/
3 https://www.w3.org/wiki/RdfStoreBenchmarking

2 Related Work

Some efforts regarding semantic search worth mentioning are shown in Table 1, which includes the system names and the references mentioned in this section.

Table 1 Overview on related work with their main contributions and key variables.

System	Reference	Main Contribution	Tested Data Sources	Results Format	Key Variables
Our approach		Iterative keyword search, reveal indirect connections between results	Mendeley, Twitter, COLINDA, DBLP[4], DBpedia[5], GeoNames[6]	Weighted graph based on the ranking	Efficiency, Effectiveness
EASE	(Li et al., 2008)	Adaptive keyword search, top-k ranking	DBLife[7], DBLP, IMDB[8]	List	Efficiency, Accuracy
Falcons	(Gong Cheng and Yuzhong Qu, 2009)	Keyword-based search system for linked data objects on the web	DBpedia, GeoNames, web crawler	List	Quality, Number of clicks
Hermes	(Tran, Wang & Haase, 2009)	Translating keyword queries to structured queries	DBLP, Freebase[9], DBPedia, semanticweb.org	List	Efficiency, Effectiveness
RelFinder	(Heim et al., 2009)	Systematic analysis of relationships in knowledge bases	DBpedia	Subgraph with keywords	Interactivity
LI	(Li, 2012)	Ranked top-k answers	semanticweb.org	Subgraph with keywords	Effectiveness
Mímir	(Tablan et al., 2014)	Combining full text search, structural annotation graph search, and SPARQL-based concept search	Annotated documents in various use cases	Multiple lists or sets	Extensibility, Effectiveness
S3	(Bonaque et al., 2016)	Top-k keyword search taking into account the social, structured, and semantic dimensions	Twitter, Vodkaster[10] and Yelp[11]	List	Efficiency, Qualitative Advantage
SemFacet	(Arenas et al., 2016)	Faceted search in RDF, establish computational complexity, updating faceted interfaces: critical in the formulation of meaningful queries.	Yago[12]	List	Efficiency

Hermes supports direct keyword translation, query expansion based upon SPARQL and is more generic as it does not focus on digital libraries and scientific research. It is a set-up with limited search personalization, this differentiates from our model. Our approach allows expanding the results using the paths among the matched keyword-entities over several iterations. Besides Hermes also *EASE*, *Falcons* and *LI* have been developed for retrieving semantic data from the Web. These engines primarily rely on keywords for the query specification. EASE is an XML-document based approach. Hermes also supports advanced querying capabilities, including basic SPARQL graph patterns. In general, the semantic matching frameworks within these search engines reside on graph pattern matching against Resource Description Framework[13] (RDF) data. *LI* introduced support for matching keywords within attributes and relations in the RDF data. The *RelFinder* tool extracts a graph covering relationships between two objects of interest and supports the systematic analysis of the

4 http://dblp.uni-trier.de/
5 http://dbpedia.org
6 http://www.geonames.org
7 (DeRose et al., 2007)
8 http://www.imdb.com/interfaces
9 https://developers.google.com/freebase/
10 http://www.vodkaster.com/
11 http://www.yelp.com/
12 http://www.yago-knowledge.org
13 https://www.w3.org/TR/rdf-schema/

discovered relationships by providing highlighting, previewing, and filtering. RelFinder resides on optimized SPARQL queries built on client side and in case of uncertainty disambiguates the results by sorting them by relevance using the SPARQL "count" feature for ranking. Unlike RelFinder, we target non semantic web expert users. A restriction of RelFinder is that users must supply valid entry points, SPARQL endpoints and repositories to query on, which requires technical knowledge.

The main contribution of *Mímir* lies besides advanced semantic search features in the natural text enrichment by first processing the documents containing it, making semantic search possible in non-structured documents. We work with structured data and assume that there are enough methods to convert unstructured documents into structured data and there is enough structured data present alread to meet the use case needs. In that regard our approach is more like *S3* where the documents (such as messages on social media) are considered as 'items' and the relations between items and social connections defined in the (meta)data are being taken into account in their model. *SemFacet* added a theoretical foundation for faceted search over RDF data, evaluated its implementation in terms of query performance and collected qualitative statistics about the facets.

Our approach does not only show the path between two related resources as a single (filtered on-demand) query result, it delivers multiple resource relations visualizations between the all found resources identifying their types and classes with the focus to the resource in center of the graph. By clicking on another resource the focus changes to the selection. All clicks are tracked and visualized as click path to show the user an overview over visited resources. The interface also includes paths and a query history. For ranking the results, our search solution considers two important aspects not offered by existing efforts; path distance to the selected resource (the proximity) and the pre-sets derived from the users' profile on social media.

3 Semantic Search Model

Users interact with the semantic model and the data model through multiple interfaces. The interfaces bridges each component in the search infrastructure. The top level interface, the user interface, delivers an aggregated and enriched view. All exposed content follows a common pattern from an aligned model resulting in a semantically interpreted and refined research repository.

3.1 Overview

The Semantic Search Engine (components shown in Figure 1) resolves queries consisting of one or more research concepts with refined entities, represented in the model as "Data Seeds". The Aligner allows configuring the interlinking of structured data, linked data (semantically described structured data) and Social Media data. The search interface allows researchers to browse and search for new research based on the researcher's previous tracked history and traversed paths (such as bookmarks or saved searches). Researchers interact with the system indirectly by contributing and monitoring posts and shares on Social Media using several Web 2.0 Tools.

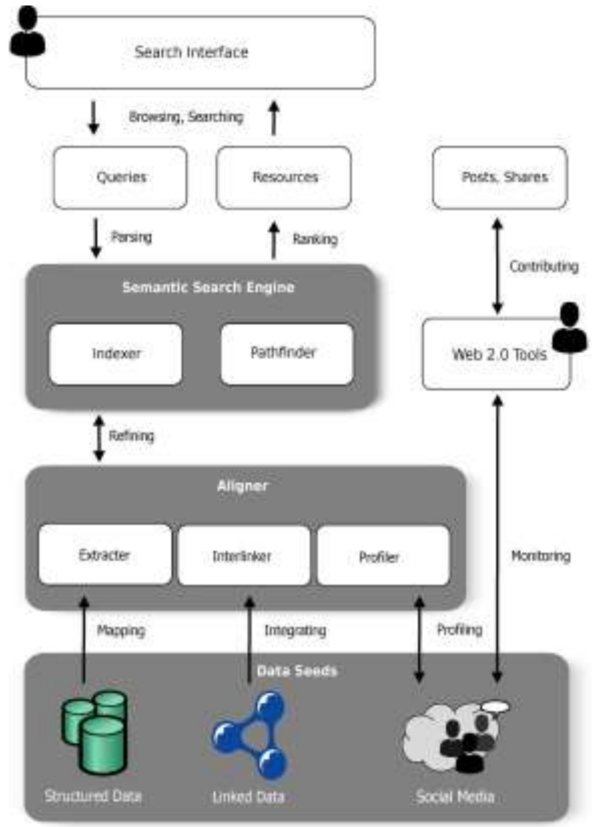

Figure 1 The system overview shows the key combination between the Aligner and the Semantic Search Engine as a bridge between the source data and researchers.

The system takes into account all relevant researchers' contributions to improve ranking found resources related to a search. Combining the Aligner and the Semantic Search Engine is an essential aspect for this infrastructure: the Aligner combines data from various heterogeneous sources configured in the Data Seeds and refines them for the Semantic Search Engine. The Semantic Search Engine parses queries and discovers relations between the resources.

3.2 Vocabularies

The Dublin Core vocabulary[14] was used together with the Semantic Web for Research Communities[15] (SWRC), the Semantically Interlinked Online Communities[16] (SIOC) and the Friend-of-a-Friend[17] (FOAF) ontology to describe the information on titles, authors, posts and descriptions. To describe tags, we used the Modular Unified Tagging Ontology (MUTO)[18], it integrates many prior defined tag ontologies. The MUTO instances resolve interlinking between the tags and conference labels (De Vocht et al., 2014) in Conference Linked Data[19] (COLINDA) (Softic & De Vocht et al., 2015).

14 http://dublincore.org/documents/dcmi-terms
15 http://ontoware.org/swrc
16 http://rdfs.org/sioc/spec/
17 http://xmlns.com/foaf/spec/
18 http://muto.socialtagging.org/core
19 http://www.colinda.org

3.3 Aligner

The Aligner module combines different social and online tools, such as Twitter and Mendeley. It interlinks data provided by the users (when they are using these social and personal media tools) to existing (Linked) Open Data such as DBpedia, GeoNames, LinkedGeoData[20], DBLP, and COLINDA. This interlinking allows enriching and connecting researchers to resources implicitly connected to them and thus initially not accessible. This allows to track communication on Social Media such as Twitter among researchers and relate it to publications and conferences. The Aligner module is optimized for Social Media and collaboration tools. Moreover, an important part of the alignment analysis, where access to restricted resources from users on Twitter and Mendeley is needed, happens on client-side. The results are aligned with the existing Linked Open Data. How and which components are integrated and implemented in the prototype is not the paper's main focus.

Profiler. When researchers sign up, they authorize access to their Twitter and/or Mendeley accounts. The Profiler extracts the timeline and followers of the user's social account and then annotates them using the FOAF and SIOC vocabularies. We link their author's profile to DBLP based on publication title and each publication's Digital Object Identifier (DOI). Listing 1 shows how we combine these identifiers with all author names and use them to find matching author identifiers in DBLP for each publication. For each article in a Mendeley account linked to a subscribing researcher it checks the DOI and publication title in DBLP and retrieves the authors. If a match occurs, the articles are aligned using *owl:sameAs*. If all author names of the publication match, we interlink the Mendeley authors with the DBLP authors based on their URI's. Because users linked their Twitter and Mendeley when signing up, the profiler links the author representation on DBLP with the author profile on Mendeley to the other social media user accounts and their contributions.

```
alignArticle ( mendeleyArticle )
title = find ( mendeleyArticle , "dcterms : title ")
articleAuthors = aligner . getAuthors ( title , article ) foreach
( articleAuthors -> ( dblpArticle , authors ))
     add ( mendeleyArticle , " owl : sameAs " , dblpArticle )
     foreach ( authors -> ( authorUri , authorName ))
          add ( articleUri , " dcterms : creator " , authorUri )
          persons = find (" foaf : name ", authorName )
          foreach ( persons -> person )
                add ( person , " rdf : type ", " foaf : Person ")
                add ( person , " owl : sameAs ", uri )
```

Listing 1: Aligning research publications.

```
annotateTag ( tag )
labels = store . find (tag , " rdfs : label ");
foreach ( labels -> ( label ))
foreach ( interlinkServices -> ( service ))
   meanings . add ( getMeaning( service , label ))
   store .add (tag , " muto : tagLabel ", literal ( label ))
   store .add (tag , " muto : tagMeans ", meanings )
```

Listing 2: Interlinking of tags

We use the social profiles of each researcher to allow personalized searches. The user profile extends the search context given a set of keywords. The original Social Media data needed to generate user profiles reside in-memory. After profiling the original tweets are erased after at most seven days. The profiling and analysis results are kept and indexed.

---

[20] http://linkedgeodata.org

Interlinker. Interlinking uses several steps to align various sources. The first step is to define which Linked Data sets to use in which context, to identify the vocabularies in them and to define which resource to link with resources occurring in another dataset. If the dataset is not available as Linked Data, then we must select a vocabulary to annotate the data. The case of Social Media is particular because Social Media content often exists of small posts and shares which we analyzed based on:

• URLs referring to and the content in it (enriched with recognized entities),
• hashtags and mentions included,
• entities occurring with the tweets.

After we extracted the URLs, hashtags, entities and mentions from each post in Social Media, we checked each of those against the Linked Open Data Cloud. COLINDA is used for matching conference hashtags, LinkedGeoData and GeoNames for locations, DBpedia for general concepts such as persons, places and events. DBpedia is well-connected to the GeoNames and DBLP which makes it a valuable source for search space expansion with more information about categories like cities and countries, persons or institutions. We give an example for the hashtags: after loading the interlink services (e.g. "colinda","geonames","dbpedia","dblp") from a configfile in a list *interlinkServices,* we do for each unique tag occurring in a microblogpost the actions listed in Listing 2.

Combining these approaches enriches tweets with Linked Data and is a good way to achieve optimal meaning. Entities occurring in the resources shared via the tweets lead to the best results (Abel et al., 2011). However, we have found in earlier research work that also the hashtags have consistent enough meaning for interlinking (Laniado & Mika, 2010).

Extracter. Most research information is unavailable as RDF, but available in relational databases as tables or spreadsheets. To make this data available in RDF we use two types of processes: (i) pre-defined (static) annotations using the resource provider's API to load the information from the data repository; and (ii) dynamic mapping between the ontology and the data repository. Each time when a certain source provides access to their structured content, the Aligner makes sure that provided content is converted conform our data model. Therefore, it selects configured properties and maps them using the supported vocabularies and triplifies them.

3.4 Semantic Search Engine

This module refines the aligned data, parses queries against it and ranks matched resulting resources. The *Pathfinder* module retrieves resources via the *Indexer* module. The Indexer pre-optimizes and stores each resource by URI and label to serve them. We have used an implementation that relies on our earlier work finding paths (De Vocht & Coppens et al., 2013).

For all data sources we make sure that we describe the semantic model using mapped and applied vocabularies so we can expose them using a uniform interface and representation, such as RDF. The engine processes keywords by first doing the keyword-to-resource mapping. With each user action, researchers can input keywords and interact with results over several iterations, the query translation within the search engine is triggered.

4 Ranking of Results

Ranking resources and relationships in the Web of Data differs from traditional document ranking because semantic search engines can take into account the meaning of the relations between resources in the results and others not included. Aleman-Meza et al. demonstrated the effectiveness of a ranking approach that distinguishes between statistical and semantic metrics (Aleman-Meza et al., 2005). They used proximity to the search context as an important metric. Because it is critical for the success of an interactive tool for research (Marchionini, 2006), the ranking should take into account the discovery of newer unexpected relations. This has been applied in SemRank, which is a method for scoring semantic relations in search results and configuring how high the most unpredictable, unlikely paths should be ranked (Anyanwu et al., 2005). Daoud et al. have shown the effectiveness of a personalized graph-based ranking model (Daoud et al., 2010).

By considering cross links between graphs and distances between nodes, we achieve personalization by affecting the original resource ranking. Pintado et al. identified relationships, using dynamical an statistical analysis, between classes and objects and used metrics to quantify these relationships in order to express them in terms of object affinity in a Software Engineering context (Pintado, 1995). Their goal and interface is similar to ours and the introduced concept 'affinities' is characterized by high levels sharing of similar properties and relations. Therefore, we apply this concept as a base for defining our ranking approach.

4.1 Pre-Ranking

Before we rank the relations between resources, the candidate resources to be included in relations are pre-ranked. The pre-ranking take into account "popularity" and "rarity", essential components in the PageRank algorithm (Page et al., 1999), and is used to sort candidate related nodes in the proposed engine. The implementation takes these relations into account by using the Jaccard-coefficient to measure the dissimilarity and to assign a random-walk based weight, which ranks more rare resources higher, thereby guaranteeing that paths between resources prefer specific relations over general ones.

4.2 Affinity Ranking

We identified three important criteria for ranking in a search engine according to our semantic model: (i) *proximity* to the search context C_R; (ii) *personalization* P_R and (iii) discovering newer unexpected relations, the *novelty* N_R, because we want to exceed predictable fact retrieval. Alternatively, we quantify the relationships to help researchers focus. These metrics are always executed between an object pair. The path between them represents whether they are directly connected or not. The results are limited and optimized according this ranking mechanism.

This section gives an overview on important semantic ranking criteria and explain why they are useful for our affinity ranking approach and discuss how they contribute to measuring affinity for a resource A_R. We define this hybrid ranking criterion as:

$$A_R = w_c * C_R + w_n * N_R + w_p * P_R \tag{1}$$

where we make sure that the weights are normalized to an application global configured constant k:

$$k = w_c + w_n + w_p \tag{2}$$

For each criterion users can configure a weight w, this can be used to optimize the focus on resources. In our evaluation we show the effectiveness of our search infrastructure with the presented ranking criterion and make a distinction between a personalized ($w_p = w_c$) and anonymous search case ($w_p = 0$). The proximity to the search context C_R is a main indicator of affinity. Novelty N_R and personalization P_R then refine the ranking further. It is very important that the weights in the affinity criterion A_R are accurately configured. Novelty becomes more important when differences in type of relations are essential, so w_n should be relatively high. The amount of personalization can be taken into account as well by making w_p greater than 0, in the order of magnitude of w_c. All weights are relative to the proximity, which always is taken into account ($w_c > 0$). The weights depend on the application and the goal of the use case.

Proximity. In our case, the proximity to the context marks the number of relations found in a path between two resources, that belong to the search context. The context can be initiated by a user profile if the user so desires. Found resources can be related to it to personalize the ranking. In an anonymous search, the relationships binding the resources that represent the researchers input query keywords determine the context. We measure "proximity" - *how semantically related resources are*. Objects being close in one context can seem quite unrelated in another context. Distance between the resources (path length) offer a different perspective on this ranking criterion for the context. The further the distance between two resources is, the less related they are, since the increasing distance between the two resources also brings with it the fact that they do not relate to each other, but have common intermediate resources which relate to them both. This on its own does not guarantee a high quality relation between the two resources at the start and end of the path.

After we have defined the resources and relations belonging to the context C we define for each other resource R, outside the context, the proximity criterion C_R such that

$$d = distance(C, R) \tag{3}$$

where $distance(C, R)$ is the weighted optimized shortest distance between any resource C_k in the context and R, as computed by the search engine. We use d to normalize the expression and then look for each relationship R_i in the path between C and R whether it belongs to the context or not. The path between C and R can be noted as: $(C_k, ..., R_i, ..., R)$.

$$x_i = \begin{cases} 1, & : R_i \in C \\ 0, & : R_i \notin C \end{cases} \tag{4}$$

$$C_R = \frac{\sum_{i=0}^{d-1} x_i}{d} \tag{5}$$

The distance d is at least 1. The context consists of the mapped keywords, the relations between those resources and their properties.

Novelty. In research, unexpected discoveries make interacting with the search results more interesting. Affinity with resources in research is affected by new discoveries and always searching within the same kind of resources and relationships does not guarantee it. We want to encourage sudden paradigm shifts in paths. More shifts lead to higher novelty. This means that if a path switches from relations that describe people to relations that describe countries, the novelty score will be high, depending on how different the new paradigm is from the original and how many shifts there are.

We compute novelty N_R for a resource R along the relations belonging to the path from the nearest resource c of the search context C. We need to define the domain D_i for a relationship R_i, typically these are all other predicates for which there exists a connection to, such that

$$n_i = \begin{cases} 1, & : R_i \notin D_{i-1} \\ 0, & : R_i \in D_{i-1} \end{cases} \tag{6}$$

which means that we check whether R_i belongs to the domain of the previous relation R_{i-1}. Except for the first relationship we can thus compute the novelty of the relation belonging to the path between C and R.

$$N_R = \frac{\sum_{i=1}^{d-1} n_i}{d-1} \tag{7}$$

We note that $N_R = 1$ if none of the relations in the domain of the previous relation and $N_R = 0$ if all relations belong to the same domain.

Personalization. To ensure a personal ranking we need to connect the found resources with the researcher's profile. Both are represented as Linked Data graphs. We merge the graph of resources and the graph of the user profile through common concepts and cross links connecting the two graphs. Even in anonymous search sessions, we optimize the ranking of the results according to the users search context defined by the input keywords and selected resource representations.

We define u as a property of the user U. We compute the personalization criterion P_R for a resource R as the averaged sum of all properties of R related to the personalized context, which consists of the properties u of the user U, resulting in following equations:

$$d_u = distance(R, u) \tag{8}$$

where the distance d_u between R and u is computed along the path between R and u.

$$R_u = \frac{1}{d_u} \tag{9}$$

The inverse distance $R_u = 0$ if there is no path. We compute P_R by iterating over each $u \in U$.

$$P_R = \frac{\sum_{u \in U} R_u}{|U|} \tag{10}$$

5 Evaluation

We consider non-Linked Data domain experts to be the typical users in the Research 2.0 case. Another user group are domain experts, they are likely to have an understanding of the data structure and content in their domain, and bring this knowledge to guide both browsing research and targeted searches. This section provides information about the measures, datasets and reports on the applied and executed benchmark results for the experimental setup we implemented for our use case.

5.1 Measures

We measure the *efficiency* and *effectiveness* to obtain insight in how well the proposed engine performs and its individual components interact.

Efficiency. The efficiency learns how the engine with its implementation E behave when parsing queries, such as the test set Q. The efficiency is divided in three independent sub-measures: *quality* (i), *complexity* (ii) and *performance* (iii). The quality indicates how much relations between concepts after translating the keyword queries were found. Complexity and performance focus on time and space (memory-usage) requirements for executing the translation and finding these relations.

Effectiveness. Effectiveness E on the other hand indicates the overall perception of the results by the users taking into account expert-user feedback. This is expressed as the search precision P (Powers, 2011).

$$P = \frac{\# relevantobjects}{\# retrieved objects} \tag{11}$$

The reason why we have measured precision but not recall is because computing relevant results for the entire dataset is complex due to its size and dynamic nature (D). However, we compute the relevance for each result set. Each query $q_i \in Q$ delivers a different number of relevant results, which makes the mean average precision MAP an important measure. The aim of this averaging technique is to summarize the effectiveness of a specific ranking algorithm over the query collection Q.

$$AvP(q_i) = \frac{\sum_{k=1}^{A_i} P(k) \cdot rel(k)}{\# relevantobjects} \tag{12}$$

where A_i is the number of actions taken by the user when resolving the query q_i and $P(k)$ is the precision in the result set after user action a_k in search iteration $k-1$ via the interface V and $rel(k)$ equals to 1 if there are relevant documents after a_k and 0 otherwise. As a result, the items contained in $P(k)$ are k (where $k > 0$) steps away from the matched keyword search context items $P(0)$.

$$MAP = \frac{\sum_{q_i \in Q} AvP(q)}{|Q|} \tag{13}$$

5.2 Queries

We have collected for the evaluation typical keyword queries that have been asked by the the use case target group ($N = 36$ users) (Dimou et al., 2014), both researchers and innovation policy makers - all in the information and communication science field, against the system during the "usefulness" evaluation. They judged their experience with search interface on a Likert-Scale with values (Strongly Disagree, Disagree, Undecided, Agree, Strongly Agree). The evaluation result is shown in Table 2. According to users, the interface is meant primarily to serve as an exploration interface which makes our approach focused more on the user experience and less on classical search issues.

Table 2 The short survey on the *perceived usefulness* results.

Answer	Score	Variance
. **Explore**	**4.00**	2.00
. Discover	3.89	1.65
. Search	3.42	1.70
. Analyse	3.05	1.72
. Clarify	3.00	1.78
. **Tell stories**	**2.35**	1.62

For the evaluation, we restricted our tests to 10 queries which are answerable by the data sets we indexed. These are shown in Table 3. These queries cover some commonly used search terms within a researcher context: Search for an event ($Q_{1,2,3,4,9,10}$), a person, author or group of authors ($Q_{1,5,6,7,8,9,10}$) or scientific resources ($Q_{1,2,3,6,9,10}$).

Table 3 Selected queries by test-users, keywords matching to loaded user profiles are underlined.

#	Keywords
Q_1	LDOW, Bizer
Q_2	ISWC2012, Lyon, France,
Q_3	ISWC2008, Linked Data, Germany
Q_4	Linked Data, WWW2012
Q_5	<u>Selver Softic</u>, Semantic Web, Michael Hausenblas
Q_6	<u>Selver Softic</u>, Linked Data, Information Retrieval
Q_7	<u>Laurens De Vocht</u>, <u>Selver Softic</u>
Q_8	<u>Laurens De Vocht</u>, <u>Selver Softic</u>, 2011
Q_9	<u>Laurens De Vocht</u>, Linked Data, WWW2013
Q_{10}	Chris Bizer, WWW2013, ISWC2010

Each search runs through the following scenario: users enter the first keyword and select the matching result that is resolving their search focus at least one step forward. The users view selected results and can expand them at any time except when the research selects the suggestions from a typeahead interface. Parallel with this selecting and narrowing down the scope, our engine finds relations between the resources and reflects the context. Additionally, neighbours which match the selection are found. In the case that users logged in via their Twitter account or Mendeley account or both at same time, their research profiles personalize the boundaries of the search space.

5.3 Experimental Setup

We developed a topological radial graph interface (De Vocht et al., 2013). It focuses on mainstream users who do not understand the RDF model or have knowledge about Linked Data technologies. An example query for the Semantic Relations between the resources *Linked Data*, *WWW2012* is explained in Figure 2, it demonstrates the interface for querying with a single search input field, the visualized relations between found resources and the search track (previous searches). Researchers can use this interface to explore and define searches without users having to manipulate different criteria values.

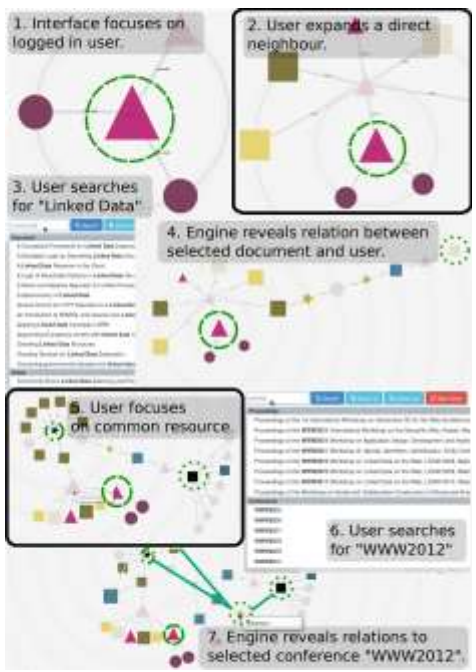

Figure 2 We see in the series of seven screenshots that the logged in user, represented as a triangle encircled with a dashed line (1, 2), is focused in the center. After searching for "Linked Data" (3), the selected article match, encircled with dots, and other

5.3.1 Datasets

The datasets used in our experiment, combine existing Linked Open Data sets: DBpedia, DBLP and GeoNames interlinked with research oriented datasets such as COLINDA and a Social Linked Data set containing information about conferences and the researchers' social profiles from Twitter and Mendeley and the data they generated. For the evaluation we use COLINDA to resolve the connections between GeoNames, DBpedia and DBLP since it has links to these three Linked Datasets. Further it serves as a conference entity resolver for social data used with the user profiles from Twitter and Mendeley. Table 4 highlights statistics on the used datasets.

Table 4 Linked Data used within the search experiments.

Dataset	#Triples	#Instances	#Literals
DBpedia	332 089 989	27 127 750	161 710 008
DBLP (L3S)	95 263 081	13 173 372	17 564 126
COLINDA	143 535	15 788	70 334
Social LD	41 438	7 344	15 350

5.3.2 Index Configuration

Table 5 shows the indexed datasets' statistics. The total time for building all indices for all the data sources is about 6 hours (throughout all the experiments, we use a 8-core single-machine server with 16GB RAM running Ubuntu 12.04 LTS). The properties *type* and *label* are indexed, because they are required for each Linked Data entity described in RDF[21] and allow retrieving entities by label and disambiguating them by type. The indices contain a special type of field *ntriple* that makes use of the SIREn Lucene/Solr plugin that allows executing star-shaped queries on the resulting Linked Data (Delbru et al., 2012). Star-shaped queries are essential to find neighbouring entities for each entity and to find paths between non-adjacent nodes.

Table 5 Resulting index properties based on input datasets.

Index	#Resources (K)	Temp Space (MB)	Size (MB)
DBpedia	28 384	38 000	30 000
COLINDA + DBLP (L3S)	3 307	15 000	12 000
Social LD	7	5	170

To ensure maximal scalability and optimal use of available resources, we use simple, but effective measures based on topical and structural features of the entities in the search engine. Relations are computed between pairs in a subgraph of the larger dataset. Every resulting relation as a path between entities are examined for ranking. Entities belonging to a specific search context are requested. Since the result set might be very large, this "targeted" exploration of relations is essential for the efficiency and scalability.

5.3.3 Resource Alignment

Our earlier results, for several user profile types using Twitter and Mendeley to varying degrees, indicate sensitivity, precision and accuracy when linking tags, authors and articles to conferences (De Vocht et al., 2014). Conference tags were better recognized than other tags, this is not surprising because we optimized our model for this task. We never obtained false positives when interlinking authors and articles. When we interlinked followed users on Twitter as authors, we encountered a high amount of negatives. All found links of users as authors were correct but there is room for reducing false negatives.

[21] http://www.w3.org/2009/12/rdf-ws/papers/ws17

6 Results

We have evaluated the efficiency and effectiveness of the proposed semantic engine and our environment in terms productivity. Therefore, we executed consecutive benchmarks and each time we tweaked the back-end performance. We tested the retrieval quality with the test queries shown in Table 2. The carried out user questionnaires and expert feedback rounds identified the functional needs of the search infrastructure in the original use case (De Vocht et al., 2011).

6.1 Efficiency

In order to measure the efficiency our approach, we stored data about all executed queries: source, destination, all the path hops with the links between them and the execution time. We qualify the combined datasets and our algorithm by measuring the average path length and the resolved paths. A found path is relevant if it belongs or has entities relevant to the search context. We measured the hit-rate, execution time distribution and path lengths for a test set.

Quality. The queries Q_1 to Q_{10} were translated into 576 pathfinding queries between pairs of resources of wich 400 were connected. The hitrate is about 76%, which is high, considering the small number of resources that had to be checked compared to the the entire dataset size (31.6M resources). Checking a resource means retrieving the resource from the index and identifying the linked resources (neighbours).

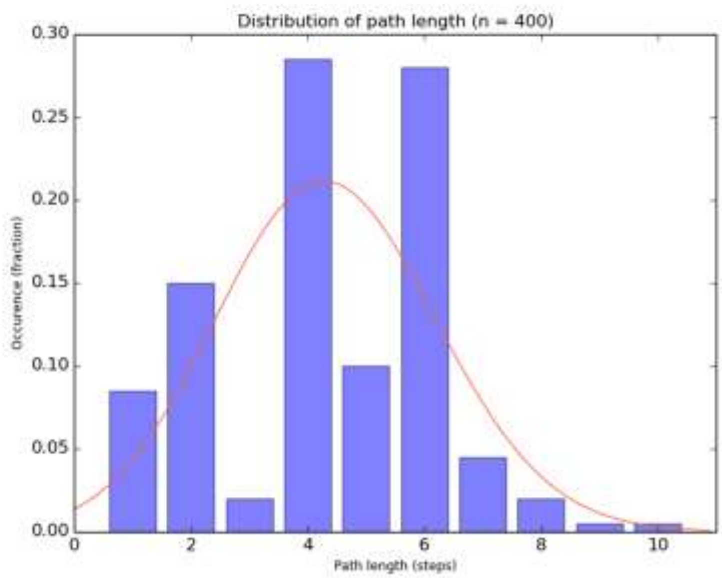

Figure 3 There are an unexpected low number of paths with length 3 and 5.

The calculated path lengths are between 0 and 8 hops, a clear majority is between 4 and 6 hops as shown in Figure 3. Paths of length 3 and 5 hava an unexpectedly low occurence. This due to the focused nature of the search queries and the resulting manageable number of pathfinding queries. The average path length is close to 4.

Complexity. Figure 4 and Figure 5 show the time and space complexity. Except for path with length of 3. The average complexities do not increase obviously linearly or exponentially.

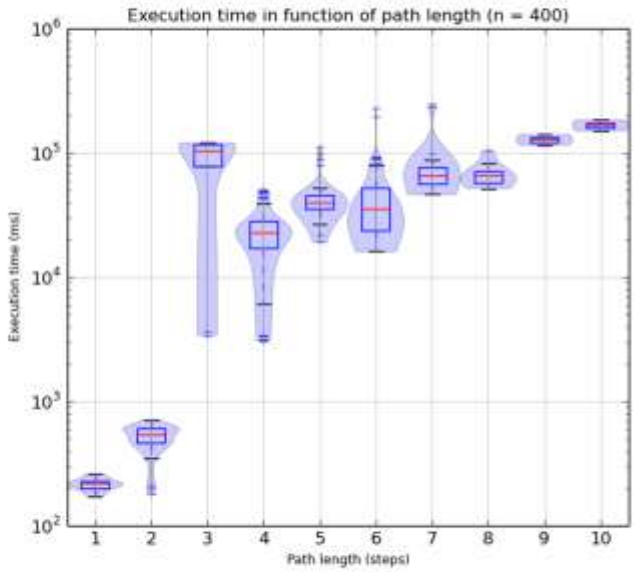

Figure 4 Time complexity on a logarithmic scale.

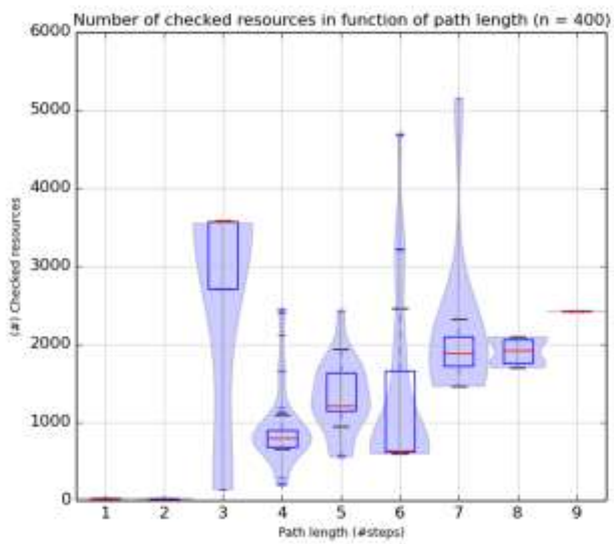

Figure 5 Space complexity.

The current results were more volatile and had pinpointed unexpected deviations with path lengths 3 and 5. This is because the queries were not randomly chosen, the number of queries was much smaller and the dataset is not homogeneous. Some paths hop between datasets while others do not. These peculiarities could not occur in the original evaluation where we isolated the pathfinding module with a single index (DBpedia) loaded (De Vocht & Coppens et al., 2013). This finding is neither 'good or bad', but it is noteworthy that the selection and the nature of datasets does impact the path length distribution and influences time and space complexity.

Performance. The algorithm's performance is promising, even though the configuration was not optimized for speed,but for quality, and was run for the evaluation on a single server, the algorithm found over 25% of paths in a couple of seconds. Within 30 seconds it found already results for over 50% of the path queries. However, there is room for improvement as the more complex queries take much more time to execute. Resolving a keyword and retrieving the matching entities happened instantly. Figure 6 shows the execution time distribution. The search interface and the search engine execute the necessary queries asynchronously and in parallel. While executing the queries – and results are coming in – the user can start exploring.

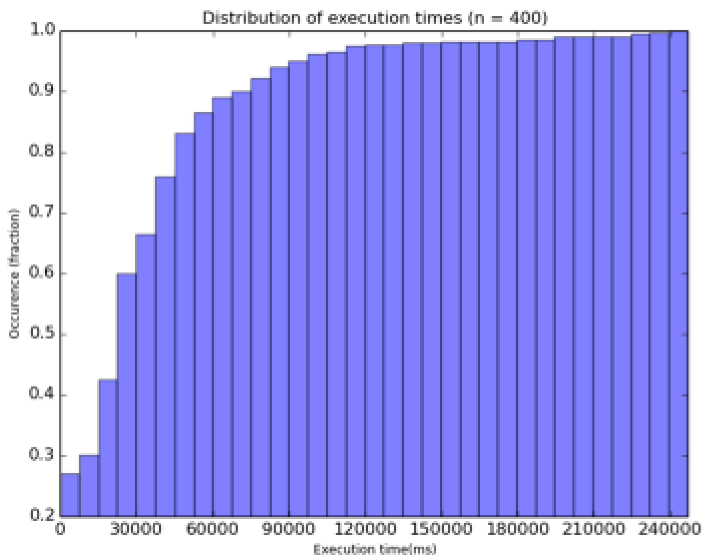

Figure 6 More than half of the relations are found in 30 seconds.

6.2 Effectiveness

Based on the recommendations and insights after initial test runs, we re-evaluated our approach with a specific focus on independent query result judgments and we compared it to a state-of-the art baseline aiming at confirmation of our achieved good results on retrieval.

6.2.1 Baseline

Virtuoso is one of the most common triple stores. It has support for the - non-standard SPARQL - transitive paths and has its own built-in index for text search (via the *bif:contains* property). In many projects dealing with the same amount of data(sets) as we did, it would be the de-facto choice. Therefore we consider it as a baseline for our solution. For the benchmarks we used version *6.1.3127*. We compare this way, executing the same ‘underlying’ SPARQL transitive queries and also the keyword queries.

Two expert-users independently evaluated the baseline results. To verify if the judgment across both is similar enough to be considered we checked the *F-Measure* (or *positive specific agreement*) of **0.68** and the *chance corrected agreement* (or *inter-rater agreement*): $\kappa = 0.62$ (where always $-1 < \kappa < 1$).. According to Landis et al. (Landis & Koch, 1977) this level of agreement is *substantial*.

The mean average precision, **MAP** for the baseline is **0.52**.

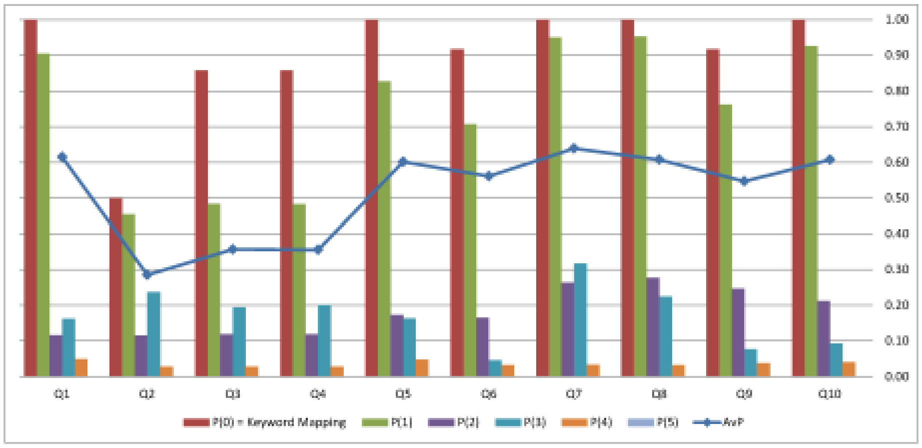

Figure 7 Precision results of the baseline for the test queries.

The results delivered by the inter rater agreement and baseline approach confirm our assumption about very solid retrieval responsiveness with traditional SPARQL queries, except the results from P(2) on are quite low.

6.2.2 Proposed Engine

To assess the effectiveness of query translation, the same expert users measured the precision and the mean average precision over all queries to evaluate that the search algorithm used in our search engine returns enough high quality relevant results for researchers to achieve their research goals effectively. There was a *F-Measure* (or *positive specific agreement*) of **0.90** and a *chance corrected agreement* (or *inter-rater agreement*): $\kappa = 0.82$. According to Landis et al. (Landis & Koch, 1977) this agreement level is *almost perfect*.

In order to assess our search system we measured the precision of the results for the queries in Table 2. To determine the relevance of each resource we relied on expert judgment and we verified expected results against the system's output according to the ranking mechanism. We defined what the expected outcome scenario was based on familiarizing with each of the visualized keyword searches and than having an expert compare the system's output against the predefined scenario by checking each visualized item one by one after each expansion.

Additionally, within the measurements we used personalized data to generate a user profile and project the expected search results. This extension is specifically important in the case of queries with *Selver Softic* and *Laurens De Vocht*, where we loaded these test user profiles. We measured effectiveness using the search interface specified in subsection 5.7.3 and as described in subsection 5.6.

We judged each result to enable a more accurate evaluation of the context driven aspect. Personalized queries Q_5 - Q_9 have been strictly evaluated: A found resource is irrelivant if it is without direct relation to the persons or event or topic specified by keyword, even if it is in wider context relevant (for instance a co-author that corresponds to the person but does not fit to the specified event).

Figure 8 shows the precision over queries. With exception of Q_1, Q_4 and Q_{10}, queries with preloaded profile data (Q_5 - Q_9) deliver more precise results than anonymous queries. This difference is because the main focus of queries Q_5 - Q_9 is a person which resolves good within key mapping step, thus following results keep the average precision high. Queries Q_1, Q_4 and

Q_{10} have very high precision since they have broader focus which includes more relevant results. Also the keyword choice matches with the linked dataset instances within COLINDA and DBLP. The Mean average precision **MAP** overall as expected reaches the score of **0.60** which is high but not surprising since the resources within the linked datasets are well-connected and all used datasets interlinked. The **MAP** we measured in is 8% higher than in baseline case. This first impression strengthens our first evaluation and brings us more near to the confirmation of hypothesis. Figure 8 shows the precision results of the queries for each path length. As expected, the precision decreases with the length of paths in most cases. As the path finding progresses over extended links relation to the core concept is becoming weaker. It is promising that the first step of keyword search as well the path finding results of length one deliver always the results that exceed the value of the mean average precision.

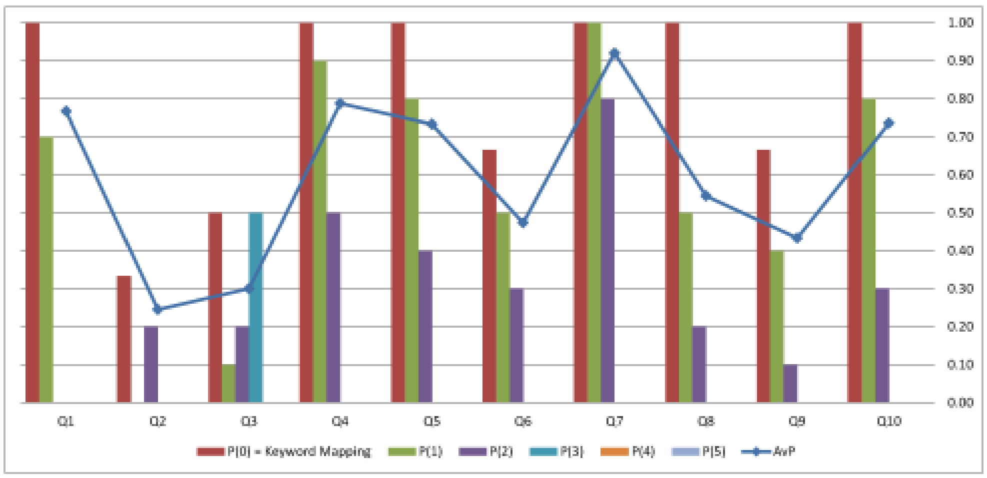

Figure 8 Precision of the proposed engine for the test queries over different path lengths.

6.2.3 Comparative Analysis

We compared the precision of both result sets. We have the baseline, Virtuoso, vs. our proposed semantic engine. While we could just average the expert results or choose one of the results as a reference we detect the overall tendencies that reoccur since the inter-rater agreement is sufficiently high, but we also learn about the cases where they disagree (Demeester et al., 2014). Therefore, we looked at two scenarios: a strict scenario (both need to agree on relevancy, Figure 9) and a tolerant scenario (at least one needs to judge a result relevant, Figure 10).

To be able to compare the results we included precision that makes sense until a certain level. Our engine did not contain any items beyond a certain level, $P(3)$ in most cases, this means that the displayed results are all contained within a range of 3 steps from the matched search keyword context. The baseline results are very low from $P(4)$, a couple of resources at this distance from the search context were considered relevant and at $P(5)$ there are no results either. We choose an strict and tolerant scenario where we either require both experts to judge an element relevant or not. We computed the difference between the precision expressed as $\Delta P = P_{proposed} - P_{baseline}$.

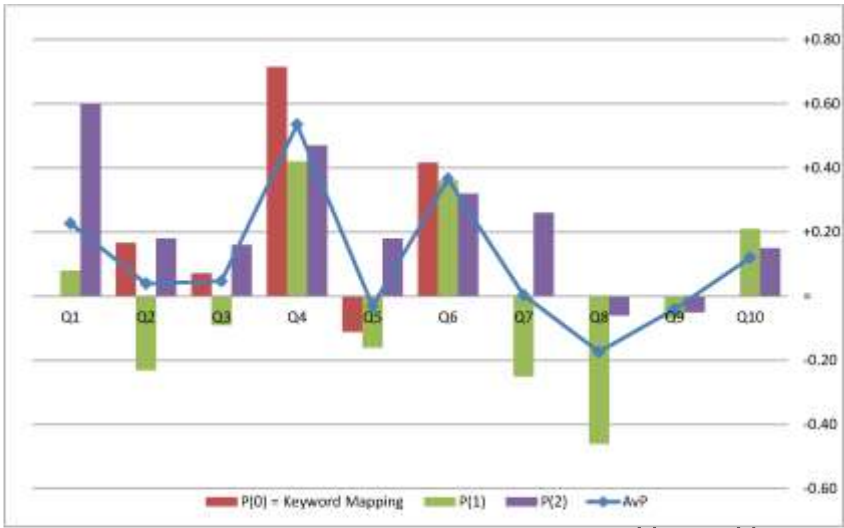

Figure 9 The strict delta precision is overall better for the proposed engine except Q_5 and Q_8. The better results at $P(2)$ is remarkable.

Overall we see the tendency that the proposed engine performs more precise or on par (very little difference) with the baseline, except for Q_5 and Q_8. Q_8 because of the failed interpretation by our engine of *2011*. In either case the precision is moderate.

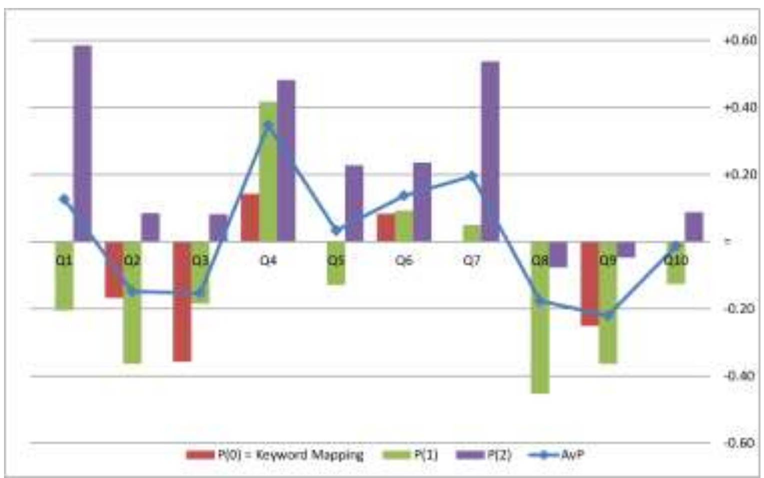

Figure 10 The tolerant delta precision is overall similar to the strict delta precision, but where the proposed engine's precision is a bit less distinct and the results Q_2, Q_3 and Q_9 have less precision compared to baseline and the strict delta precision.

In the tolerant scenario we detect overall similar results, but they are more clear-cut, except for Q_2, Q_3 and Q_9. Mainly due to improved results for $P(0)$ and $P(1)$ for the baseline. In Q_2, which was on par in the strict case, scores here better for the baseline, because at $P(1)$ one of the reviewers thinks the baseline is more precise. Q_2 is a tricky query because, *ISWC2012* did not take place in *Lyon,France*, depending on the exact results that were given to query and their relation to the other terms, it is left open to interpretation. The difference is even more distinct for Q_9, the baseline scored better in the strict case. This is because we found there a larger part of the results at $P(1)$ relevant. Q_9 is contains a topic keyword, so it is not trivial for an (expert) user to judge whether the results matching this keyword were relevant to both of the keywords or not. We also see in $Q_2 - Q_3$ that the judgment of $P(0)$ is on par in the tolerant case but much worse for the proposed engine, this is because that the expert users did not agree about the relevance of the keyword mapping in the proposed engine. There is a strong similarity of the results of Q_1, Q_4, Q_6, Q_7 and Q_8. These are also the cases where the proposed engine has the highest precision, this finding is backed with a strong agreement between the raters for both systems.

7 PhD Contribution

The work in this paper connects to the broader scope of my PhD by explaining the efficiency and effectiveness trade-offs for query processing for a semantic search use case. The presented semantic search model forms a bridge between the content from a user's perspective and the representation as linked data.

Overview. Formulating search queries from a users' point of view is difficult in case of linked data sources, because they contain many different relationships and are often described by a wide variety of vocabularies. Because most users cannot realistically construct their intended search query correctly at the first attempt, they benefit from an environment in which they can iteratively refine what they are searching for. Therefore, my PhD proposes a set of techniques to develop semantic search engines that drive this kind of environments and implements them for several use cases and measures the performance and the effect on the search precision.

Discussion. The semantic search model contributes to data authenticity by guaranteeing that the final output towards the user has useful results in the application domain. Because the model works with a linked data structure, this method is applicable to other domains if it is structured by adapting the chosen vocabularies according to the datasets used.

In semantic search scenarios, intermediary link dynamics leading to relevant discoveries are important to take into account. Therefore, the evaluation did not only focus on pure information retrieval metrics, such as precision (which is more biased towards the final results), but also highlighted how the search effectiveness was gradually influenced by the user's actions.

In terms of *effectiveness*, the proposed engine is more precise for well-defined query contexts, i.e. consisting of keywords in which the meaning is unambiguous, for example when a specific conference, author or publication are combined in a search. On the other hand when there are inconsistencies or vague terms, such as topics or years, even mismatches in the query context, expert users disagree about the effectiveness.

In terms of *efficiency*, facilitating exploration and search across semantically aligned data sources is feasible as the evaluation showed that with a linear execution time complexity (scaling with increasing number of hops between resources) and an optimized space complexity, the presented approach is able to outperform the raw SPARQL baseline in query contexts that are well-defined, i.e. consist of keywords in which the meaning is unambiguous, for example when a specific conference, author or publication are combined in a search. The typical alternative – constructing separate search queries for each of those sources – is a laborious task.

Future work. Two aspects to further investigate are: *generalization* and *scalability*.

Generalization: looking into the impact on the behavior and best practices when extending the existing data selection with data from other relevant scientific archives or from non-technical areas to enrich it with sources for a specific topic. For instance, integrate Life Science archives like PubMed[22] and other scientific publication archives like BibBase[23]. More variation could introduce the need for more vocabularies which in turn might lead to more complicated queries.

Scalability: while the current implementation is able to handle the search queries and interlinking on a single machine, possibilities for upscaling and distributing need to be investigated so the engine can be deployed to support more concurrent users and bigger datasets.

Committee. I would like to thank my supervisors Ruben Verborgh and Erik Mannens for their support and the opportunity for the realization of this work; and my co-authors Selver Softic and Martin Ebner for the valuable joint research we did while implementing this use case.

Acknowledgements

The research activities that have been described in this paper were funded by Ghent University, imec, Graz University of Technology, Flanders Agency for Innovation and Entrepreneurship (VLAIO), the Fund for Scientific Research-Flanders (FWO-Flanders), and the European Union.

[22] http://linkedlifedata.com/resource/pubmed

[23] http://data.bibbase.org/

References

(Abel et al., 2011) Fabian Abel, Qi Gao, Geert-Jan Houben, and Ke Tao. Semantic enrichment of twitter posts for user profile construction on the social web. In *The Semanic Web: Research and Applications*, volume 6644 of *Lecture Notes in Computer Science*, pages 375–389. Springer, 2011.

(Aleman-Meza et al., 2005) Boanerges Aleman-Meza, C Halaschek-Weiner, I Budak Arpinar, Cartic Ramakrishnan, and Amit P Sheth. Ranking complex relationships on the semantic web. *Internet Computing, IEEE*, 9(3):37–44, 2005.

(Anyanwu et al., 2005) Kemafor Anyanwu, Angela Maduko, and Amit Sheth. Semrank: ranking complex relationship search results on the semantic web. In *Proceedings of the 14th international conference on World Wide Web*, WWW ’05, pages 117–127, New York, NY, USA, 2005. ACM.

(Arenas et al., 2016) Arenas, Marcelo, Bernardo Cuenca Grau, Evgeny Kharlamov, Šarūnas Marciuška, and Dmitriy Zheleznyakov. Faceted search over RDF-based knowledge graphs. *Web Semantics: Science, Services and Agents on the World Wide Web* 37: 55-74, 2016.

(Auer et al., 2012) Sören Auer, Jan Demter, Michael Martin, and Jens Lehmann. Lodstats - an extensible framework for high-performance dataset analytics. In, *EKAW*, volume 7603 of *Lecture Notes in Computer Science*, pages 353–362. Springer, 2012.

(Bakshy et al., 2011) Eytan Bakshy, Jake M. Hofman, Winter A. Mason, and Duncan J. Watts. Everyone’s an influencer: quantifying influence on twitter. In *Proceedings of the fourth ACM international conference on Web search and data mining*, WSDM 1′1, pages 65–74, New York, NY, USA, 2011. ACM.

(Bonaque et al., 2015) Bonaque, Raphaël, Bogdan Cautis, François Goasdoué, and Ioana Manolescu. "Social, Structured, and Semantic Search. In P*roceedings of the 19th International Conference on Extending Database Technology*, EDBT’16, pages 29–40. OpenProceedings, 2016.

(Daoud et al., 2010) Mariam Daoud, Lynda Tamine, and Mohand Boughanem. A personalized graph-based document ranking model using a semantic user profile. In *User Modeling, Adaptation, and Personalization*, pages 171–182. Springer, 2010.

(Delbru et al., 2012) Renaud Delbru, Stephane Campinas, and Giovanni Tummarello. Searching web data: An entity retrieval and high-performance indexing model. *Web Semantics: Science, Services and Agents on the World Wide Web*, 10:33–58, 2012.

(Demeester et al., 2014) Thomas Demeester, Robin Aly, Djoerd Hiemstra, Dong Nguyen, Dolf Trieschnigg, and Chris Develder. Exploiting user disagreement for web search evaluation: an experimental approach. In *Proceedings of the 7th ACM international conference on Web search and data mining*, pages 33–42. ACM, 2014.

(DeRose et al., 2007) DeRose, Pedro, Warren Shen, Fei Chen, Yoonkyong Lee, Douglas Burdick, AnHai Doan, and Raghu Ramakrishnan. "DBLife: A community information management platform for the database research community." In *CIDR*, pages 169-172. 2007.

(De Vocht & Coppens et al., 2013) Laurens De Vocht, Sam Coppens, Ruben Verborgh, Miel Vander Sande, Erik Mannens, and Rik Van de Walle. Discovering meaningful connections between resources in the web of data. In *Proceedings of the 6th Workshop on Linked Data on the Web (LDOW2013)*, 2013.

(De Vocht et al., 2015) De Vocht, L., Softic, S., Verborgh, R., Mannens, E., Ebner, M., & Van de Walle, R. Benchmarking the Effectiveness of Associating Chains of Links for Exploratory Semantic Search. In *Proceedings of the 4th Workshop on Intelligent Exploration of Semantic Data at the 14th International Semantic Web*, pages 1-13, CEUR-WS.

(De Vocht et al., 2014) Laurens De Vocht, Selver Softic, Erik Mannens, Martin Ebner, and Rik Van de Walle. Aligning web collaboration tools with research data for scholars. In *Proceedings of the Companion Publication of the 23rd International Conference on World Wide Web Companion*, WWW Companion ’14, pages 1203–1208, 2014. International World Wide Web Conferences Steering Committee.

(De Vocht et al., 2013) Laurens De Vocht, Erik Mannens, Rik Van de Walle, Selver Softic, and Martin Ebner. A search interface for researchers to explore affinities in a Linked Data knowledge base. In *Proceedings of the 12th International Semantic Web Conference Posters & Demonstrations Track*, pages 21–24. CEUR-WS, 2013.

(De Vocht et al., 2011) Laurens De Vocht, Selver Softic, Martin Ebner, and Herbert Mühlburger. Semantically driven social data aggregation interfaces for research 2.0. In *Proceedings of the 11th International Conference on Knowledge Management and Knowledge Technologies*, i-KNOW ’11, pages 43:1–43:9, New York, NY, USA, 2011. ACM.

(Dimou et al., 2014) Anastasia Dimou, Laurens De Vocht, Mathias Van Compernolle, Erik Mannens, Peter Mechant, and Rik Van de Walle. A visual workflow to explore the web of data for scholars. In *Proceedings of the Companion Publication of the 23rd International Conference on World Wide Web Companion*, pages 1171–1176, Republic and Canton of Geneva, Switzerland, 2014. International World Wide Web Conferences Steering Committee.

(Ebner et al., 2010) Martin Ebner, Herbert Mühlburger, Sandra Schaffert, Mandy Schiefner, Wolfgang Reinhardt, and Steve Wheeler. Getting granular on twitter: Tweets from a conference and their limited usefulness for non-participants. In *Key Competencies in the Knowledge Society*, volume 324 of *IFIP Advances in Information and Communication Technology*, pages 102–113. Springer, 2010.

(Gong Cheng and Yuzhong Qu, 2009) Gong Cheng and Yuzhong Qu. Searching linked objects with Falcons: Approach, implementation and evaluation. *International Journal on Semantic Web and Information Systems*, 5(3):49–70, 2009.

(Heim et al., 2009) Philipp Heim, Sebastian Hellmann, Jens Lehmann, Steffen Lohmann, and Timo Stegemann. Relfinder: Revealing relationships in rdf knowledge bases. In *Proceedings of the 4th International Conference on Semantic and Digital Media Technologies (SAMT 2009)*, pages 182–187, Berlin, Heidelberg, 2009. Springer.

(Herzig & Tran, 2012) Daniel M. Herzig and Thanh Tran. Heterogeneous web data search using relevance-based on the fly data integration. In *WWW*, pages 141–150. ACM, 2012.

(Landis & Koch, 1977) J Richard Landis and Gary G Koch. The measurement of observer agreement for categorical data. *Biometrics*, pages 159–174, 1977.

(Laniado & Mika, 2010) David Laniado and Peter Mika. Making sense of twitter. In *International Semantic Web Conference (1)*, volume 6496 of *Lecture Notes in Computer Science*, pages 470–485. Springer, 2010.

(Letierce et al., 2010) Julie Letierce, Alexandre Passant, John Breslin, and Stefan Decker. Understanding how twitter is used to widely spread scientific messages. In *Proceedings of the WebSci10: Extending the Frontiers of Society On-Line*, 2010.

(Li et al., 2008) Li, G., Ooi, B., Feng, J., Wang, J., Zhou, L. EASE: An Effective 3-in-1 Keyword Search Method for Unstructured, Semi-structured and Structured Data. In *Proceedings of International Conference on Management 0.4 of Data*. pages 903-914 Principles of Database Systems., 2008.

(Li, 2012) Li, Huiying. An Approach to Semantic Information Retrieval. In *Proceedings of the International Conference on Cloud and Service Computing (CSC)*. Pages 161-167. IEEE, 2012.

(Lohmann et al., 2011) Steffen Lohmann, Paloma Díaz, and Ignacio Aedo. Muto: the modular unified tagging ontology. In *I-SEMANTICS*, pages 95–104. 2011. ACM.

(Marchionini, 2006) Gary Marchionini. Exploratory search: from finding to understanding. *Communications of the ACM*, 49(4):41–46, April 2006.

(Mathioudakis & Koudas, 2010) Michael Mathioudakis and Nick Koudas. Twittermonitor: trend detection over the twitter stream. In *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*, pages 1155–1158, New York, NY, USA, 2010. ACM.

(Page et al., 1999) Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The PageRank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab, November 1999. Previous number = SIDL-WP-1999-0120.

(Parra & Duval, 2010) Gonzalo Parra Chico and Erik Duval. Filling the gaps to know More! about a researcher. In *Proceedings of the 2nd International Workshop on Research 2.0. At* *the 5th European Conference on Technology Enhanced Learning: Sustaining TEL,*, pages 18–22. CEUR-WS, September 2010.

(Pintado, 1995) Xavier Pintado. *Object-oriented software composition*, chapter The affinity browser, pages 245–272. Prentice Hall, 1995.

(Powers et al, 2011) David MW Powers. Evaluation: from precision, recall and f-measure to roc, informedness, markedness & correlation. *Journal of Machine Learning Technologies*, 2(1):37–63, 2011.

(Rheinhardt et al, 2009) Wolfgang Reinhardt, Martin Ebner, Günter Beham, and Cristina Costa. How people are using twitter during conferences. In *Hornung-Prähauser, V., Luckmann, M.(Hg.): 5th EduMedia conference, Salzburg*, pages 145–156, 2009.

(Softic & De Vocht et al., 2015) Selver Softic, Laurens De Vocht, Erik Mannens, Martin Ebner, and Rik Van de Walle. COLINDA: modeling, representing and using scientific events in the web of data. In *Proceedings of the 4th International Workshop on Detection, Representation, and Exploitation of Events in the Semantic Web (DeRiVE 2015)*, volume 1363 of CEUR-WS, pages 12–23, 2015.

(Softic & Ebner et al., 2010) Selver Softic, Martin Ebner, Herbert Mühlburger, Thomas Altmann, and Behnam Taraghi. twitter mining# microblogs using# semantic technologies. *6th Workshop on Semantic Web Applications and Perspectives*, pages 1–9, 2010.

(Tao et al., 2011) Ke Tao, Fabian Abel, Qi Gao, and Geert-Jan Houben. Tums: Twitter-based user modeling service. In *ESWC Workshops*, volume 7117 of *Lecture Notes in Computer Science*, pages 269–283. Springer, 2011.

(Tablan et al., 2014) Tablan, Valentin, Kalina Bontcheva, Ian Roberts, and Hamish Cunningham. Mímir: An open-source semantic search framework for interactive information seeking and discovery. *Web Semantics: Science, Services and Agents on the World Wide Web*. 30: 52-68, 2014.

(Tran, Herzig & Ladwig, 2011) Thanh Tran, Daniel M. Herzig, and Günter Ladwig. Semsearchpro - using semantics throughout the search process. *Web Semantics: Science, Services and Agents on the World Wide Web.*, 9(4):349–364, 2011.

(Tran, Wang & Haase, 2009) Thanh Tran, Haofen Wang, and Peter Haase. Hermes: Dataweb search on a pay-as-you-go integration infrastructure. *Web Semantics: Science, Services and Agents on the World Wide Web*, 7(3), 2009.

(Ullman et al., 2007) Thomas Daniel Ullmann, Fridolin Wild, Peter Scott, Erik Duval, Bram Vandeputte, Gonzalo Alberto Parra Chico, Wolfgang Reinhardt, Nina Heinze, Peter Kraker, Angela Fessl, Stefanie Lindstaedt, Till Nagel, and Denis Gillet. Components of a research 2.0 infrastructure. In *Lecture Notes in Computer Science,*, pages 590–595. Springer, 2010.

(Uren et al., 2010) Victoria Uren, Marta Sabou, Enrico Motta, Miriam Fernandez, Vanessa Lopez, and Yuangui Lei. Reflections on five years of evaluating semantic search systems. *International Journal of Metadata, Semantics and Ontologies (IJMSO)*, 5(2):87–98, 2010.

(Van Noorden, 2014) Van Noorden, R. Online collaboration: Scientists and the social network. *Nature*, 512(7513), pages 126-129. 2014.

(Weller et al., 2011) Katrin Weller, Evelyn Dröge, and Cornelius Puschmann. Citation analysis in twitter: Approaches for defining and measuring information flows within tweets during scientific conferences. In *Making Sense of Microposts (#MSM2011)*, pages 1–12, 2011.