Exploratory Browsing in the Web of Data

Roberto Mirizzi
Politecnico di Bari – Via Orabona, 4, 70125 Bari, Italy
mirizzi@deemail.poliba.it

1 Research Direction

Thanks to the recent Linked Data initiative, the foundations of the Semantic Web have been built [2]. Shared, open and linked RDF datasets give us the possibility to exploit both the strong theoretical results and the robust technologies and tools developed since the seminal paper in the Semantic Web [1] appeared in 2001. In a simplistic way, we may think at the Semantic Web as a ultra large distributed database we can query to get information coming from different sources. In fact, every dataset exposes a SPARQL endpoint to make the data accessible through exact queries. If we know the URI of the famous actress Nicole Kidman in DBpedia [3] we may retrieve all the movies she acted with a simple SPARQL query. Eventually we may aggregate this information with users ratings and genres from IMDB. Even though these are very exciting results and applications, there is much more behind the curtains. Datasets come with the description of their schema structured in an ontological way. Resources refer to classes which are in turn organized in well structured and rich ontologies. Exploiting also this further feature we go beyond the notion of a distributed database and we can refer to the Semantic Web as a distributed knowledge base. If in our knowledge base we have that Paris is located in France (ontological level) and that Moulin Rouge! is set in Paris (data level) we may query the Semantic Web (interpreted as a set of interconnected datasets and related ontologies) to return all the movies starred by Nicole Kidman set in France and Moulin Rouge! will be in the final result set. The ontological level makes possible to infer new relations among data.

The Linked Data initiative and the state of the art in semantic technologies led off all brand new search and mash-up applications. The basic idea is to have smarter lookup services for a huge, distributed and social knowledge base. All these applications catch and (re)propose, under a semantic data perspective, the view of the classical Web as a distributed collection of documents to retrieve. The interlinked nature of the Web, and consequently of the Semantic Web, is exploited (just) to collect and aggregate data coming from different sources. Of course, this is a big step forward in search and Web technologies, but if we limit our investigation to retrieval tasks, we miss another important feature of the current Web: browsing and in particular exploratory browsing (a.k.a. exploratory search) [7]. Thanks to its hyperlinked nature, the Web defined a new way of browsing documents and knowledge: selection by lookup, navigation and trial-and-error tactics [8] were, and still are, exploited by users to search for relevant information satisfying some initial requirements. The basic assumptions behind a lookup search,
typical of Information Retrieval (IR) systems, are no more valid in an exploratory browsing context. An IR system, such as a search engine, assumes that: the user has a clear picture of what she is looking for; she knows the terminology of the specific knowledge space. On the other side, as argued in [13], the main challenges in exploratory search can be summarized as: support querying and rapid query refinement; offer facets and metadata-based result filtering; leverage search context; support learning and understanding; offer visualization to support insight/decision making; facilitate collaboration. In Section 3 we will show two applications for exploratory search in the Semantic Web addressing some of the above challenges.

2 PhD in the context of the Semantic Web

The graph-based structure of datasets in the Semantic Web allows a natural visualization and browsing of the formalized information simplifying the implementation of learning and investigating strategies for knowledge acquisition and discovery. Using an Orienteering [12] behavior, a user starts from an initial vague idea of what she is looking for and navigates through the information space.

Effective user interfaces play a crucial role in order to provide a satisfactory user experience during an exploratory search. There are two main trends in visualizing and navigating RDF datasets [4]: via browsing a labeled oriented graph or displaying RDF properties as browsable facets of a node. The main issue of both approaches is to filter information which is not relevant for the exploratory task. If we visualize all the triples together with their connections we have a very huge unreadable representation of the underlying knowledge [11]. In fact, RDF triples are conceived to represent information for machine-to-machine interaction. A lot of information triplified in datasets is very useful in many automated computational tasks (e.g., web service interaction) but it is completely useless from a human perspective during an exploratory search [5].

3 The work done so far

In this section we describe two systems we developed for exploratory search in the Semantic Web. Currently, both systems rely on one of the most relevant and biggest datasets in the Linked Data cloud: DBpedia [3]. The first tool SWOC (Semantic Wonder Cloud) has been inspired by Google Wonder Wheel3 (GWW). SWOC main aim is to reproduce the same functionalities of GWW in a Semantic Web setting. The approach to represent and navigate through DBpedia semantic metadata is graph-based. The second tool MORE (Movies Recommendation) exploits exploratory search as the core of a movies recommender system. MORE (developed as a Facebook application) uses a faceted browsing approach to metadata navigation and exploration.

It is worth noticing that in both cases the complex semantic nature of the underlying metadata is completely hidden to the end user. The user is completely unaware of the information sources. The main goals we had in mind before developing our applications were: (1) intuitive and usable for end-users, (2) rich in

3 http://www.googlewonderwheel.com/
SWOC: Semantic Wonder Cloud. SWOC is a tool that helps users in exploratory knowledge search in DBpedia. The front-end of the system is available at http://sisinflab.poliba.it/semantic-wonder-cloud/index/. SWOC consists of two main subsystems. The first subsystem, the SWOC explorer, is a GUI designed to allow exploration of DBpedia. The second subsystem – DBpediaRanker – is the back-end of the whole system whose main task is to compute similarity between pairs of DBpedia nodes. In order to compute a similarity value between two resources res_1 and res_2, DBpediaRanker performs three main tasks (refer to [9] for more details): (i) analysis of the underlying RDF graph structure; (ii) textual analysis on the value of the dbpedia-owl:abstract property; (iii) exploitation of external information sources, such as search engines and social tagging systems. In our current implementation we use Google, Yahoo!, Bing and Delicious. We stress that, differently from common RDF visualization tools, SWOC does not visualize the DBpedia RDF graph as it is (see Fig. 1). There are some key differences w.r.t. other RDF explorers. The graph displayed by SWOC represents possible new associations between resources computed by mixing the semantic knowledge formalized in the DBpedia dataset and the non-semantic statistical knowledge available from web search engines and social tagging systems. Hence, the navigation is driven by knowledge associations between resources rather than guided by the structure of the underlying dataset. Displayed associations may even not exist in the original DBpedia graph. Each node in the displayed graph has a different size representing how relevant it is with respect to the resource represented in the center of the graph itself. There are two main classes of nodes in SWOC: one representing DBpedia categories, the other representing instances. This reflects in some way the different nature of the knowledge they represent. Categories are used to group and classify resources. Hence, if the user clicks on a category node then she sees the most relevant (popular) instances of that category together with its most relevant categories. SWOC has been designed to be a user centered explorer of RDF knowledge bases rather than a common RDF graphic visualizer. Actually, SWOC can not be considered an RDF visualizer in the classical sense. It is a tool useful to help end-users in exploratory knowledge search. Usually end-users do not care about obtaining a sketch of the whole result set (think to results returned by a search engine: rarely you would go beyond the second page), but are more interested in finding something relevant among the first results. Thanks to SWOC, end-users can navigate from a resource to another one exploring and discovering new knowledge. Following the exploratory search paradigm, SWOC differentiates also from traditional search engines. The latter allow to find what you are looking for and already know, SWOC allows to explore what you probably did not know.

MORE: Movie Recommendation. Differently from SWOC, in order to rank pairs of resources the approach adopted for MORE relies exclusively on semantic metadata (in this case we concentrate on movies domain). We extracted the whole DBpedia subgraph related to movies through SPARQL and then we post-process the graph off-line for performance reasons. We decided to develop MORE as a Facebook application (see Fig. 2) because we believe that this social network is one of the best place where to promote quite easily a new initiative and to allow a big part
of Internet users (Facebook has more than 400 million active users) to benefit of the Semantic Web innovations. The application is available at http://apps.facebook.com/film-recommendation/.

MORE has been designed to be multilingual exploiting the multilingual nature of the DBpedia dataset.

The interface is very intuitive and simple. After the application is loaded, the simplest action a user can do is to search for a movie by keying in some characters in the corresponding text field. The system returns a list of movies, ranked by popularity, that refers to what the user has searched for. To rank the movies in this list we applied a variant of the PageRank algorithm [10] to the DBpedia subgraph related to movies. After the system has returned the list, the
user can select one of the suggested movies from it. Then, the system inserts the chosen film in the area of the favourite films by the user and recommend a ranked list of the top-40 movies, related to the selected one. If the user moves the pointer over the image related to a movie in the area of the selected films, a list of icons fades in. Clicking on the first icon on the left, it is possible to obtain information about the selected movie, e.g., starring, director, subject, etc. Each of these facets corresponds to a DBpedia property associated with the film. Each facet can be used for an eventual exploratory search starting from the selected movie. The other icons retrieve information related to the selected movie from Wikipedia, YouTube, Facebook, Amazon, IMDB, RottenTomatoes and Netflix. We are currently planning to link some of these resources external to DBpedia by publishing RDF-related triples in a publicly available dataset.

For each recommended movie, MORE provides two explanations: (i) the one representing the common characteristics between the movie for which you require explanation and all the movies in the favorite area, (ii) the other consisting of an explanation for each preferred film.

MORE allows the user to perform an exploratory search starting from each recommended movie. In fact, each property value is a link that can be further explored to have more information about it. For example, if you had selected the movie Moulin Rouge!, and were looking at the characteristics of this film, you would find Nicole Kidman among the actors. If you were interested in the movies where Nicole Kidman starred, you may click on the link, to obtain a new list containing all the movies related to the selected property value. The list can be ordered either by name or by popularity. As we said before, the popularity of a movie is computed via a PageRank-style algorithm, i.e., a greater number of backlinks to a movie corresponds to a higher popularity. From an exploratory search perspective, MORE can be used for faceted browsing in the movies domain. As an example, suppose you are coming back home after a trip to San Francisco and you were fascinated by the city. As movie fan, the first thing you do when you get home, is to look on the Internet for the films set in San Francisco to rent one of them. With MORE, the operation is very simple. Once connected to the application, instead of searching straight for a movie, you can switch the facet to Subject (as shown in Fig. 3) and start to type the string San Francisco. The system will return a ranked list of subjects about San Francisco, like Films set in San Francisco, California or Films set in the San Francisco Bay Area. You can explore the films belonging to a given subject, and rent the one you prefer on Netflix directly through MORE. This way you can search for a movie through any possible facets depicted in Fig. 3. Differently from a mere search on Wikipedia and differently from Faceted Wikipedia Search [6], MORE really supports you in browsing a huge amount of information by exploring it in an innovative and simple way.

4 Future work

In the near future we plan to extend our approaches and tools to cover, whenever possible, as many linked RDF datasets as possible. We also want to perform a systematic evaluation of our approaches with respect to the challenges introduced at the end of Section 1. By these analysis we aim at understanding how important
is the role played both by the user interface and by the “quality” of the underlying semantic data. We also will investigate the relations existing between recommender systems and exploratory search. As shown by MORE, we argue that exploratory search can be used as an explanation on results provided by (social and semantic) recommender systems.

References