Semantically Enabled Exploratory Video Search

Jörg Waitelonis
joerg.waitelonis@hpi.uni-potsdam.de

Harald Sack
harald.sack@hpi.uni-potsdam.de

Johannes Hercher
johannes.hercher@hpi.uni-potsdam.de

Zalan Kramer
zalan.kramer@hpi.uni-potsdam.de

Hasso-Plattner-Institut für Softwaresystemtechnik GmbH
Prof.-Dr.-Helmert-Str. 2-3
14482 Potsdam, Germany

ABSTRACT

With the exponential growth of video data on the World Wide Web comes the challenge of efficient methods in video content management, content-based video search, filtering and browsing. But, video data often lacks sufficient metadata to open up the video content and to enable pinpoint content-based search. With the advent of the 'web of data' as an extension of the current WWW new data sources can be exploited by semantically interconnecting video metadata with the web of data. Thus, enabling better access to video repositories by deploying semantic search technologies and improving the user's search experience by supporting exploratory search strategies. We have developed the prototype semantic video search engine 'yovisto' that demonstrates the advantages of semantically enhanced exploratory video search and enables investigative navigation and browsing in large video repositories.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Search process, Information filtering, Query formulation; H.5.1 [**Multimedia Information Systems**]: Video

General Terms

Semantic video search

Keywords

linked data, semantic search, video search, exploratory search

1. INTRODUCTION

The popularity of video and multimedia data has become a driving force for growth in the World Wide Web (WWW). Portal sites such as, e. g., YouTube[1] and other content aggregators provide billions of video resources. While content management systems already are able to maintain large video repositories, search and pinpoint access of video content is left over to the user. Textual metadata enable content-based retrieval of video resources. But, for the most part, textual metadata originate from manual annotation carried out by authors or expert users. Automated video analysis techniques mostly provide low-level descriptions with sufficient accuracy, but the subsequent aggregation of different low-level features to derive high-level features concerning the contents is error-prone and algorithmic expensive. With the advent of Web 2.0 also user generated textual metadata such as, e. g., user tags or comments complement the authoritative manual annotation. Nevertheless, insufficient textual metadata is responsible for poor search performance.

Today, Google[2] has become synonymous for web search. The user enters a query string that might consist out of one or several keywords and Google's web search engine delivers (text) documents containing these keywords or multimedia documents annotated with metadata containing these keywords. In the majority of cases this approach is absolutely sufficient. But, users might have different information needs and different ways to search for information. If the search task is getting more complex, i. e. if a single document is not the answer to the user's search problem, different search strategies have to be applied. Moreover, if the user tries to get an overview of available information about a certain topic, today's web search engines often are flooding search results by millions. Thus, giving the user no chance to review but the first few resulting pages.

Sometimes, users are looking for a specific set of documents that do contain almost all the keywords of the query string (navigational searches), while in many other cases the user tries to gather information about a specific subject with no particular document in mind (research searches) [12]. In complex search tasks, the user first has to retrieve some facts (i. e. documents containing those facts), which are required to enable further search queries to solve the overall search problem. Often, the user is not familiar with the topic she is searching for, and sometimes, the user isn't sure about her search goal in the first place. This kind of search is known as 'exploratory search' [17]. In this paper, we address the problem of how to deploy exploratory search for video data by using semantic search technology.

In the Semantic Web as being an evolving extension of the WWW the meaning of information and services is well defined. Linked Data has become one of the most popular topics among the emerging Semantic Web [2]. By Linked Data we refer to a method of exposing, sharing, and connecting data via dereferenceable URI on the WWW. The Linking

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

[2] http://www.google.com/

WWW2010, April 26-30, 2010, Raleigh, North Carolina.
.

Open Data (LOD) project aims at making semantic data freely available to everyone and provides starting points to extract relationships among information resources.

By mapping available video metadata to Linked Data, we show how to utilize cross-connections and data interdependencies to enable new ways of search space visualization and navigation. One of the key interlinking hubs of the LOD cloud is DBpedia[3], the semantic counterpart of the online encyclopedia Wikipedia [1]. But, DBpedia often provides too many facts and details about entities. Thus, sophisticated heuristics have to be applied to achieve a ranking to identify the most important, i. e. interesting facts for a given entity [28].

Contrariwise to facetted search approaches, which aim to further refine an original search query by clustering the search results according to common properties, exploratory search broadens the scope of the search query by suggesting associated terms, concepts, and resources. These exploratory search suggestions can be used to navigate among the entire search space and to explore the repository content by user guided browsing [27]. Thus, enabling the possibility to achieve search results that the user wasn't looking for in the first place by serendipity.

The major contributions of this paper are the following: We have developed a frequency-based heuristics for ranking entity properties and relationships of LOD resources including efficient offline indexing for semantic video search. The most relevant properties are applied for the generation of contentually related search suggestions, and the quality of the overall exploratory search approach is shown with appropriate evaluation methods.

The paper is organized as follows. Section 2 presents related work and introduces to exploratory search, the yovisto semantic video search, and the general problem of evaluating exploratory search systems. Section 3 details, how Linked Data facilitates exploratory semantic search. In Section 4, our evaluation method for exploratory video search is presented together with first qualitative evaluation results. The last section concludes the paper with a brief outlook on future work.

2. RELATED WORK

This section covers research related to the work presented in this paper. This includes various approaches of semantic and exploratory search emphasizing the focus on video search in general and the yovisto academic video search platform in particular, including evaluation.

According to Guha et al. 'Semantic Search is the application of the Semantic Web to search' [12]. Correspondingly, Semantic Web resources and technologies are applied to augment the traditional search scenario. In the Semantic Web as being an extension of the current WWW information is complemented by formal knowledge representations with well defined meaning [3]. Linked Open Data has become one of the most popular topics among the emerging Semantic Web [4]. It supports the Semantic Web vision by leveraging free access to interconnected semantic data. In this paper we present a semantically enhanced exploratory search based on LOD resources, in particular on DBpedia, which contains structured information from the popular online encyclopedia Wikipedia [1]. A comprehensive survey on different approaches to Semantic Search is given in [16], while in [26] a formal model of ontology-based Information Retrieval in general is presented.

There are two main challenges for Semantic Search [12]: (1) the query input has to be mapped to concepts and entities [14, 18] and (2) the search domain has to be augmented with semantic content [9]. We address both challenges in this paper.

Alternatively, Semantic Search is often referred to as retrieval of data from the Semantic Web, as being represented by semantic search engines such as Sindice [20]. The work in this paper does not relate to this interpretation of Semantic Search.

Exploratory search assists in exploring the data space by providing data visualization, related content, a wide range of sophisticated selection options, and various possibilities for analysis. In order to allow a multi-facetted, multi-tactical, opportunistic, and open ended search, exploratory search is targeting learning and investigation tasks [17, 29]. To enable this complex and investigative retrieval option, underlying metadata including implicit and formerly unknown relationships among resources need to be fully made accessible. We extend and complement previous work on this topic by applying improved exploratory search to video data on the WWW and by framing suitable evaluation procedures for exploratory search [27, 28]. A motivation to use exploratory search over traditional search methods is given in [17].

State of the art video retrieval systems use a combination of visual and textual feature extraction for search index generation and combine these techniques with machine learning procedures [25, 11, 8]. However, the chosen feature extraction depends on domain and task characteristics and determines the quality of the retrieval system.

Yovisto is a video search engine specialized on video recordings of academic lectures and conference talks [22, 13, 27]. To enable pinpoint search access within the video recordings a combination of automated video analysis and user generated collaborative annotation is deployed in connection with fine granular, time-dependent metadata. We extend the search capabilities of yovisto by adding an exploratory search feature that enables the user to browse the content of the underlying video repository in a multi-facetted way. In difference to popular recommender systems, our approach is neither based on logfile analysis and statistical usage analysis of content popularity, nor on similarity-based methods such as query by example [15, 10, 5].

Evaluation of traditional information retrieval systems is based on rather quantitative than qualitative measurements of the achieved retrieval results. The retrieval results are compared to a ground truth resulting in an objective assessment of the achieved quality, i. e. by statistical classifications such as recall and precision. Evaluation strategies from the well known TrecVid benchmarks fall into this category. Their focus lies on pure system evaluation. Evaluation based on direct user involvement, referred to as 'User evaluation' is explicitly mentioned as out of scope for these benchmarks [24].

To demonstrate the added value of newly implemented retrieval features it is suitable to execute the same evaluation task with and without application of the specific retrieval feature. The differences between the resulting measurements mark the effect of the new retrieval feature. Singh et al. apply this evaluation strategy in [23]. We adopt this

[3] http://www.dbpedia.org/

approach for our evaluation to demonstrate the usefulness of exploratory search in yovisto. The motivation to use this strategy lies in the subjective and investigative nature of exploratory search [17], which makes it difficult to determine an objective ground truth for reference. Therefore, we apply additional qualitative evaluation measures by monitoring user satisfaction throughout the work task, as proposed in [21].

In [7] a framework for evaluation of interactive information retrieval systems is presented where the user task is formulated in a cover story leading to the work task and finally to the actual search task. Two evaluation strategies are compared, which distinguish multiple types of relevance, inter alia, so-called 'Situational Relevance', which reflects the dynamic nature of relevance [6]. Situational relevance also applies to exploratory search scenarios, where the user's relevance scale may be influenced by the receipt of new information.

3. IMPLEMENTING SEMANTIC EXPLORATORY SEARCH

This section presents the process of the exploratory semantic search and its implementation in the video search engine yovisto. In addition, a novel frequency-based heuristic for ranking entity properties and relationships of LOD is described complementing previous work [28]. In addition, a prototype graphical user interface (GUI) to deploy exploratory search is presented.

3.1 Exploratory Search Workflow

Exploratory search aims to broaden the scope of search by suggesting related terms, concepts and resources. Our approach uses LOD resources to support the search process by exposing additional information about indexed resources in yovisto, which are semantically interrelated to the users search query. For example, if someone is looking for video information about *'barack obama'* an additional navigation component shows supplementary and associated information resources about president Barack Obama also being available in the repository in addition to the direct search results (cf. sect. 3.5). This supplementary information includes, as e. g., related places (birth place, work place, etc), predecessor and successor in the presidential office, or his alma mater. Almost any DBpedia information can be made available as supplementary information to propose ancillary search possibilities to the user. But, there is too much information available in DBpedia to be displayed to the user in total. Therefore, the most important of all related resources have to be determined.

To enable this functionality a three-part approach is applied: (1) the user query is mapped to one or more DBpedia entities, (2) for each entity the best matching associated entities are determined, and (3) it is checked, which of the associated concepts are even available in the yovisto repository.

The best matching associated entities are determined based on predefined heuristics as described in [28]. Statistical measurements like property frequency or vice-versa property relatedness are taken into account to rank the associated entities. Fig. 1 depicts the overall workflow of exploratory search in yovisto including its particular components. The process is based on two main information sources, the *yovisto search index* and *DBpedia*. The yovisto search index is generated from automated, manual and collaboratively generated metadata for more than 8.000 videos [13, 22] and it contains more than 2 million terms. DBpedia provides interoperable RDF structured data about 2.9 million 'things' with 479 million 'facts'[4] for different domains.

Synonym	Type
John F. Kennedy	URI-suffix
John F. Kennedy	label
John Fitzgerald Kennedy	label
John Kennedy	redirect
J. F. K.	redirect
JFK	redirect
35th President of the United States	redirect
John f kenedy	redirect

Table 1: Synonyms generated for the concept: http://dbpedia.org/resource/John_F._Kennedy

Within the workflow a *gazetteer* dictionary and a *related entities* list are generated form the original data sources and stored in a database. The processing of this intermediate data is very computing-intensive. Therefore, computation is performed offline to maintain fast query responses. The online search process comprises *entity mapping* of the query string and *exploratory search suggestions.* While entity mapping connects the query string to DBpedia entities, the exploratory search suggestion proposes the most relevant associated resources available in the yovisto index to the user. The following sections encompass the workflow stages in detail.

3.2 Mapping Queries to Entities

The *user query* component propagates the query string q to the yovisto search index to compute the direct search results and to a *gazetteer lookup* component, which maps the query string q to a list of DBpedia entities $\{uri_i\}$. The gazetteer is a named-entity list created by *synset generation.* The gazetteer comprises a list of synonyms (synset) $\{s_i\}$ for every entity in DBpedia represented by its URI. Hence, the gazetteer is defined as set of (synset, URI)-tuples: $\{(\{s_i\}, uri_j)\}$.

To create the synset for an entity, different sources in DBpedia have to be utilized. The most reliable DBpedia source for entity mapping is the DBpedia URI-suffix. The URI-suffix denotes the string, which remains after removing the prefix '`http://dbpedia.org/resource/`' from the URI and replacing underscores '_' by single whitespaces. In the majority of cases, the URI-suffix denotes the entity most suitably. Additionally, literals of the DBpedia property *rdf:label* are eligible, if they differ from the URI-suffix. Table 1 shows synonyms determined for a given entity. In many cases *rdf:label* also provides labels in different languages, but in some cases, there is no *rdf:label* provided at all. Furthermore, so-called 'DBpedia redirects' are an additional source for synonyms. A redirect occurs, if a widely accepted different spelling or a common misspelling for the resource does exist. Redirects are identified by the DBpedia property '`http://dbpedia.org/property/redirect`'. Finally, the URI-suffix or labels of a redirect object are taken into account for synonyms.

We have applied the DBpedia dump files to generate syn-

[4] as of November 2009, http://wiki.dbpedia.org/Datasets

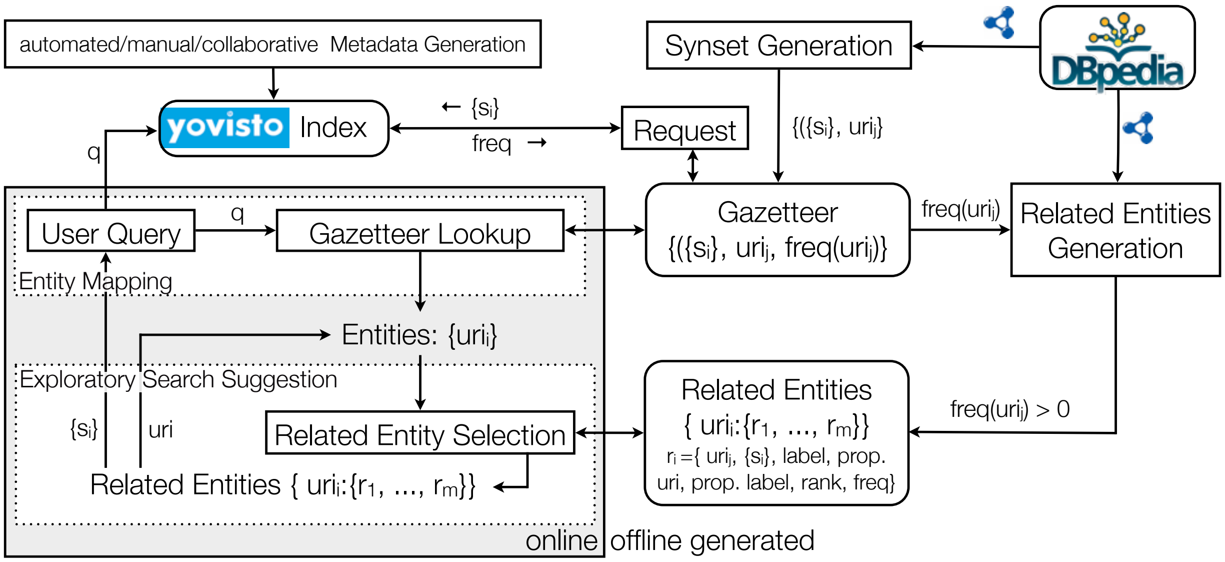


Figure 1: The process of exploratory search.

sets for more than 3.1 million different URIs. The gazetteer lookup maps the query string within the synsets and returns the matching entities $\{uri_i\}$. For every entity in this list, the *related entity selection* determines associated entities, which are finally suggested to the user, to carry on with the exploratory search. The structure of the related entities list and its generation is discussed in the following section.

3.3 Find Associated Entities

Related entities are determined by the application of a heuristic based ranking for the associated entities of a given entity. The component *related entities generation* is responsible for applying various heuristics to all 3.1 million entities in DBpedia and to filter out low ranked, i. e. not important associated entities. For a given entity a related resource is initially defined as:

$$r = (uri, \{s_i\}, l, p, pl, rank)$$

with uri the URI of the related entity, $\{s_i\}$ the synset for the URI, l a label to display to the user, p the property the entity is connected with the related entity, pl a property label for display, and $rank$ the ranking within all related resources of the entity.

In addition to the basic filtering heuristics already presented in [28] we have developed an improved frequency-based heuristic. This heuristic is based on the assumption that the more often a property occurs on instances of a specific category or type, the more relevant is it. As input for this heuristic the frequency of RDF properties used in conjunction with concepts of a specific RDF type (rdf:type) or SKOS[5] category (skos:subject) in DBpedia are taken into account. Table 2 shows the frequencies of various properties for all entities with skos:subject *Category:Presidents_of_the_United_States*. For exploratory search we suggest only related entities connected to the high frequent, i. e. most important properties. If a resource belongs to more than one skos:subjects, the frequencies of each property are added up.

No.	Property	Frequency
1	battles	29
2	predecessor	10
3	successor	10
4	navy	10
5	order	10
6	alongside	9
7	state	9
	...	
22	list	6
23	name	6
24	office	6
25	oldstyledatedyProperty	6
26	years	6

Table 2: Properties and occurrence frequencies on DBpedia entities with skos:subject *Category:Presidents_of_the_United_States.*

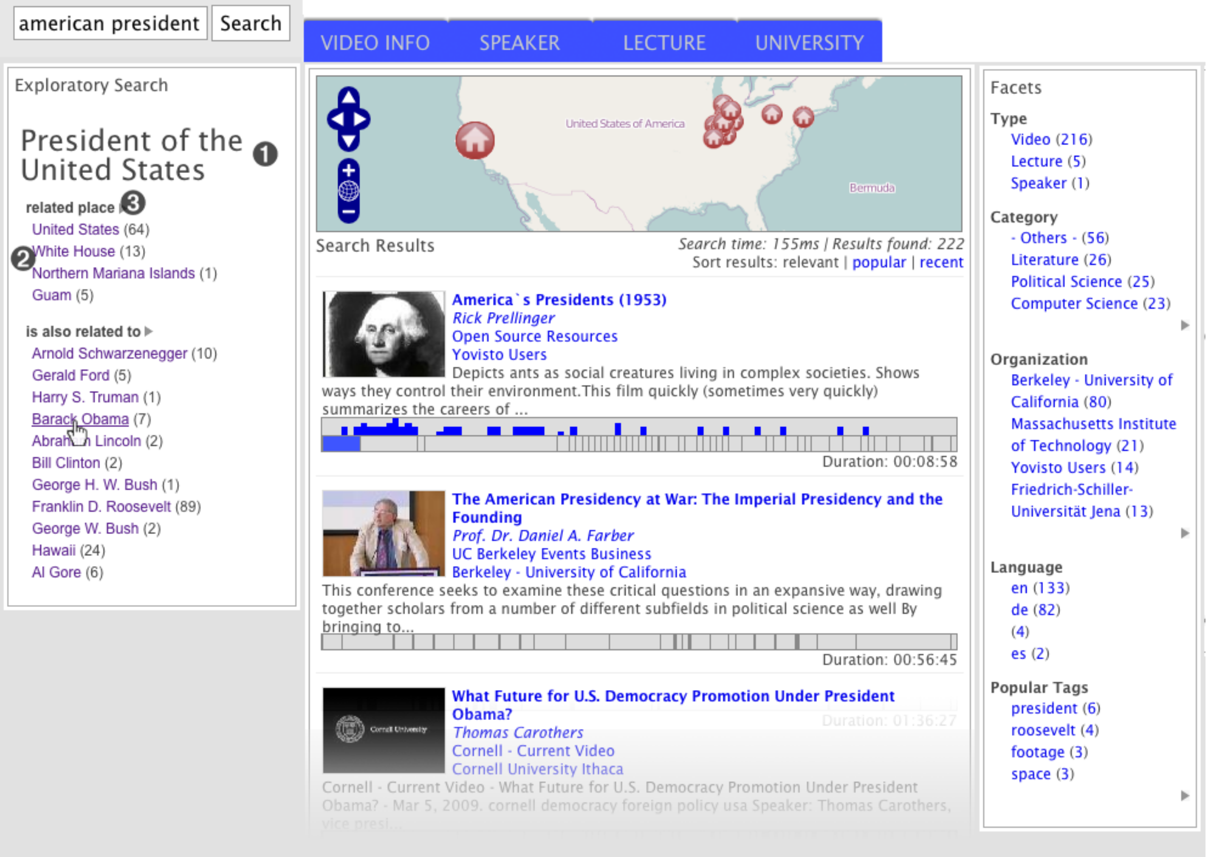
Furthermore, we have to ensure that only resources are suggested to the user that are also available in the yovisto index.

3.4 Mapping Index Data to Entities

To determine if resources for particular DBpedia entities are also available in the yovisto search index, an index *request* consisting of the related entity synset is generated. For this query all synonyms are interlinked with boolean OR. This ensures the completeness of the result list, consequentially increasing recall, but in some cases at the expense of precision. This index request returns the number of results for a direct search based on the entity synset, which is additionally stored in the gazetteer list and subsequently in the related resources. Finally, the related entities list contains only entities that are also present in the yovisto search index. Hence, the related entity list is defined as: $\{(uri_i : \{r_1, ..., r_m\})\}$ with $r_j = (uri_j, \{s_i\}, l, p, pl, rank, freq)$ with $freq > 0$.

Related entities to the original user query are displayed

[5] Simple Knowledge Organization Systems, a family of formal languages designed for representation of structured controlled vocabulary [19]



America`s Presidents (1953)
Rick Prellinger
Open Source Resources
Yovisto Users
Depicts ants as social creatures living in complex societies. Shows ways they control their environment.This film quickly (sometimes very quickly) summarizes the careers of ...
Duration: 00:08:58

The American Presidency at War: The Imperial Presidency and the Founding
Prof. Dr. Daniel A. Farber
UC Berkeley Events Business
Berkeley - University of California
This conference seeks to examine these critical questions in an expansive way, drawing together scholars from a number of different subfields in political science as well By bringing to...
Duration: 00:56:45

What Future for U.S. Democracy Promotion Under President Obama?
Thomas Carothers
Cornell - Current Video
Cornell University Ithaca
Cornell - Current Video - What Future for U.S. Democracy Promotion Under President Obama? - Mar 5, 2009. cornell democracy foreign policy usa Speaker: Thomas Carothers, vice presi...
Duration: 01:36:27

Figure 2: The exploratory search GUI showing related entities for 'american president'.

within a navigation window and the user is free to choose one of the recommended search suggestions. By clicking on one of the related resources a new user query is issued. At this stage the entity mapping can be omitted, because the URI for the query is already known. The synset of the selected suggestion is used to retrieve the new search results from the yovisto index.

3.5 The User Interface for Exploratory Search

We have designed the user interface for exploratory search to consist out of three main areas: the *direct search results* in the center including geographical information displayed in a map on top of the search results, the *facet filter* on the right, and the *exploratory search navigation* on the left (cf. Fig. 2). In the direct search results a timeline exposes the automatically generated temporal segmentation of the videos with highlighted segments that are relevant according to the current query string. The facet filter allows to narrow the search results according to the type of resource, the scientific category, the issuing organization of the video, the language of the video, as well as popular user tags attached to the video segments.

The exploratory search navigation is designed with a simple vertical layout to be displayed on the left to the direct search results. The exploratory navigation reflects the exploratory search workflow. When the user enters a query string, the labels of the mapped entities (1) are shown distinctly below the search input field followed by all related entities (2) grouped by their connecting properties (3). Next to the related entity labels a number in brackets denotes how many video resources for this particular entity exist in the yovisto video repository.

Fig. 2 depicts the result of a query for *'american president'* that is mapped to the DBpedia entity 'President of the United States'. The exploratory search GUI suggests a list of related entities. By clicking on, e. g. '*Barack Obama*', a new search is issued and the exploratory search GUI switches to the newly selected entity with its related entities and properties (cf. Fig. 3). To retain previous actions, a list (4) provides links, which allow to jump back in search history. Optionally, the user may activate a preview of the search results being evoked by a related entity, when clicking on it (5). Moving the mouse pointer over these previews causes a popup to show brief information about the video resource (6).

Some DBpedia properties such as *'predecessor'* (7, 8) have the characteristic trait to connect entities of the same type. They allow to move 'hand over hand' from one entity of a distinct category to the next. This enables the user to quickly exploit the information of individual entities.

A prototype implementation of the GUI is available at: `http://testing.yovisto.com/`.

4. EVALUATION

In this section an evaluation method for exploratory search scenarios is presented including the discussion of first results. While evaluation of traditional information retrieval systems

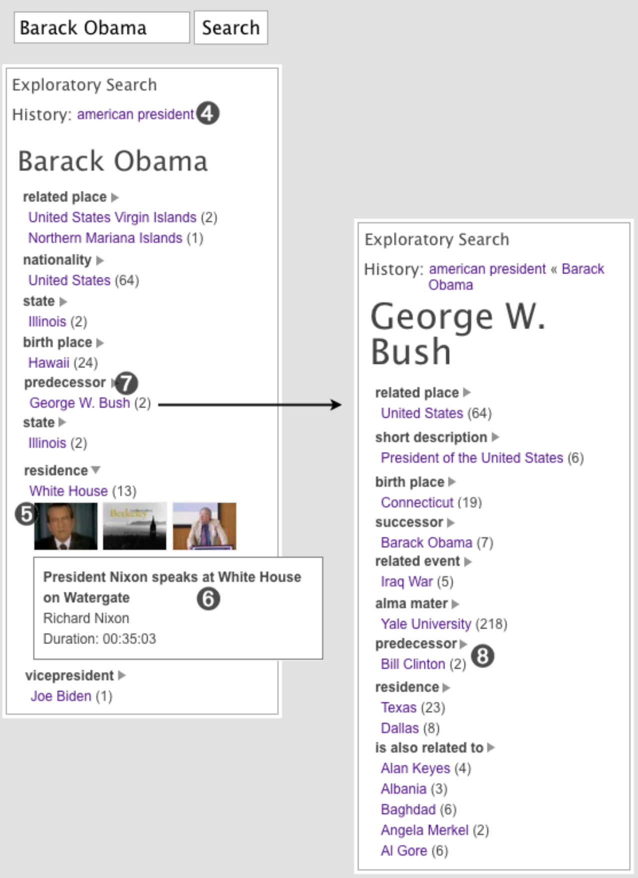


Figure 3: The exploratory search GUI showing related entities for 'Barack Obama' and 'George W. Bush'.

1. Which other scientists did Albert Einstein know personally in the 1920s and on which event he might got to know them?

2. Which philosophers build on the theories of the greek philosopher Plato?

3. Find videos with information about the German chancellors from 1949 until today.

4. Find videos about celestial bodies of the solar system.

5. Find videos about film directors.

6. Which videos contain information about US federal states?

7. Find videos about the founders and main promotors of the Enlightenment movement.

8. Find videos about cities of the Hanseatic League.

Table 3: Search tasks for evaluation.

focus on quantitative measures for the quality of retrieval results, the evaluation of exploratory search strongly depends on qualitative measurements. Concerning the definition of exploratory search, the user does not always exactly know what documents she is looking for. This stems because the user may be not familiar with the search topic, maybe she does not know where to begin and where to end the search, as well as she might not be sure about the search goal in the first place. Thus, it is rather difficult to define an objective ground truth for given exploratory search tasks, because individual search strategies, motivations, and interests cause ground truth also to depend on the eye of the beholder.

Therefore, quantitative evaluation measures such as precision and recall are less significant for exploratory search tasks than qualitative measurements, such as e. g., user satisfaction with the achieved search results and user experience during the search process.

We have made up 9 different search scenario tasks to be solved by test users. For exploratory search, the tasks have to be formulated in a way that there is most likely no direct answer possible. Moreover, the tasks must involve an iterative search strategy, where the answers being achieved in the first step are applied as input to the second search step, etc. E. g., instead of asking 'find videos about Barack Obama' we asked the user to retrieve videos about all US presidents. Thereby, in the first place, the user has to find out the names of the former US presidents before retrieving videos about them. To compare exploratory video search with traditional video search, we presented the same search tasks to users and asked one subset of users to solve the tasks with the exploratory search feature, while the rest had to solve the task without the exploratory search feature, i. e. without the exploratory search sidebar activated in the GUI.

Of course we first had to figure out which task topics were really suited for the yovisto video repository. The resulting evaluation tasks are listed in table 3. For the evaluation we did not limit the time required for each single task, but left it to the user to decide when to finish. Not all tasks were processed by every test person. While working on the retrieval tasks the test persons were asked after every partial search step, if it is still be possible to achieve the search objective in this search session, to gather information about the motivation of the test person. The evaluation interface also provides the possibility to select and mark relevant videos among the retrieval results according to the test person's opinion. The decision, if a video in the retrieval result is relevant or not can be made based on investigating the search results, which comprises surrogates of the videos such as image previews, preview text, user tags, comments, the video timeline, as well as reviewing the video itself. After finishing the search task, the user was instructed to review the selected videos again and to decide of the selection was appropriate. Finally, after finishing each task the user was asked, if she had achieved the search goal, how satisfied she felt with the achieved result, how helpful the search functionality was in general, and how familiar she is with the domain of the search task. Satisfaction, helpfulness, and familiarity were measured on scale from 0 (not at all) to 4 (very much).

Table 4 shows the results of the evaluation with respect to the tests *with* exploratory search (2nd column) and the control group tests *without* exploratory search (3rd column). A total number of 19 persons were participating the tests, 13 of them where using the exploratory search feature, 10 were involved in a control group. 70 tasks were processed with utilization of the exploratory navigation and 54 without ex-

	with exploratory search	**without exploratory search**
# persons	13 of 19	10 of 19
# of tasks	70	54
# of queries	790	699
task accompl.	38 (54.3 %)	15 (27.7 %)
task not accompl.	32 (45.7 %)	39 (72.3 %)
motivating queries	773 (97.8 %)	579 (82.2 %)
satisfaction (0–4)	1.99 (d: 1.37)	1.01 (d: 1.14)
helpfulness (0–4)	2.37 (d: 1.43)	1.60 (d: 0.83)
familiarity (0–4)	1.10 (d: 0.98)	0.89 (d: 0.98)
processing time	6.2 min/task (d: 3.7 min)	6.8 min/task (d: 4.1 min)
selected videos	183 (2.61 video/task)	111 (2.06 video/task)

Table 4: Results of evaluation. (d - standard deviation)

ploratory navigation. For all 70 tasks a total number of 790 queries where issued. The control group needed 699 queries for 54 tasks. 54.3 % of the exploratory search tasks were accomplished successfully by the participants. The control group accomplishes only 27.7 % of tasks successfully. While processing the queries, in 97.8 % of queries the participants felt, that it is possible to achieve the search objective. The control group was just with 82.2 % of the queries motivated.

On scale from 0 to 4, with exploratory search the satisfaction was evaluated to 1.99 in average. The control group was only satisfied with 1.01 in average. The helpfulness was assessed with 2.37 with exploratory search, whereas the control group has only assessed 1.60. The familiarity was measured to 1.10 with exploratory search and 0.89 without. The average task processing time is observed with 6.2 minutes using exploratory search and 6.8 minutes without exploratory search. Finally, 2.61 videos per task were considered to be relevant with exploratory search, whereas 2.06 videos per task were selected without exploratory search. Table 4 shows also the standard deviations for the particular results.

Summarizing the results, the number of tasks accomplished successfully was doubled up to 54.3 % by use of the exploratory search. The motivation of participants was significantly higher with the exploratory search feature. Satisfaction was increased by 20 %, helpfulness was increased by 15 %. Familiarity and processing time were improved by use with exploratory search, but not very much. In general, exploratory search leads to more selected videos. A detailed overview on the presented and further evaluation results can be found at: `http://www.yovisto.com/evaluation/`.

Our results reflect the usability of the GUI and the user's satisfaction with the quality of the achieved results. A refinement of the evaluation could be achieved by focusing on these two aspects separately. Furthermore, the heuristics-based recommendation of related entities to a given user query is an integral part of the exploratory search. These heuristics are depending on local repository data and on LOD resources from DBpedia. Thus, an evaluation of these heuristics independently of the achieved results could be conducted to measure the impact of recommendations independently form the usability of the GUI.

5. CONCLUSION

In this paper, we addressed the problem of how to deploy exploratory search for video data by using semantic search technology and demonstrated an improved exploratory search for the yovisto video search engine along with an evaluation of the exploratory search process. Exploratory search is supported by heuristics based on semantic data from the LOD resources, which are used to augment direct search result with navigational information that might be also relevant for the user. We presented a new frequency-based heuristic for determining related information resources and evaluated the exploratory search feature by using a comparative evaluation technique. Thus, we were able to objectively point out the positive aspects of the exploratory search approach such as a higher task accomplishment, higher motivation and satisfaction rates.

Exploratory search is at it's early stages as a research area. Currently, there does not exist an overall accepted best-practice neither on how to realize nor on how to evaluate exploratory search. We aimed to overcome this shortcoming by strongly relating our accomplishments and methods to existing research in this area.

Since the heuristics for the determination of 'interesting' relationships among data entities play a major role in the exploratory search approach, we are developing objective evaluation procedures for the optimization of these heuristics. In addition, improvements of the graphical user interface explicitly supporting the investigative and navigational aspect of our approach will be considered in future work.

For better support in data space navigation, future work is focussed on the combination of facetted and explorative search features to satisfy the searchers curiosity and to foster serendipitous discovery.

6. REFERENCES

[1] 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.*, pages 722–735, November 2008.

[2] T. Berners-Lee. Linked Data. World wide web design issues, July 2006.

[3] T. Berners-Lee, J. Hendler, and O. Lassila. The Semantic Web. *Scientific American*, 284(5):34–43, 2001.

[4] 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*, pages 1265–1266. ACM, 2008.

[5] J. Bollen, M. L. Nelson, G. Geisler, and R. Araujo.

Usage derived recommendations for a video digital library. *J. Netw. Comput. Appl.*, 30(3):1059–1083, 2007.

[6] P. Borlund. The concept of relevance in IR. *J. Am. Soc. Inf. Sci. Technol.*, 54(10):913–925, 2003.

[7] P. Borlund. The IIR evaluation model: a framework for evaluation of interactive information retrieval systems. *Information Research*, 8(3), 2003.

[8] X. Chen and C. Zhang. An interactive semantic video mining and retrieval platform - Application in transportation surveillance video for incident detection. In *ICDM 2006: Proc. of 6th IEEE Int. Conf. on Data Mining, Hong Kong*, pages 129–138. IEEE Computer Soc., 2006.

[9] A. Duke and J. Heizmann. Semantically Enhanced Search and Browse. *Semantic Knowledge Management*, pages 85–102, 2009.

[10] F. Fouss and M. Saerens. Evaluating performance of recommender systems: An experimental comparison. *Web Intelligence and Intelligent Agent Technology*, 1:735–738, 2008.

[11] G. Friedland, L. Gottlieb, and A. Janin. Joke-o-mat: browsing sitcoms punchline by punchline. In *Proc. of the 17th ACM Int. Conf. on Multimedia*, page 11151116, Beijing, China, 2009. ACM.

[12] R. Guha, R. McCool, and E. Miller. Semantic search. In *WWW '03: Proc. of the 12th Int. Conf. on World Wide Web*, pages 700–709, New York, NY, USA, 2003. ACM Press.

[13] 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.

[14] J. Hu, G. Wang, F. Lochovsky, J. T. Sun, and Z. Chen. Understanding user's query intent with wikipedia. In *WWW '09: Proc. of the 18th international conference on World wide web*, pages 471–480, New York, NY, USA, 2009. ACM.

[15] E. H. S. Lo, M. R. Pickering, M. R. Frater, and J. F. Arnold. Query by example using invariant features from the double dyadic dual-tree complex wavelet transform. In *Proc. of the ACM Int. Conf. on Image and Video Retrieval (CIVR '09)*, pages 1–8, New York, NY, USA, 2009. ACM.

[16] C. Mangold. A survey and classification of semantic search approaches. In *Int. J. Metadata, Semantics and Ontology*, volume 2, pages 23–34, 2007.

[17] G. Marchionini. Exploratory search: from finding to understanding. *Commun. ACM*, 49(4):41–46, 2006.

[18] E. J. Meij, M. Bron, B. Huurnink, L. Hollink, and M. de Rijke. Learning semantic query suggestions. In *8th Int. Semantic Web Conf. (ISWC 2009)*. Springer, October 2009.

[19] A. Miles and S. Bechhofer. Skos simple knowledge organization system reference. World Wide Web Consortium, Working Draft, August 2008.

[20] E. Oren, R. Delbru, M. Catasta, R. Cyganiak, H. Stenzhorn, and G. Tummarello. Sindice.com: a document-oriented lookup index for open linked data. *IJMSO*, 3(1):37–52, 2008.

[21] Y. Qu and G. W. Furnas. Model-driven formative evaluation of exploratory search: A study under a sensemaking framework. *Inf. Process. Manage.*, 44(2):534–555, 2008.

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

[23] H. Singh, A. Cheung, S. Guadarrama, C. Loer, and M. Nikravesh. Evaluating Ontology Based Search Strategies. In *Soft Computing for Information Processing and Analysis*, volume 164, pages 189–202, 2006.

[24] A. F. Smeaton, P. Over, and W. Kraaij. Evaluation campaigns and TRECVid. In *MIR '06: Proceedings of the 8th ACM international workshop on Multimedia information retrieval*, pages 321–330, New York, NY, USA, 2006. ACM.

[25] A. W. M. Smeulders, J. C. van Gemert, B. Huurnink, D. C. Koelma, O. de Rooij, K. E. A. van De Sande, C. G. M. Snoek, C. J. Veenman, and M. Worring. Semantic video search. In *Proc. of 14th Int. Conf. on Image Analysis and Processing*, pages 51–58. IEEE Computer Soc., 2007.

[26] D. T. Tran, S. Bloehdorn, P. Cimiano, and P. Haase. Expressive resource descriptions for ontology-based information retrieval. *Proc. of the 1st Int. Conf. on the Theory of Information Retrieval (ICTIR'07)*, October 2007.

[27] J. Waitelonis and H. Sack. Augmenting video search with linked open data. In *Proc. of Int. Conf. on Semantic Systems 2009 (i-Semantics 2009)*, 2009.

[28] J. Waitelonis and H. Sack. Towards Exploratory Video Search Using Linked Data. In *ISM '09: Proc. of the 2009 11th IEEE Int. Symp. on Multimedia*, pages 540–545, Washington, DC, USA, 2009. IEEE Computer Society.

[29] R. W. White, G. Marchionini, and G. Muresan. Evaluating exploratory search systems: Introduction to special topic issue of information processing and management. *Information Processing & Management*, 44(2):433 – 436, 2008.