published as: Waitelonis, J. & Sack, H. Towards Exploratory Video Search Using Linked Data. In Proceedings of the 2nd IEEE International Workshop on Data Semantics for Multimedia Systems and Applications (DSMSA2009), in conjunction with IEEE International Symposium on Multimedia (ISM2009), 14-16 December, 2009, San Diego, California, USA, IEEE, 2009, pp. 540-545. (ISBN 978-0-7695-3890-7)

2009 11th IEEE International Symposium on Multimedia

Towards Exploratory Video Search Using Linked Data

Jörg Waitelonis, Harald Sack
Hasso-Plattner-Institute Potsdam
Prof.-Dr.-Helmert-Str. 2-3
14482 Potsdam, Germany
{*joerg.waitelonis*|*harald.sack*}*@hpi.uni-potsdam.de*

***Abstract*—Keyword-based search in general is particularly applicable if the searcher really knows what she is looking for and how to find it. But in many cases either the objectives of the searcher are intrinsically fuzzy or she has no idea of the appropriate keywords. One way to solve this problem is to navigate and explore the search space along a guided route. In this paper we show, how Linked Open Data can be adopted to facilitate an exploratory semantic search for video data. We present a prototype implementation of exploratory video search and give first results that show how traditional keyword-based search can be augmented by the use of Linked Open Data.**

I. Introduction

The search for information, no matter whether you consider archives, libraries, or the World Wide Web (WWW), strictly speaking turns out to be a win-win-situation for both the information consumer as well as for the information provider. The information consumer is looking for information that the information provider supplies, while the information provider wants the consumer to find and to select his information offer. But, how can they meet?

WWW search engines as well as the subject heading catalogue of the library indicate the way to the information seeker. While the subject heading catalogue is arranged manually – suitable keywords or keyword chains are assigned to information resources –, sophisticated algorithms are used to generate keywords automatically from (textual) information resources in the WWW. But, how to assign suitable keywords remains expert knowledge, i.e. the ordinary user hardly knows anything about which keywords are required to actually find a specific resource. Even worse, in the WWW the user never can be sure about the completeness and the integrity of the achieved search results.

Part of the responsibility for that situation bears the traditional keyword-based search paradigm. You have to know the right keyword to find a specific ressource. That's all. But, what if the prospected resource hides several hundred pages later in the list of returned results? Traditional keyword-based search does not consider the meaning (semantics) of the content of the underlying information.

On the other hand, semantic search promises to enhance keyword-based search by taking into account the actual content of the information and its semantics. By semantic annotation information resources can be related with each other, hidden and implicitly existing relationships can be made explicit. Instead of turning a small keyword spotlight towards our information universe, we can make use of all the properties of its information resources and their relationships among each other to enable the guided exploration of the search space as well as the possibility for serendipitous discovery.

In recent years, especially audiovisual media have become the predominant media of the internet. To enable content based video retrieval, high quality textual metadata have to be provided. Most times, sufficient quality can only be achieved by time- and cost-intensive manual annotation, while collaborative approaches deploy non-autoritative user-generated metadata, and automated video analysis is achieving progress. But, even though sufficient metadata can be provided, explorative investigations will be limited by the paradigm of keyword-based search.

In this paper, we address the problem of how to deploy explorative (semantic) search for video data. The Linked Open Data (LOD) [1] project aims at making semantic data freely available to everyone and provides starting points to extract relationships among information resources. We show how to use the LOD resources DBPedia [2] and 'Wortschatz Leipzig' (an automatically compiled thesaurus for various languages) [3] to implement an exploratory search for our video search engine yovisto.com[1]. Starting with a simple keyword-based query, relationships between information instances within yovisto's database are discovered by mapping terms with LOD resources and by utilizing their ontological structure. Thus, the user has not only access to keyword-based search results, but will also be guided by content-based associations to enable serendipitous discovery.

The paper is organized as follows. Section 2 presents related work and introduces to exploratory search, the Yovisto video search portal, and the LOD project. Section 3 details, how Linked Data can facilitate exploratory search. In Section 4, first results and a brief evaluation of how Linked Data resources complement our original video data is presented. The last section concludes the paper with a brief outlook on future work.

[1]Yovisto - Academinc Video Search: http://www.yovisto.com/

978-0-7695-3890-7/09 $26.00 © 2009 IEEE
DOI 10.1109/ISM.2009.111

II. Exploratory Search – Foundations and Related Work

This section introduces prerequisites to implement exploratory search for the video search engine Yovisto.com by using LOD. The concept of exploratory search and the Yovisto search engine and its mapping to the semantic web are explained.

A. Exploratory Search

In contrast to traditional keyword-based search, exploratory search assists the user in exploring the data space to improve search experience. Thereby, the user is able to navigate the search space, as well as to reorganize the content and the user interfaces for her own needs with appropriate interactive elements. While searching, the user is able to choose between alternatives, move along paths, and move back to choose an alternative way. To implement explorative search, the underlying data needs to be fully made accessible. Relationships between associated resources have to be made explicit to let the user navigate along them. Typically, there are different kinds of relationships, e. g. resources belonging to the same category, authored by the same person, etc. One way to establish a simple exploratory search is to reorganize and to filter the search results according to these relationships by so-called faceted search [4].

For example, Schreafel et. al. developed mSpace, a multi-column faceted spatial browser for multimedia data [5]. Petratos described facets as conceptual categories, which are created to organize the presentation of all the available data into an easy to view concise set of conceptual groups [6]. Furthermore, explorative search also means to discover new associations and new kinds of knowledge.

Marchionini differentiates between lookup, learn and investigation search [7]. Besides lookup search, which is the most basic type of search, investigation search involves multiple iterations, and it takes place over a long time period to create knowledge bases from the critically assessed information of the returned results. Marchionini considers learn- and investigation search to be exploratory search. Active user involvement in the search process and uncovering new connections between resources is an essential characteristic of exploratory search. This implies new search interfaces with exploratory navigational components to be able to delve deeper into a repository than before.

Exploratory search can be applied to any type of document. Especially time-dependent multimedia such as video facilitates the visualization of different views on the media. For example, Christel discusses video storyboards as exploratory interfaces and how to move beyond fact-finding by investigating multiple views and visual exposition of metadata and multimedia surrogates [8]. Basically, storyboards give an overview of the visual characteristics of a video. But, for visually homogeneous video data, as e. g., in the recording of a conference talk or a lecture, it is difficult to deduce the content from its visual features only.

To enable keyword-based search in general, video search engines require content-related metadata which can be generated automatically and also manually by the user.

B. yovisto.com

Yovisto is a video search engine specialized in academic lecture recordings and conference talks. Unlike other video search engines, Yovisto provides a time based video index, which allows to search within the videos. Yovisto's index is built up from fine-granular time-dependent metadata. Automated analysis such as scene detection and intelligent character recognition (ICR) are used for metadata generation [9]. In addition, time dependent collaborative annotation enables the user to annotate tags and comments at any point within a video [10].

Yovisto's metadata is encoded in the standardized and interchangeable metadata description framework MPEG-7 to ensure interoperability [11]. Currently, Yovisto provides more than 5.500 videos (5.800 hours) with 1.5 million keywords and 20.000 user generated annotations.

To facilitate a suitable application programming interface (API) for mashup web applications, Yovisto's metadata is published in RDF [12] format, being embedded as RDFa in the webpages and also accessible via a RDF triple-store[2]. By publishing RDF data, Yovisto supports the Linked Open Data initiative, which is discussed in the following section.

C. Linked Open Data

The aim of the Linking Open Data (LOD) project is to identify datasets that are available under open licenses, republish these in RDF on the Web and interlink them with each other [13]. Interlinking resources across various data sources leads to a huge network of data, referred to as the LOD cloud, currently consisting out of more than 4.7 billion RDF triples interlinked by 142 million RDF links (May 2009) [1].

One of the key interlinking hubs of the LOD cloud is DBpedia, the semantic counterpart of the online encyclopedia Wikipedia. DBpedia generates RDF-triples from Wikipedia infoboxes and publishes them via SPARQL [14] and RDF dump files [13]. Together with Wortschatz Leipzig [3], DBpedia serves as the main source for interlinking Yovisto's metadata to put exploratory search in practice. Wortschatz Leipzig collects large volumes of natural language text from the WWW and applies sophisticated linguistic and statistic analysis in large scale to provide thesaurus information. Wortschatz Leipzig supports more than 17 languages and is also publicly available as RDF Linked Data[3].

[2] http://sparql.yovisto.com/

[3] http://corpora.uni-leipzig.de/rdf/sparql/

Following the Linked Data principles Yovisto data is mapped to the LOD cloud [15]. To achieve this, an OWL-DL ontology has been defined to represent the Yovisto data structure[4]. One important issue was to make sure to reuse already existing ontologies to enable interoperability (DublinCore [16], FOAF [17], tag-ontology [18], MPEG-7 Ontology for the MPEG-7 XMLSchemas [19]) while extending or restricting them to meet our needs.

Instances of organizations, categories and persons are mapped via `owl:sameAs` and `rdfs:seeAlso` to DBpedia and via `rdfs:seeAlso` to Wortschatz Leipzig.

How Linked Data is used to uncover implicit relationships among Yovisto resources and to facilitate exploratory search is described in the following section.

III. Using Linked Data to Enable Exploratory Search

Exploratory search aims to find results, which are not considered to be related at a first glance. E. g., if the user enters the query string `hemingway`, resources about the famous american writer `Ernest Hemingway` may be expected result. But, what if there are no resources about Ernest Hemingway? Maybe the user is also interested in similar authors. It could also be useful to suggest resources about the american writer `Ezra Pound`, who was influenced in his work by Ernest Hemingway. This fact cannot be deduced from Yovisto data only. Linked Data from DBpedia can be utilized to obtain this information.

To achieve this, the Yovisto search index has to be mapped to appropriate DBpedia resources. Relevant properties and related resources for every mapped resource in DBpedia needs to be identified. The entire process is three-step:

1) **Matching with DBpedia.**. For every index keyword, user tag and query string, find matching resources in DBpedia while taking into account ambiguities and redirections.
2) **Relevant properties and resources for association.** For every resource found in step (1), find other related resources and meaningful properties.
3) **Alignment with Yovisto.** For every related resource found in step (2), look up matching Yovisto data.

Step 1 - 3 are now explained in more detail.

A. Matching terms with DBpedia

There are different possibilities to match Yovisto terms with DBpedia resources, as e. g., to match resource URIs or to match `rdf:label` literals.

First, Yovisto terms are transformed into the DBpedia resource schema (first letters as capitals and using underscore for compound words, as e. g., "ernest hemingway" will become "Ernest_Hemingway"). If the DBpedia URI cannot be retrieved, the Yovisto term is matched with DBpedia's `rdfs:label`. Almost all DBpedia resources provide an `rdf:label`, which denominates the resource in natural language. If there is no match, the Yovisto term is extended with significant cooccurences from Wortschatz Leipzig. For example: The term "a340" is mapped to a DBpedia resource `dbpedia:A340`. There is a redirection to the resource `dbpedia:A340_(disambiguation)`, but no connection to the expected resource `dbpedia:Airbus_A340`. By supplementing the term with the left neighbor cooccurence "Airbus" from Wortschatz Leipzig a matching with `dbpedia:Airbus_A340` succeeds. If the cooccurence matching fails too, all resources with labels containing the term as a substring are matched.

There is not always a unique match. Because of polysemy the matching of Yovisto terms with DBpedia resources can be ambiguous. For our implementation all relevant matching resources are taken into account.

Once all the resources matching with the term are identified, relevant properties and related resources have to be determined.

B. Find relevant resources and properties for associations

DBpedia resources are part of RDF triples (subject, property, object) and usually occur as subject or object within the RDF Triple. For a comprehensive visual presentation we are also interested in displaying the connecting properties. For example, if `Airbus_A340` is an associated with `Boing_777`, we want to display the connecting property `dbpedia:similarAircraft`.

DBpedia provides numerous properties for every resource and different properties can also be connected with different DBpedia resources. To display all the relations a resource is involved with would not be feasible (because of restrictions to the layout and not to overextend the user's perception). Hence, it is necessary to determine, which properties are the most important. Likewise, a DBpedia resource (subject) is connected to numerous other resources (objects), but not all do really matter. Therefore, the task is to determine, which objects are semantically closer related to a resource than others.

We now discuss the heuristics, which are used to identify important resources and properties. Thereby, the chosen ordering represents the final ranking of the related resources.

1. Properties based on same rdf:type: Starting off with the idea to consider resources of the same category relevant to each other, properties connecting resources of the same `rdf:type` are considered to be important, because they are semantically very closely related. The same holds for the resources being connected by these properties. To detect important properties, all connected resources (objects) of the same category have to be verified against interlinked instances. Fig. 1 illustrates the following example: Albert Einstein and Alfred Kleiner are

[4]Yovisto ontology: http://www.yovisto.com/ontology/0.9/

both of the type dbpedia:Scientists. Albert Einstein is as well a scientist as also an american vegetarian. Bill Cosby is an american vegetarian, too. The property dbpedia:doctoralAdvisor is identified as relevant, because it connects both instances of the category dbpedia:Scientists. In contrast, the other american vegetarian (Bill Cosby) is not tightly coupled to Albert Einstein, because there are no properties connecting both directly. In spite of Albert Einstein, Alfred Kleiner, and Bill Cosby are also persons (i.e. they are of the rdf:type dbpedia:Person, not depicted), Albert Einstein and Alfred Kleiner are more semantically related than Albert Einstein and Bill Cosby, which is to be achieved.

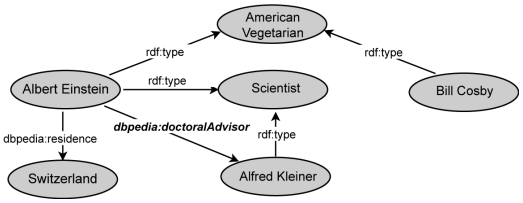

Figure 1. Property between classes of same rdf:type.

2. Properties to pre-defined types: Selected DBpedia categories are considered to be of general importance for our application. Among these are, as e. g., dbpedia:Person, dbpedia:Place, and dbpedia:Event. Yovisto is specialized on academic content, which always comprises information from and about persons as well as places or events. Furthermore, most things of interest for the arbitrary searcher are related to persons, locations, or event. Properties directing to instances of the special classes are pursued and presented to the user.

3. Inverse properties: Resources that are involved in relations, which have an opposite directed connection are considered to be important, because there is evidence that both resources have similar characteristics. For example, Fig. 2 depicts Albert Einstein and Alfred Kleiner. Each one is connected to the other with a different property. Both properties dbpedia:doctoralAdvisor and dbpedia:doctoralStudent are connecting the resources in opposite directions and therefore we deduce evidence for a closer relationship.

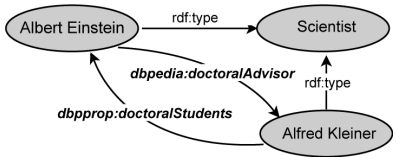

Figure 2. Inverse Properties.

4. Disambiguations: RDF-objects linked to the property dbpedia:disambiguates are considered to be also relevant. Disambiguation links are indicating polysemes, which extend the search query and suggest alternatives to the user to search for.

5. Cooccurences: Cooccurence resources are determined from Wortschatz Leipzig during the matching. Only cooccurences of high significance are considered to be important and will be associated with the original resource.

6. Backlinks: The property dbpedia:wikilink[5] represents a link between two articles in Wikipedia. If Wikipedia article <A> contains a link to article <B>, there will be a triple **<A>** dbpedia:wikilink **<B>**. Objects, which have a bidirectional wikilink are considered to be of higher relevance and closer related to the subject. It is assumed that Resources connected with bidirectional wikilinks are highly interrelated (c. f. Fig. 3).

Figure 3. Bidirectional wikilinks.

7. Unidirectional wikilinks: Similar to bidirectional wikilinks, unidirectional wikilinks are indicating a semantic interrelation. But this relationship is considered to be weaker than bidirectional wikilinks.

8. Part of label: Resources that are found during the substring matching are also taken into consideration. But, because these results might be very inaccurate, this relationship is considered to be the weakest among the others.

C. Alignment with Search Index

As a result of the process described in the previous section, a set of related resources and properties is determined. But, up to now it is not known, whether these newly found resources are also covered by some video of the Yovisto search engine at all. Therefore, all related resources are looked up in the Yovisto search index. If the index search returns no results, the resource will be discarded. If the provenance of a resource is a disambiguation link, a boolean search will be performed on the resource name and the disambiguation addendum, e. g., http://dbpedia.org/resource/David_Bates_(physicist) will result in a search for "David Bates" AND physicist.

D. User Interface

In the next step, a simple exploratory user interface is designed, which presents all the information and proposes alternatives extending the current search query. Fig. 4 shows two different implementations of a vertical and a horizontal

[5]Currently only supported by DBpedia dump files, and not by the SPARQL endpoint.

widget for exploratory search. The horizontal widget at the top of the figure displays the currently active resource name (1) originating from the search query. All related resources with important and meaningful properties are displayed first (2). At the bottom of the first widget other related resources without any meaningful property can be found (3).

The vertically oriented widget on the bottom part of Fig. 4 presents the currently active resource name in the center (1) and arranges the other information in a top down manner. On top, a history list (4) shows the search terms of previous searches. With one click on a history item the user can turn back to a previous search, to reconsider its decision and follow alternative ways. The arrows indicate, where the user comes from and which way she may choose to follow. In contrary to the first widget, related resources with an important and meaningful property are displayed in the highlighted area to the left (2) while other related resources are displayed to the right (3).

Next to the resource labels, a number is displayed in braces, indicating the number of Yovisto videos related to this keyword. Clicking on a resource causes a new search process. In addition, all resources are linked to their corresponding Wikipedia article via the W icon.

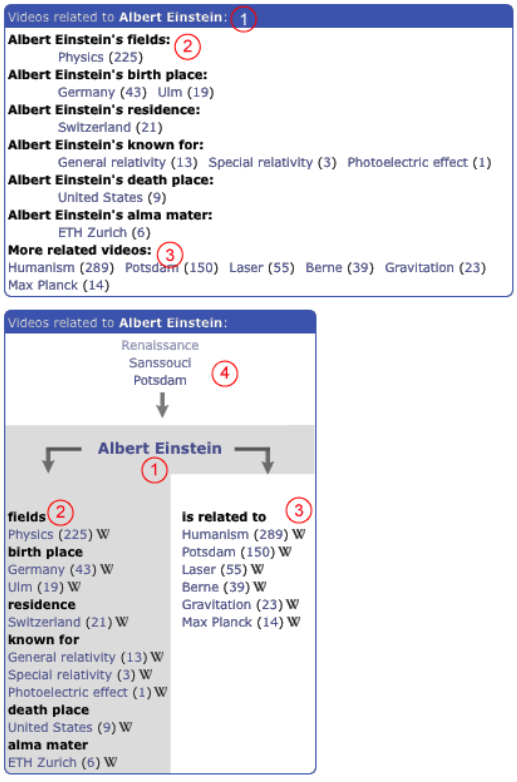

Figure 4. Exploratory search widgets.

IV. First Results

In this section performance issues are pointed out, first quantitive results were shown with a brief evaluation of how LOD resources complement our original video data.

A. Performance Issues

For the proposed explorative search, processing live queries with DBpedia and Wortschatz Leipzig is considered to be too time consuming. Especially, if many associated resources are found the processing of a single term might take up to one minute. Hence, an offline processing is set up to process every term beforehand. Furthermore, due to the numerous SPARQL queries, local replications of DBpedia and Wortschatz Leipzig have been set up.

On a standard PC about 1.000 index keywords can be processed per hour per machine in average. Overall, Yovisto's search index contains more than 1.5 million keywords and tags. Therefore, back-end processing has to be performed in parallel by multiple machine. However, performance optimization was not the objective of this work.

B. First Quantitative Results

Currently, more than 37.567 terms from the yovisto search index have been processed and 56.543 related resources have been found. Among them, 10.764 related resources have a meaningful property assigned to. 5.366 terms have at least one related resource with meaningful property assigned to and a total number of 637 meaningful properties have been identified. More than 300.000 new links have been generated between terms and related resources at all.

Fig. 5 shows the distribution of the number of related resources connected to terms *with* identified important properties (red solid line). More than 2.500 terms have at least one related resource, more than 1.000 terms have at least two related resources, etc. It is notable that the distribution reflects a power law, which was to be expected.

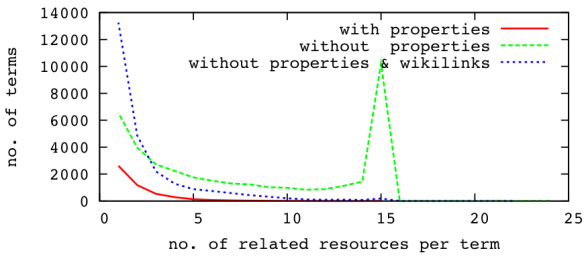

Figure 5. Associated resources with and without important properties found.

Furthermore, fig. 5 illustrates the distribution of related resources found *without* important properties (green dashed line). The same characteristics are shown as in the first case, except a peak at 15 caused by the wikilinks. Wikilinks do occur in large numbers and are limited to a threshold of 15, which distorts the diagram. Table I shows the distribution of

related results achieved by our heuristics. Wikilinks are the most frequently found relations, while inverse properties do only show up rather rarely, which was not expected.

Heuristic	# of occurence
Wikilink	233.161
Backlink	28.157
Cooccurence	24.685
Part of label	12.829
Property connecting same types	2.785
Pre-defined categories	1.772
Disambiguation	645
Inverse property	2

Table I
HEURISTICS DISTRIBUTION

V. CONCLUSION AND FUTURE WORK

We have shown, how to use Linked Open Data to enable a simple exploratory search for the Yovisto video search engine. By using LOD, we were able to make implicit existing relations among Yovisto resources explicit and to augment the ordinary keyword-based search by presenting additional related information and resources to the user via an appropriate interactive user interface.

In the first place, future work will address a proper evaluation of the achieved search results with traditional keyword-based search. Although, we have obviously increased the recall of obtained results by providing an exploratory search interface, the precision of the suggested resources has to be determined by the user and her personal information needs. Another problem to deal with lies in our multilingual approach (currently, we are working with English resources as well as with German resources). Furthermore, additional heuristics for finding meaningful related resources have to be evaluated.

Overall, we have implemented a first prototype of an exploratory video search, which gives the user the possibility to serendipitously discover and to explore resources that are usually hidden away from the user's eyes in the search engine index.

REFERENCES

[1] Semantic Web Education and Outreach Interest Group, "http://esw.w3.org/topic/SweoIG/TaskForces/CommunityProjects/LinkingOpenData/," 2009.

[2] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, "DBpedia: A Nucleus for a Web of Open Data," in *Proc. of 6th Int. Semantic Web Conf., 2nd Asian Semantic Web Conf.*, November 2008, pp. 722–735.

[3] M. Richert, U. Quasthoff, E. Hallensteinsdottir, and C. Biemann, "Exploiting the Leipzig Corpora Collection," in *Proceedings of the IS-LTC 2006*, Ljubliana, Slovenia, 2006. [Online]. Available: http://wortschatz.uni-leipzig.de/

[4] A. Pollitt, G. P. Ellis, and M. P. Smith, "HIBROWSE for Bibliographic Databases," *Databases Journal of Information Science*, vol. 20, no. 6, pp. 413–426, December 1994.

[5] Schraefel, M. Wilson, A. Russell, and D. A. Smith, "mSpace: improving information access to multimedia domains with multimodal exploratory search," *Commun. ACM*, vol. 49, no. 4, pp. 47–49, April 2006.

[6] P. Petratos, " Informing through User-Centered Exploratory Search and Human-Computer Interaction Strategies," *Issues in Informing Science and Information Technology*, vol. 5, pp. 705–727, 2008.

[7] G. Marchionini, "Exploratory search: from finding to understanding," *Commun. ACM*, vol. 49, no. 4, pp. 41–46, 2006.

[8] M. G. Christel, "Supporting video library exploratory search: when storyboards are not enough," in *Proc. of the Int. Conf. on Content-based image and video retrieval*. New York, NY, USA: ACM, 2008, pp. 447–456.

[9] H. Sack and J. Waitelonis, "Automated annotations of synchronized multimedia presentations," in *In Proceedings of the ESWC 2006 Workshop on Mastering the Gap: From Information Extraction to Semantic Representation, CEUR Workshop Proceedings*, june 2006.

[10] H. Sack and J. Waitelonis, "Integrating Social Tagging and Document Annotation for Content-Based Search in Multimedia Data," in *Proc. of the 1st Semantic Authoring and Annotation Workshop*, Athens (GA), USA, 2006.

[11] N. Day and J. M. Martínez, "Introduction to MPEG-7," International Organisation for Standardisation, Tech. Rep. ISO/IECT JTC1/SC29/WG11 N3751, Oct 2000.

[12] D. B. Ivan Herman, Ralph Swick, "Resource Description Framework (RDF)," W3C, Tech. Rep., 2004.

[13] C. Bizer, T. Heath, K. Idehen, and T. Berners-Lee, "Linked data on the web," in *Proc. of the 17th Int. Conf. on World Wide Web*. ACM, 2008, pp. 1265–1266.

[14] E. Prud'hommeaux and A. Seaborne, "SPARQL query language for RDF," W3C, January 2008.

[15] J. Waitelonis and H. Sack, "Augmenting video search with linked open data," in *Proc. of Int. Conf. on Semantic Systems 2009*, September 2009.

[16] International Organization for Standardization, "Information and Documentation – The Dublin Core Metadata Element Set," ISO 15836, November 2003.

[17] D. Brickley and L. Miller, "The Friend Of A Friend (FOAF) vocabulary specification," November 2007.

[18] S. R. Richard Newman, Danny Ayers, "Tag ontology," December 2005. [Online]. Available: http://www.holygoat.co.uk/owl/redwood/0.1/tags/

[19] R. Garcia and O. Celma, "Semantic integration and retrieval of multimedia metadata," in *Proc. of 4rd Int. Semantic Web Conf. Knowledge Markup and Semantic Annotation Workshop*, Galway, Ireland, 2005.