

FuhSen: A Platform for Federated, RDF-based Hybrid Search

Diego Collarana
University of Bonn, Germany
collaran@cs.uni-bonn.de

Christoph Lange
University of Bonn /
Fraunhofer IAIS, Germany
lange@cs.uni-bonn.de

Sören Auer
University of Bonn /
Fraunhofer IAIS, Germany
auer@cs.uni-bonn.de

ABSTRACT

The increasing amount of structured and semi-structured information available on the Web and in distributed information systems, as well as the Web's diversification into different segments such as the Social Web, the Deep Web, or the Dark Web, requires new methods for horizontal search. FuhSen is a federated, RDF-based, hybrid search platform that searches, integrates and summarizes information about entities from distributed heterogeneous information sources using Linked Data. As a use case, we present scenarios where law enforcement institutions search and integrate data spread across these different Web segments to identify cases of organized crime. We present the architecture and implementation of FuhSen and explain the queries that can be addressed with this new approach.

Keywords

Federated Search, Social Web, Deep Web, Linked Data, RDF

1. INTRODUCTION

The more the amount of digital information grows, the more important are efficient and effective querying, exploration, search and retrieval strategies. Much attention has been given to querying, exploration, search, and retrieval for various information modalities. For text, for example, *Information Retrieval* is a long established research field. A vast number of commercial products as well as mature open-source implementations such as *Apache Solr* are now driving large-scale applications. Also, in the areas of databases or the Semantic Web, a number of approaches, techniques and platforms have been developed [9, 6, 2, 5]. However, for many applications, *heterogeneous* information represented in *different modalities* (structured, semi-structured, unstructured) and spread across *distributed* information sources has to be made searchable and explorable for end users in an integrated way. We briefly describe such a distributed search

application scenario. During the process of crime investigation, collecting and analyzing information from different sources is one of the key steps performed by investigators.

Before digitization this process was only performed in the physical world, but today it is being transferred to the digital world. As a consequence, it is now a common practice that investigators access information from the Web, focusing especially on social networks [8]. Similarly, e-commerce platforms represent another relevant source of information during crime investigations, since illegal items (e.g. stolen, counterfeited, prohibited) are frequently offered and acquired on such platforms.

The current process of searching for such items or suspect information is extremely cumbersome and time-consuming because it requires access to a large number of different data sources and manually integrating individual search results. In order to address such use cases, we developed a federated, hybrid search architecture implemented in the FuhSen system, which uses a local as view approach employing vocabularies and semantic mappings for information integration. FuhSen semantically aggregates information from distributed sources, simplifies the search process and adds the power of semantics to extract and link information about entities from a diverse set of information sources.

In particular, during a typical investigation the following sources need to be accessed: (a) *Social networks* potentially containing profiles of suspects, (b) *e-commerce websites* where illegal items are traded, (c) *internal databases* of the law enforcement authorities, and (d) *public knowledge bases* such as lists of politically exposed persons (so called PEP lists) etc.

Each of this information sources is accessible via different search mechanisms. Moreover, it is neither possible nor – at least in many countries (e.g. Germany) with strict data protection and privacy laws – desirable to create a fully integrated dataset comprising the information collected from all these sources.

FuhSen is *federated* because it searches multiple information sources with a single user query request and then aggregates the results before presenting them to the user. It is *RDF-based* because all the heterogeneous data is transformed to a semantic representation and then enriched. Finally, it is *hybrid* because it is powered by different types of data from unstructured, semi-structured to structured. We present an implementation of the FuhSen federated hybrid search platform including a web user interface, where

Google+, Twitter, eBay, Amazon, Google Knowledge Graph and a PEP list are searched to aggregate relevant information about people, organizations, and products. Further information about FuhSen can be found in the repository wiki¹. A comprehensive demo video is also available².

2. ARCHITECTURE

The architecture of the FuhSen platform, shown in Figure 1 (a), comprises three main elements: (1) a faceted browsing UI for navigating the results, (2) an RDF-based federated search and integration engine, and (3) a wrappers layer for accessing and mapping from distributed information sources. The communication between the components is performed through HTTP connections. The main innovation in the architecture is the use of RDF as the core data model for the federated integration of hybrid search. This enables FuhSen to (a) deal effectively with heterogeneity, (b) find relations between the resulting entities, and (c) link the search results with other sources.

Faceted Browsing UI

It is the entry point for the user. A multi-faceted navigation is very useful to explore large-scale data [7]. Figure 2 illustrates the main user interface elements that comprise: a text box for the search query, a result list, entity summaries, and a faceted navigation component. We chose JSON-LD as the messaging format for communication with the backend. This avoids unnecessary data transformation for the UI components, as they use JSON natively.

RDF-based Search Engine

The main component in FuhSen, it is responsible for expanding and enriching the keyword query string, for orchestrating the data extraction through the wrapper components, aggregating the results and applying a ranking algorithm. The FuhSen API exposes the search service via the following URL pattern:

```
http://<base_url>/api/<version>/search?
query=<string>&rows=<int>&offset=<int>
```

We follow a modular component architecture approach so that all components are loosely coupled; this means that each of them may evolve at its own pace. The main process workflow is:

(1) Expanded Search Generation.

It takes the initial query string and produces a list of sequential subsets for this string and assigns the query weight. To ensure a high recall, the query expander produces the power set of the key words entered when the input contains more than 2 elements. For example the search keyword “John Smith Allegro” is expanded to { (“John Smith Allegro”, w=3), (“John Smith”, w=2), (“Smith Allegro”, w=2)}.

(2) Federated Query Execution.

This component takes as its input the expanded list of query strings and coordinates the communication with the different wrapper components. It sends one by one to the wrappers all elements of the expanded query string list.

¹<https://github.com/LiDaKra/FuhSen>

²<https://www.youtube.com/watch?v=gG0TpuImLJQ>

Then, it receives the local results from the wrappers and sends them to vocabulary based aggregation. A configuration element provides flexibility to the search engine at run time. This configuration defines the list of wrappers to be queried and has the following elements: wrapper name, wrapper URL address, and secure API key to connect to the wrapped API. Based on the configuration, the Federated Query Execution communicates to all wrapper components in parallel.

(3) Wrappers layer.

This layer is responsible for extracting data from the different data sources defined in the platform. A specific collection of wrappers is created for each type of data source. For example, for social networks the platform comprises wrappers for Twitter and Google+. The same principle applies to RDF wrappers and database wrappers, that is, when necessary, another wrapper is implemented to handle specific issues of a specific data source. Linked Data wrappers have been used as an alternative to integrating data using a materialized RDF graph [3, 1]. We use wrappers as a data extraction layer for the hybrid search engine. Thanks to the uniform invocation interface and representation of results, the implementation of every wrapper may evolve internally at its own pace. This is especially important in the case of API wrappers, whose underlying APIs also evolve frequently, as described in [1]. The wrappers are designed as a service layer, so a web API is defined for the wrappers. The API exposes the search service via the following URL pattern:

```
http://<base_url>/ldw/<datasource_name>/search?
query=<string>&rows=<int>&offset=<int>
```

The main components of a wrapper are shown in Figure 1 (b). The *Controller* is responsible for handling requests. It transforms the parameters into elements that the wrapper can handle. The controller creates responses depending on the request of the client. The *Query Builder* is responsible for transforming the list of query strings into queries that the data source understands. In the case of an RDF Wrapper the query strings are expressed as SPARQL queries. In the case of an API Wrapper the query strings are expressed, for example, as REST requests. In the case of internal (relational) data the query strings are transformed into SQL queries. The *Information Extractor* is responsible for connecting to the data source. It is responsible for negotiating security and handles the limitations and constraints of the specific data source (e.g. the number of calls per minute or the number of results per call). Finally, the *RDF Translator* takes the results from the information extractor and creates a local RDF graph holding the results. For example, in the case of an API wrapper it transforms JSON results to an RDF graph expressed in the OntoFuhSen vocabulary (see below).

(4) Vocabulary-based Aggregation.

This component is responsible for aggregating all local results produced by the wrappers and creates the final RDF containing the results. It works in close synchronization with the Federated Query Execution. We use a Global Schema approach[3] to aggregate the results. The *OntoFuhSen* vocabulary is used as a common language between the federated search engine and wrappers. The rationale of the

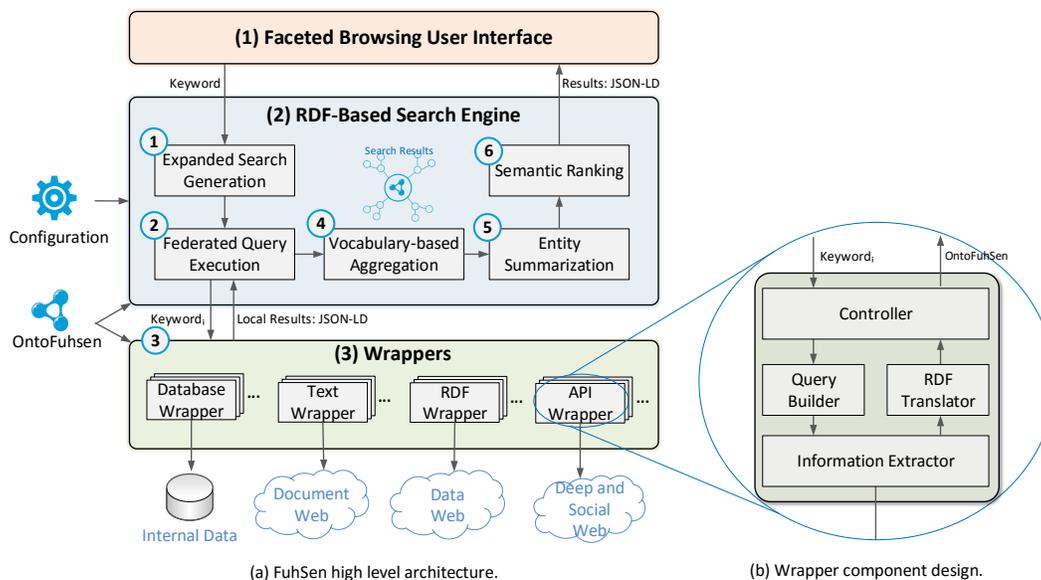


Figure 1: High level architecture comprised of a faceted browsing interface, an RDF-based search engine and wrappers for information sources.

vocabulary is threefold: (1) as a response format for transmitting its local results to the engine, (2) as a core data model to apply semantic search algorithms, and (3) to support the display of the results and facets. The vocabulary-based aggregation component creates an in-memory *RDF Results Graph*. The vocabulary-based approach keeps the aggregator’s task relatively simple. It adds all the RDF local results from the wrappers to the RDF results graph and adds provenance information according to the wrapper’s metadata. *Reuse* is considered a best practice in vocabulary engineering [4]. Hence, we do not create new terms to describe the entities that FuhSen can find, but utilize those that are present in existing vocabularies. We utilized terms from well-known ontologies such as *FOAF*³ for person, *GoodRelations*⁴ for products, the *Organization Ontology*⁵ for organizations, and *PROV*⁶ for provenance.

(5) Entity Summarization.

This component generates a summary of each entity among the search results; it adds triples that contain an image and a human understandable textual description of the entity. This entity summary is displayed in the faceted browsing user interface. For each type of entity (person, organization, product), the configuration specifies the properties that should be used to summarize an entity.

(6) Semantic Ranking.

This component adds a triple with ranking information to all results. The rank is calculated from a normalization of three elements: (a) the query weight, (b) the order given by the wrapper, and (c) the amount of entity information found in the aggregation process.

3. USE CASE AND PROTOTYPE

The Web not only gives rise to new forms of crime, it also enables new technology for crime investigation. Suspects leave traces on the Web, items are being sold and bought on the Web, and a wealth of public open data about organizations and places is available on the Web. One of the goals of FuhSen is to help law enforcement organizations to find such traces. Based on the architecture, we have developed a prototype of a platform supporting criminal analysis based on the FuhSen federated search engine, including a web user interface. This prototype currently searches for information about persons, organizations and products across *Google+*, *Twitter*, *eBay*, *Amazon*, *Google Knowledge Graph* and *PEP lists*. We have developed instances of the API Wrapper component to query *Google+*, *Twitter*, *Google Knowledge Graph* and *eBay*, which use their respective APIs⁷. We harmonize the local results to the global schema using the *Silk* integration framework, which supports data transformation tasks, in our case from JSON to the *OntoFuhSen* vocabulary. To query the *PEP list*, we have developed an instance of the *RDF Wrapper* component, which basically translates the given query string into a *SPARQL* query against a *Fuseki server*⁸, i.e. a *SPARQL* endpoint for a static *RDF* file.

Attendees of the demo session will be able to see FuhSen in action, understand the challenges posed by such a horizontal search, and realize how FuhSen helps to face and overcome such challenges. The demonstration will focus on 3 use cases:

Use Case 1: Search for a person profile. The objective is to find the social networks and other web services in which a person has an account.

Use Case 2: Identify the location of a specific organization. The objective is to find out address, telephone number,

³<http://xmlns.com/foaf/spec/>

⁴<http://purl.org/goodrelations/v1>

⁵<http://www.w3.org/ns/org#>

⁶<http://www.w3.org/ns/prov#>

⁷<https://developers.google.com/+web/api/rest/>,
<https://go.developer.ebay.com/>

⁸http://jena.apache.org/documentation/serving_data/

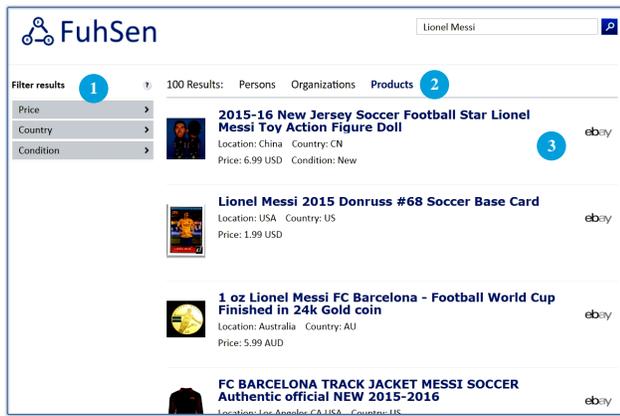


Figure 2: FuhSen – results screen: 1. Filters or Facets. 2. Category navigation. 3. Result list.

email address, or bank account of a specific organization. It is possible to find such information in internal databases of the law enforcement agencies, on social networks, or even in the Document Web.

Use Case 3: *Where is a certain product being offered?* The objective is to find out in which e-commerce platforms a certain product is being offered at what price.

The web user interface follows a simple search box style where a user can enter search criteria, which could be the name of a person, an organization, or a product. Figure 2 shows how results are displayed to the user grouped by entity type, where the entity summary includes an image and essential properties of the entity. The results are sorted according to the rank value assigned during the search process. Finally, Figure 3 illustrates how facets can be applied for filtering the results. Facets' values are generated automatically from the aggregated results and sorted by frequency. The prototype has been demonstrated to domain experts, where its practical relevance and its general usability have been validated. The platform-as-a-service design opens the possibility to provide *semantic search agents* that automatically search for information, analyze the results, and alert the police authorities when a result is found.

4. CONCLUSIONS

In this paper we demonstrated the foundation for a novel federated, RDF-based hybrid search platform, starting from the design of an architecture, to a comprehensive prototype implementation. The federated, vocabulary-based hybrid search concept constitutes a novel architectural pattern incorporating elements from universal search, semantic integration as well as multi-modal search and retrieval. The aggregation of information from Social Web, Deep Web, Data Web and internal databases represents a novel approach to summarize relevant information about crime investigation.

Although we initially focus on use cases in the criminal investigation domain, we deem that there are numerous further use cases, e.g., related to e-commerce (e.g. price comparison) or social media. Consequently, FuhSen is designed in such a generic, modular and flexible way that it can be adapted easily to other scenarios beyond crime

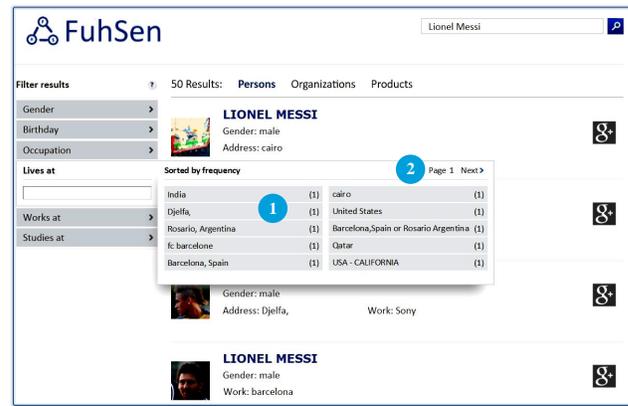


Figure 3: FuhSen – facets selection: 1. Facets values sorted by frequency. 2. Navigation functionality for the facets values.

investigation. Through its clear interface definitions, the flexible vocabulary-based integration models and the modular architecture, new information sources can be plugged-in with minimal effort and the platform can be tailored easily towards new application domains. When applied more widely, this federated search approach can contribute to realizing novel applications and business models, previously prevented by the prohibitive cost of a full data integration.

Acknowledgments. This work was funded by the German Ministry of Education and Research grant no. 13N13627.

References

- [1] J. Iturrioz, I. Azpeitia, and O. Díaz. “YQL as a Platform for Linked-Data Wrapper Development”. In: *Engineering the Web in the Big Data Era*. Springer, 2015.
- [2] V. Lopez et al. “PowerAqua: Supporting users in querying and exploring the semantic web”. In: *Semantic Web 3.3 (2011)*, pp. 249–265.
- [3] G. Montoya et al. “Semlav: Local-as-view mediation for SPARQL queries”. In: *Transactions on Large-Scale Data-and Knowledge-Centered Systems XIII*. Springer, 2014.
- [4] C. Pedrinaci, J. Cardoso, and T. Leidig. “Linked USDL: a vocabulary for web-scale service trading”. In: *The Semantic Web: Trends and Challenges*. Springer, 2014.
- [5] N. Schlaefler et al. “Semantic Extensions of the Ephyra QA System for TREC 2007”. In: *TREC. 2007*.
- [6] S. Shekarpour et al. “Sina: Semantic interpretation of user queries for question answering on interlinked data”. In: *J. of Web Semantics 30 (2015)*, pp. 39–51.
- [7] G. Simonini and S. Zhu. “Big data exploration with faceted browsing”. In: *Int. Conf. on High Performance Computing & Simulation (HPCS)*. IEEE, 2015.
- [8] *Social Media Use in Law Enforcement: Crime prevention and investigative activities continue to drive usage*. Tech. rep. LexisNexis Risk Solutions., 2014.
- [9] R. Usbeck et al. “HAWK – Hybrid Question Answering over Linked Data”. In: *ESWC*. Springer, 2015.