Squirrel: An Advanced Semantic Search and Browse Facility

Alistair Duke, Tim Glover, and John Davies

Next Generation Web Research Group, BT Group, Adastral Park, Ipswich, UK
{alistair.duke,tim.glover,john.nj.davies}@bt.com

Abstract. Search is seen as a key application that can benefit from semantic technology with improvements to recall and precision over conventional Information Retrieval techniques. This paper describes Squirrel, a search and browse tool that provides access to semantically annotated data. Squirrel provides combined keyword based and semantic searching. The intention is to provide a balance between the speed and ease of use of simple free text search and the power of semantic search. In addition, the ontological approach provides the user with a much richer browsing experience. Squirrel builds on and integrates a number of semantic technology components. These include machine learning and information extraction components which generate, extract and manage semantic metadata contained within and about textual documents at index time. A number of run-time components have also been integrated to deliver an enhanced user experience which goes beyond merely presenting a list of documents as a query response. The tool has been trialled and evaluated in two case studies and we report early results from this exercise, revealing promising results.

1 Introduction

Search engines based on conventional IR techniques (employing keyword and phrase matching between the query and index) alone tend to offer high recall and low precision. The user is faced with too many results and many results that are irrelevant. A major reason for this is the failure to handle polysemy and synonymy, or more generally, the view of a document purely as a bag of words upon which no semantic analysis is carried out. Although it is possible to disambiguate queries by adding more terms and by making use of more sophisticated functions (e.g. Boolean operators) there is evidence to suggest that the majority of users do not do this [8]. Algorithms such as Google's PageRank offer significant improvement over previous approaches, however, the approach tends to attribute a low rank to new pages about popular topics and the rank stays low if people cannot find the site and then link to it. Also, the result set continues to be too large and context is not considered which means results from queries about unpopular topics are still hard to pick out. Moreover, there are doubts about the suitability of the algorithm for ranking pages on corporate intranets due to their more regular structure [15]. A further point is that people require information rather than just a list of links where they might find what they are looking for.

This paper describes Squirrel, a search and browse tool based on semantic technology. Squirrel aims to address the issues stated above through combined keyword based and semantic searching by allowing the user to initially enter free text terms and see results immediately but following this to allow them to refine their queries with the use of ontological support e.g. by selecting from a set of returned topics or matching entities. The intention is to provide a balance between the speed and ease of use of simple free text search and the power of semantic search. In addition, the ontological approach provides the user with a much richer browsing experience.

Squirrel has been developed in the EU SEKT project[1]. One of the main aims of SEKT has been to address the metadata bottleneck by providing semi-automatic semantic annotation and ontology generation tools. Squirrel builds upon the efforts of SEKT in this area to allow users to more easily find relevant knowledge with the support of semi-automatically derived semantic metadata about unstructured textual documents. The metadata describes both documents themselves and their content. Squirrel is targeted a large enterprises where it aims to offer a single search interface over the heterogeneous data sources that are common in that setting.

Squirrel provides a number of novel features to enhance the search and browse experience. These include natural language generation which provides natural language summaries of knowledge held in formal (ontological) structures; device independence which allows the tool to be run on multiple devices; and result consolidation which presents the most relevant textual content of result documents rather than a simple list of results.

The paper is structured as follows. The next section introduces a scenario which illustrates both the requirements and benefits of enhanced search and browse. Section 3 describes the salient features of the supporting components that comprise Squirrel and presents an architecture for the system. In Section 4 further detail is provided about the user experience with a description of the important features of the interface. Early results from an evaluation exercise are described in Section 5. Section 6 discusses related work and outstanding issues whilst we conclude in Section 7 with a view of the way forward.

2 Scenario

Squirrel has been trialled within the SEKT project in a case study which is developing an improved Digital Library. The following scenario describes a Digital Library user carrying out a knowledge seeking task, making use of the features provided by Squirrel and its supporting components.

The user has an initial high level goal to seek information about the field of Home Health Care and has little idea about what is contained in the library which might be of use. They first enter 'Home Health Care' into the search box and hit the 'Go!' button. The first result screen includes a summary of the sorts of resources that have been found that might be able to meet their needs. These include the textual resources from the library i.e. that there are a number of journal articles, conference papers,

[1] http://www.sekt-project.com

periodicals and web pages (that have been shared by library users and indexed as library resources) that match their query. In addition, there are a number of library topics (subject areas against which relevant documents are classified) that also match their query including one which is itself called 'Home Health Care'. Further matches include a number of organizations from a domain knowledge base whose description includes the search term.

In addition to this summary, the user is also presented with the top ranked textual resources (in typical search engine style). This simple list of documents is augmented with the ability to refine the search based on the properties of the documents in the result set, including a hierarchical display of the topics of the results documents, date of publish, author name, etc.

Our user decides that they are interested in current affairs related to their topic rather than scientific papers. They choose to refine the search to the periodicals section of the library. The topic hierarchy is rebuilt to accommodate the documents in the refined results set. Furthermore, our user is interested in the financial aspects of home health care and notices that a topic called 'Economic Conditions' has been presented. They then choose this topic to further refine their results set.

A short taster of each result together with its title and other metadata is presented. This is enhanced by highlighting the named entities that have been recognized in the text such as people, organization, locations, etc. Our user places their mouse of one such highlighted term, 'United Health Products' and is shown a popup which tells them that this is a company. Intrigued about the company, they click on the term and are taken to a screen giving further information about the company including a natural language summary about the company as well as related entities such as the people who work for it. There is also a link to all of the documents where this company (including any of its aliases such as UHP, etc.) is mentioned. Our user clicks on this link but then decides they'd rather view the results as a consolidated summary rather than a list of discrete results. In this view the most relevant portions of the result documents are shown to the user meaning they can quickly view the available material rather than having to navigate to several different pages. The user again refines the contents of the summary by selecting one of more of the topics that have been assigned to the presented material.

3 Architecture

Squirrel integrates a number of components developed in the SEKT project. This section briefly describes the features of the components and then presents an architecture illustrating how they have been integrated.

3.1 PROTON

PROTON is a lightweight general purpose ontology developed in SEKT which includes a module specific to the knowledge management domain. PROTON is used by SEKT components for metadata generation and as a basis for knowledge modelling and integration.

A world knowledge base (originally developed as part of the KIM platform [11]) has been expressed in PROTON. The knowledge base is comprised of more than 200,000 entities. These are gathered semi-automatically from a range of high quality public data sources. It includes around 36,000 locations, 140,000 companies and other organisations and the world's pre-eminent politicians, businesspeople and technologists

3.2 Full-Text Index

The Lucene full-text indexing facility is used to provide an index over the various forms of textual resource but also over the labels and literal properties of the ontological instances and classes. This allows metadata e.g. authors' names, topic names, knowledge base entities to be discovered and presented to the user as a way to refine their search or as an alternative way to find documents, since the metadata is always related to a set of documents in some way. The index of ontological instance will, given a search term, provide a set of matching URIs which can then be used to query the ontology repository to extract further details of the concept they refer to.

3.3 KAON2

The KAON2 [7] ontology management and inference engine provides an API for the management of OWL-DL and an inference engine for answering conjunctive queries expressed using the SPARQL[2] syntax. KAON2 also includes a module for extracting ontology instances from relational databases via a mapping facility. This greatly increases the number of instances that can be held in the ontology as compared to a file based version.

KAON2 also supports the Description Logic-safe subset of the Semantic Web Rule Language[3] (SWRL). This allows knowledge to be presented against concepts that goes beyond that provided by the structure of the ontology. For example, one of the attributes displayed in the document presentation is 'Organisation'. This is not an attribute of a document in the PROTON ontology; however, affiliation is an attribute of the Author concept and has the range 'Organisation'. As a result, a rule was introduced into the ontology to infer that the organisation responsible for a document is the affiliation of its lead author.

3.4 Natural Language Generation

Natural Language Generation (NLG) takes structured data in a knowledge base as input and produces natural language text, tailored to the presentational context and the target reader [12]. In the context of the semantic web and knowledge management, NLG is required to provide automated documentation of ontologies and knowledge bases and to present structured information in a user-friendly way [1].

When a Squirrel search is carried out and certain people, organisations or other entities occur in the result set, the user is presented with a natural language summary of information regarding those entities. In addition, document results can be enhanced

[2] http://www.w3.org/TR/rdf-sparql-query/

[3] http://www.w3.org/Submission/SWRL/

with brief descriptions of key entities that occur in the text. For example, the knowledge base contains company related metadata, which can be presented to the user as natural language. Squirrel uses the ONTOSUM component described in [1]. An example is shown in Figure 3.

3.5 DIWAF

The SEKT Device Independence Web Application Framework is a server side application, which provides a framework for presenting structured data to the user [5]. The framework does not use a mark-up language to annotate the data. Instead, it makes use of templates, which are "filled" with data, rather like a mail merge in a word processor. These templates allow the selection, repetition and rearrangement of data, interspersed with static text. The framework can select different templates according to the target device, which present the same data in different ways.

Squirrel has been built as a DIWAF application. This enables the applications to be easily adapted to alternative device or context requirements (such as different languages). Currently, templates are available to deliver the application to users via a WAP-enabled mobile device, via a PALM PDA and via a standard web browser.

3.6 Ontology Generation

The Ontology Generation (OG) component can automatically identify an ontology from the content of the documents and then classify the documents according to the ontology [4]. In SEKT this process is generally carried out at index-time which allows the knowledge engineer to modify the generated ontology if required. The process results in hierarchical topic ontology and a set of document classifications into that ontology (in accordance with PROTON). These results are used by Squirrel to present the topic hierarchy thus allowing the user to refine their search by topic. The OG component can also be used by Squirrel to generate clusters of documents at query-time. This could be used in the general case where some or all of the documents being searched have not been classified at index-time.

3.7 User Profile Construction and Usage

The PROTON ontology includes concepts allowing the interests of users to be stored and modelled as a profile. Such a profile is an important step in providing relevant knowledge to people and is used within Search and Browse to personalise the search experience. The profile allows the search tool to predict what the user is interested in based upon their observed interests. Profiles can be manually or automatically created. An automatic approach places less burden on the user than a manual one but relies on the assumption that the techniques used to collect the profile are accurate (which is why a combination of automatic and manual approaches is often adopted). The approach adopted in SEKT has been to observe the web browsing habits of the individual users and to extract the topics associated with the pages that have been accessed. A level of interest for each of the topics is also determined.

In PROTON, the profile consists of an expression of both long-term and short-term interest in topics from a PROTON topic ontology. The Squirrel search tool makes use of these profiles to modify the way the results are presented to the user i.e. providing context to the user's search query. When ranking documents, the topics of the documents in the result set can be considered against the level of interest in topics in the user's profile. Thus documents with a topic recorded in the profile with a strong short-term interest would be presented first. Topics are related to other topics (either by a sub or super topic or some other weaker relation) so these relationships can also be considered to support the application of profiles.

3.8 Massive Semantic Annotation

Squirrel makes use of the results of the Massive Semantic Annotation process, carried out at index time. The function uses KIM [11], which employs GATE's ontology-based information extraction module to identify named entities in texts. For each of these it carries out instance disambiguation i.e. it either identifies an existing PROTON instance of which the entity is an occurrence or creates a new instance. An annotation (which consists of the instance URI, the document it occurs in and its location within the text) is stored using KIM's highly scalable (due to the very large number of annotations) knowledge store.

In order to support user responsive entity presentation and browsing, Squirrel makes use of the OWLIM semantic repository [9]. OWLIM is considered to be the fastest OWL repository available. The dual repository approach adopted by Squirrel is justified by the benefits that each repository provides. KAON2 offers relational database mapping and rule support whilst the scalability and speed of OWLIM are suited to handling the volume of data resultant from the semantic annotation process.

3.9 Segmentation

During the indexing process, documents are segmented at topical boundaries in order to support the presentation of consolidated results to the user. The segmenter used is a C99 segmenter [2] which is enhanced to consider the distribution of named entities and key phrases identified in the text in addition to the distribution of words. Following segmentation the subdocuments are classified using the OG classifier described above. These classifications are used in two ways by Squirrel. Firstly, to allow the user to refine their summaries based upon topics of interest and secondly to reorder the presentation of the summary based upon the topics in the user's profile.

3.10 Integration

The components described above have been integrated as a complete end-to-end architecture for indexing and querying

The sources of textual data are stored together with their associated metadata in a database. These sources may be e.g. records of bibliographic data (as is the case in the SEKT Digital Library case study), webpages identified by a crawler, or legal judgements (as is the case in the SEKT Legal case study). At this stage, the metadata can be augmented by one or more entity extraction or classification components.

Once these steps have been carried out, the contents of the database can then be indexed by Lucene which indexes all the textual fields in the database, allowing subsequent keyword based retrieval. Entities within the PROTON world knowledge base are described by one or more aliases. These aliases are also added to the index allowing the entities to which they refer to be retrieved at query time.

A parallel activity to free-text indexing is to load the contents of the database into KAON2. KAON2 provides a facility to map database schema to an ontological representation. This allows KAON2 to represent the contents of the database as a PROTON ontology. In addition to the database, KAON2 also reads in the PROTON ontology, its world knowledge base and user profiles.

User queries are first passed to Lucene in order to retrieve a set of matching documents and entities by their URI. Squirrel then queries KAON2 to extract further details about these concepts as described in Section 3.2

Following this, Squirrel is able to present an initial result to the user. The user interface and the various options provided by Squirrel for refining and presenting results are described in Section 4.

4 Interface Description

4.1 Initial Search

Users are permitted to enter terms into a text box to commence their search. This initially simplistic approach was chosen based on the fact that users are likely to be comfortable with it due to experience with traditional search engines. If they wish, the user can specify which type of resource (from a configurable list) they are looking for e.g. publications, web articles, people, organisations, etc. although the default is to search in all of these.

The first task Squirrel carries out after the user submits a search is to call the Lucene index and then use KAON2 to look up further details about the results, be they textual resources or ontological entities. In addition to instance data, the labels of ontological classes are also indexed. This allows users to discover classes and then discover the corresponding instances and the documents associated with them without knowing the names of any of the instances e.g. a search for 'Airline Industry' would match the 'Airline' class in PROTON. Selecting this would then allow the user to browse to instances of the class where they can then navigate to the documents where those instances are mentioned. This is an important feature since with no prior knowledge of the domain it would be impossible to find these documents using a traditional search engine.

Textual content items can by separated by their type e.g. Web Article, Conference Paper, Book, etc. Squirrel is then able to build the meta-result page based upon the textual content items and ontological instances that have been returned.

4.2 Meta-result

The meta-result page is intended to allow the user to quickly focus their search as required and to disambiguate their query if appropriate. The page presents the different types of result that have been found and how many of each type. In order not to introduce unnecessary overhead on the user, the meta-result page also lists a default set of results allowing the user to immediately see those results deemed most relevant to their query by purely statistical means.

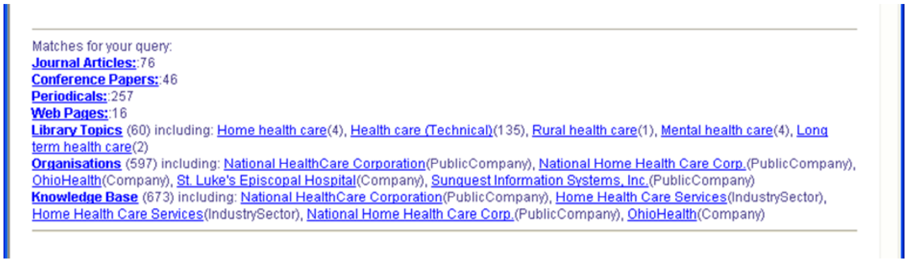

Matches for your query:
Journal Articles::76
Conference Papers::46
Periodicals::257
Web Pages::16
Library Topics (60) including: Home health care(4), Health care (Technical)(135), Rural health care(1), Mental health care(4), Long term health care(2)
Organisations (597) including: National HealthCare Corporation(PublicCompany), National Home Health Care Corp.(PublicCompany), OhioHealth(Company), St. Luke's Episcopal Hospital(Company), Sunquest Information Systems, Inc.(PublicCompany)
Knowledge Base (673) including: National HealthCare Corporation(PublicCompany), Home Health Care Services(IndustrySector), Home Health Care Services(IndustrySector), National Home Health Care Corp.(PublicCompany), OhioHealth(Company)

Fig. 1. Meta-result

The meta-result for the 'home health care' query is shown in Figure 1 under the sub-heading 'Matches for your query'. The first items in the list are the document classes. Following this is a set of matching topics from the topic ontology. In each case, the number in brackets is the number of documents attributed to each class or topic. Following the topics, a list of other matching entities is shown. The first 5 matching entities are also shown allowing the user to click the link to go straight to the entity display page for these. Alternatively they can choose to view the complete list of items. Squirrel can be configured to separate out particular entity types (as is the case with topics and organisations as shown in Figure 1) and display them on their own line in the meta-result.

The returned documents can be ranked according to the user's profile if one has been specified. The algorithm to perform this ranking checks to see which documents are attributed to topics that exist in the profile and then adjusts the ranking using the degree of interest for those topics. For each document, the degree of relevance provided by Lucene is added to the cumulative degree of interest from the topics in the profile that are also attributed to the document. The documents are then re-ranked based upon these new relevance figures and displayed to the user. The algorithm can be formulised as shown in Equation 1.

$$R = r + \sum_{i=1..k.} \begin{cases} w(t_i) & t_i \in U \\ 0 & t_i \notin U \end{cases} \tag{1}$$

Where R is the degree of relevance for a document and r is the (statistical) degree of relevance provided by Lucene. A topic attributed to the document is denoted by t_i where i is the index of a topic in the set of attributed topics. $w(t_i)$ denotes the level of interest in the topic t_i if it is a member of the set of topics in the user's profile, U. If the topic is not in the profile then no adjustment is made.

The algorithm attempts to maintain a balance between results with a high relevance ranking mainly due to the Lucene result and those that are principally deemed to be of interest according to the profile. Thus, if the user has just switched their focus, the results should not skew too far in favour of the profile which might be slightly biased towards their previous focus.

4.3 Refining by Topic

Alongside the results, the user is presented with the topics associated with the documents in the result set. Not all topics are shown here since in the worst case where each document has many distinct topics the list of topics presented to the user would be unwieldy. Instead an algorithm takes the list of topics that are associated with the collection of documents in the result set, and generates a new list of topics representing the narrowest "common ancestors" of the documents' topics.

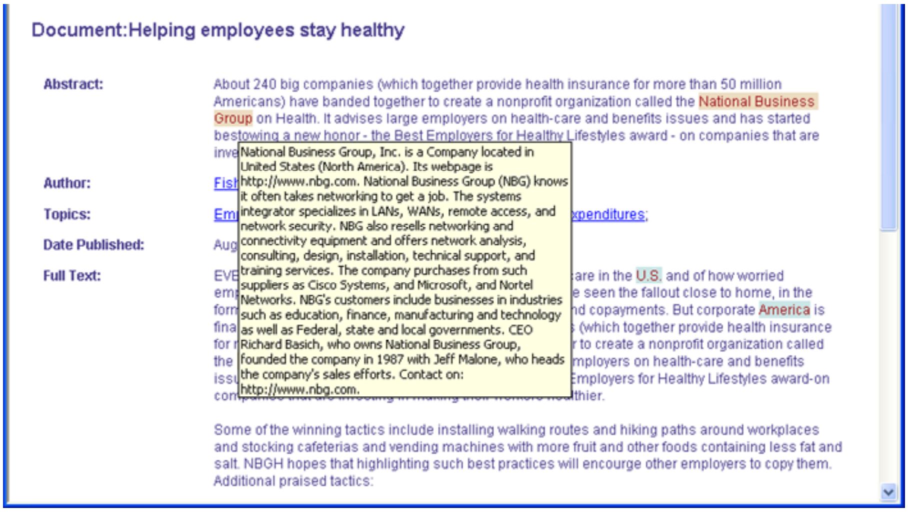
Having selected a topic and viewed the subset of documents from the result set, the user can switch to an entity view for the topic. The user can also reach this view by selecting a topic from the meta-result section. Instead of showing the documents of the topic, the meta-data for the topic is shown, which includes the broader, narrower and related topics. This allows the user to browse around the topic ontology. Each topic is shown with the number of documents it contains in brackets after it. Two links to document results are also shown. The first takes the user to a list of documents that have been attributed to the topic itself. The second takes the user to a list of documents that include all subtopics of the current topic. The layout of entity views in Squirrel are defined by templates. This allows the administrator to determine what meta-data is shown and what is not. The use of these templates is discussed further in Section 4.6 where a 'Company' entity view is described.

4.4 Attribute-Based Refinement

Any document result list has a link, which opens a refiner window. This allows the user to refine the results based upon the associated metadata. The metadata shown and the manner in which it is displayed are configurable through the use of entity type specific templates that are configured by an administrator or knowledge engineer. Documents can be refined by the user based upon their authors, date of publication, etc. The approach adopted has been to allow the user to enter free text into the attribute boxes and to re-run the query with the additional constraints. An alternative would be to list possible values in the result set. However, the potential size of this list is large and is difficult to present to the user in a manageable way. The downside to the free-text approach is that the user can reduce the result set to zero by introducing an unsatisfiable constraint – which is obviously undesirable. Squirrel attempts to address this by quickly showing the user the size of the result set once constraints have been set. The user can then modify them before asking Squirrel to build the result page.

4.5 Document View

The user selects a document from the reduced result set, which takes them to a view of the document itself. This shows the meta-data and text associated with the document and also a link to the source page if appropriate – as is the case with web-pages. Since web-pages are published externally with specific formatting data, the text of the page is extracted at index-time. Any semantic mark-up that is applied at this stage can then be shown on the extracted text at query-time. However, the user should always be given the option to navigate to the page in its original format. A screenshot of the document view is shown in Figure 2.


Document:Helping employees stay healthy

Abstract: About 240 big companies (which together provide health insurance for more than 50 million Americans) have banded together to create a nonprofit organization called the National Business Group on Health. It advises large employers on health-care and benefits issues and has started bestowing a new honor - the Best Employers for Healthy Lifestyles award - on companies that are inve

National Business Group, Inc. is a Company located in United States (North America). Its webpage is http://www.nbg.com. National Business Group (NBG) knows it often takes networking to get a job. The systems integrator specializes in LANs, WANs, remote access, and network security. NBG also resells networking and connectivity equipment and offers network analysis, consulting, design, installation, technical support, and training services. The company purchases from such suppliers as Cisco Systems, and Microsoft, and Nortel Networks. NBG's customers include businesses in industries such as education, finance, manufacturing and technology as well as Federal, state and local governments. CEO Richard Basich, who owns National Business Group, founded the company in 1987 with Jeff Malone, who heads the company's sales efforts. Contact on: http://www.nbg.com.

Author: Fis

Topics: Em xpenditures;

Date Published: Aug

Full Text: EVE are in the U.S. and of how worried em e seen the fallout close to home, in the form nd copayments. But corporate America is fina (which together provide health insurance for r to create a nonprofit organization called the mployers on health-care and benefits issu Employers for Healthy Lifestyles award-on con thier.

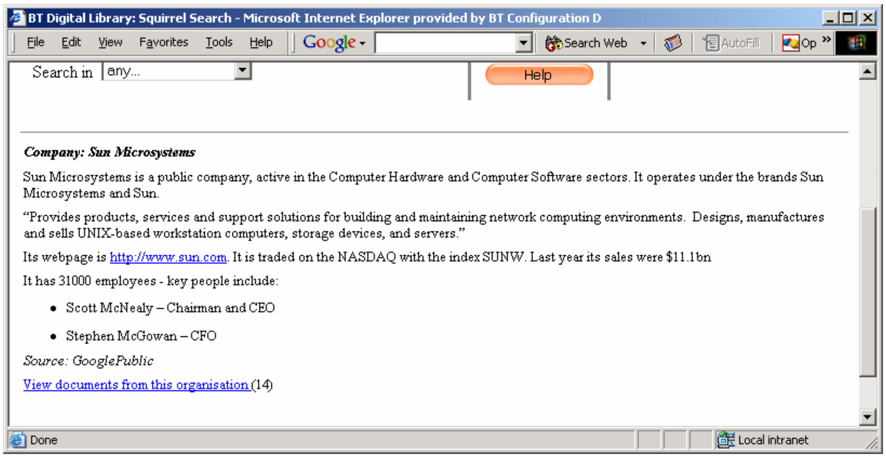
Some of the winning tactics include installing walking routes and hiking paths around workplaces and stocking cafeterias and vending machines with more fruit and other foods containing less fat and salt. NBGH hopes that highlighting such best practices will encourge other employers to copy them. Additional praised tactics:

Fig. 2. Document View

The document view also shows the whole document abstract, marked-up with entity annotations. ‘Mousing-over’ these entities provides the user with further information about the entity extracted from the ontology. Clicking on the entity itself takes the user to the entity view.

4.6 Entity View

The entity view for ‘Sun Microsystems’ is shown in Figure 3. It includes a summary generated by the NLG Web Service described in Section 3.4. The summary displays information related not only to the entity itself but also information about related


BT Digital Library: Squirrel Search - Microsoft Internet Explorer provided by BT Configuration D
File Edit View Favorites Tools Help
Google
Search Web
AutoFill
Search in any...
Help
Done
Local intranet

Company: Sun Microsystems

Sun Microsystems is a public company, active in the Computer Hardware and Computer Software sectors. It operates under the brands Sun Microsystems and Sun.

“Provides products, services and support solutions for building and maintaining network computing environments. Designs, manufactures and sells UNIX-based workstation computers, storage devices, and servers.”

Its webpage is http://www.sun.com. It is traded on the NASDAQ with the index SUNW. Last year its sales were $11.1bn

It has 31000 employees - key people include:

- Scott McNealy – Chairman and CEO
- Stephen McGowan – CFO

Source: GooglePublic

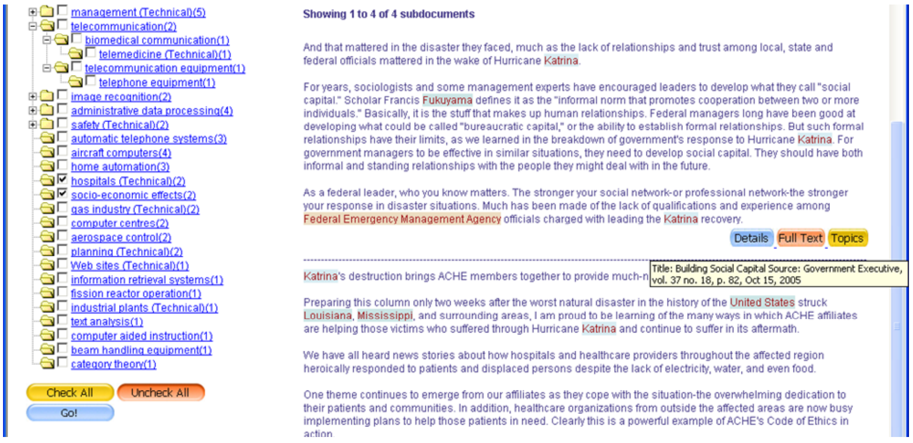
View documents from this organisation (14)

Fig. 3. Company Entity View

entities such as people who hold job roles with the company. This avoids users having to browse around the various entities in the ontology that hold relevant information about the entity in question. The relationship between people and companies is made through a third concept called JobPosition. Users would have to browse through this concept in order to find the name of the person in question.

4.7 Consolidated Results

Users can choose to view results as a consolidated summary of the most relevant parts of documents rather than a discrete list of results. This view is appropriate when a user is seeking to gain a wider view of the available material rather than looking for a specific document. The view allows them to read or scan the material without having to navigate to multiple results. Figure 4 shows a screenshot of a summary for a query for ‘Hurricane Katrina’. For each subdocument in the summary the user is able to view the title and source of the parent document, the topics into which the subdocument text has been classified or navigate to the full text of the document. A topic tree is built which includes a checkbox for each topic. These allow the user to refine the summary content by checking or unchecking the appropriate checkboxes.


management (Technical)(5)
telecommunication(2)
biomedical communication(1)
telemedicine (Technical)(1)
telecommunication equipment(1)
telephone equipment(1)
image recognition(2)
administrative data processing(4)
safety (Technical)(2)
automatic telephone systems(3)
aircraft computers(4)
home automation(3)
hospitals (Technical)(2)
socio-economic effects(2)
gas industry (Technical)(2)
computer centres(2)
aerospace control(2)
planning (Technical)(2)
Web sites (Technical)(1)
information retrieval systems(1)
fission reactor operation(1)
industrial plants (Technical)(1)
text analysis(1)
computer aided instruction(1)
beam handling equipment(1)
category theory(1)
Check All
Uncheck All
Go!
Details
Full Text
Topics

Showing 1 to 4 of 4 subdocuments

And that mattered in the disaster they faced, much as the lack of relationships and trust among local, state and federal officials mattered in the wake of Hurricane Katrina.

For years, sociologists and some management experts have encouraged leaders to develop what they call "social capital." Scholar Francis Fukuyama defines it as the "informal norm that promotes cooperation between two or more individuals." Basically, it is the stuff that makes up human relationships. Federal managers long have been good at developing what could be called "bureaucratic capital," or the ability to establish formal relationships. But such formal relationships have their limits, as we learned in the breakdown of government's response to Hurricane Katrina. For government managers to be effective in similar situations, they need to develop social capital. They should have both informal and standing relationships with the people they might deal with in the future.

As a federal leader, who you know matters. The stronger your social network-or professional network-the stronger your response in disaster situations. Much has been made of the lack of qualifications and experience among Federal Emergency Management Agency officials charged with leading the Katrina recovery.

Katrina's destruction brings ACHE members together to provide much-n

Title: Building Social Capital Source: Government Executive, vol. 37 no. 18, p. 82, Oct 15, 2005

Preparing this column only two weeks after the worst natural disaster in the history of the United States struck Louisiana, Mississippi, and surrounding areas, I am proud to be learning of the many ways in which ACHE affiliates are helping those victims who suffered through Hurricane Katrina and continue to suffer in its aftermath.

We have all heard news stories about how hospitals and healthcare providers throughout the affected region heroically responded to patients and displaced persons despite the lack of electricity, water, and even food.

One theme continues to emerge from our affiliates as they cope with the situation-the overwhelming dedication to their patients and communities. In addition, healthcare organizations from outside the affected areas are now busy implementing plans to help those patients in need. Clearly this is a powerful example of ACHE's Code of Ethics in action.

Fig. 4. Consolidated Results

5 Evaluation

Squirrel has been being subjected to a three-stage user-centred evaluation with users of BT’s Digital Library. Firstly, a heuristic evaluation [10] of the user interface was undertaken. A small group of researchers, acting as usability experts, judged whether the user interface adhered to a list of usability heuristics; a checklist was adapted from the Xerox heuristic evaluation system [3]. A number of observations were made, most of which were concerned with minor interface problems and system performance. The results of the evaluation were collated and discussed with the development team. Squirrel was then modified in accordance with these observations.

The second stage comprised a cognitive walkthrough evaluation [14]. Users were asked to use Squirrel in order to complete a number of tasks, where the user's actions and behaviours were recorded. Users were encouraged to talk through their actions and their concerns as they undertook each task, since this provided additional information about usability and the user's thought processes. At the end of each task, users were also asked to complete a short questionnaire. The findings were again discussed with the Squirrel development team in order to resolve interface issues.

The final stage involved a set of field tests. Subjects were provided with set of information seeking tasks which they were invited to complete using both the existing keyword based search system and Squirrel. They then provided feedback on the perceived quality of results and progress in search. Qualitative feedback was sought via rating forms and a Software Usability Measurement Inventory (SUMI) questionnaire.

Initial results reveal promising results in the perceived information quality (PIQ) of search results obtained by the subjects. From 20 subjects, using a 7 point scale the average (PIQ) using the existing library system was 3.99 vs. an average of 4.47 using Squirrel – an 12% increase. The SUMI assessment showed that users rate the application positively and believe that it has attractive properties, but were concerned about its performance and speed.

6 Discussion and Related Work

A number of emerging products and prototypes exist in the Search and Browse space. This section will now briefly consider how the best of these relate to the work described in this paper

The approach adopted by ISX Corporation's Concept Object Web [13] is to gather information from documents using automated and human extraction techniques and then present this to users as answers to queries rather than leaving the user to read the set of documents in the query response in order to gain the answer. The response to an initial free-text query consists of summaries of documents together with a list of instances that appear in them split into customisable categories. These instances can be selected and added to the initial query to refine the results set. The user is also able to find out more about the instances. They are shown a knowledge object, which is an aggregation of the semantic information about the entity with links to the source documents and how each fact was obtained. Users can navigate through the knowledge base by selecting the relation values of the current object. The approach is similar in many respects to that adopted by Squirrel. It provides better support for refinements based upon entities but does not allow the user to refine their search based on the topic of documents or browse a topic ontology in order to find related documents. Additional facilities offered by Squirrel include NLG and tailoring results according to a user profile.

/facet [6] adopts a navigational model or facet browsing approach where a user can navigate to heterogeneous resources by making selections on the properties of other semantically related types. This benefits users who do not have a clear idea of what they are searching for or know how it is described. Only those attribute values that will lead to at least one result being shown are made available. This ensures the user does not take any ‘blind alleys’. They can also back-up by removing previously chosen attributes from their search. One issue with facet browsing is that where there are many possible selections the interface becomes unwieldy. /facet overcomes this by allowing the user to enter terms in a textbox and by suggesting possible terms as they type. Again, only keywords that produce actual results are suggested. An impressive feature of /facet is its ability to build facet selection directly from the metadata with no configuration and at query time. Whilst /facet does not contain many of the features of Squirrel such as named entity recognition, result consolidation or NLG, the navigation approach and its ability to cope with a large number of facets with no configuration indicate how Squirrel could be extended in this fashion.

The Excalibur[4] product from Convera provides a search interface that allows the user to enter queries and then refine based on topics which tailors the results indicating that documents have been attributed to topics at index time. Further topics are offered to broaden or narrow the search which indicates that there is some sort of topic hierarchy behind the system, however, it is not clear that this extends to a full ontology akin to PROTON. The system highlights named entities in the text, but the user is not able to select these and find out more about them, browse to related entities or refine a search based on the values of properties that particular entities have. Furthermore, the lack of an ontology behind the entities makes it harder to apply rules of the nature described in Section 3.

7 Conclusion and Future Work

We have described Squirrel, a tool for searching and browsing semantically annotated textual resources. The tool provides a hybrid approach to search allowing the user to enter simple search terms and then refine their search or browse to related material through the presentation of appropriate metadata. The tool integrates a number of state-of-the-art components to provide ontology management, named entity recognition on ontology generation and classification. It also includes a set of novel features such as result consolidation, natural language generation, ontology based user profiling and device independence.

The tool has been trialled in two case studies where evaluation shows promising results in terms of Perceived Information Quality. Performance concerns were raised by users and this certainly requires further attention with respect to the choice and use of the reasoner. Whilst KAON2 is more flexible and includes relational database support, OWLIM is faster and may prove more appropriate for this application were it to be extended to provide similar database facilities.

A number of potential areas for further development include the ability to identify the relationships between entities that are discovered in the query response. For example, a query such as ‘Java Microsoft’ would match a number of different entities including both a topic and a company. The appropriate entities could be chosen based upon a combination of the popularity in the knowledge base i.e. how many documents contain them as annotations, the user’s profile and lastly an order of precedence that is specified for the domain e.g. in the digital library, if a topic is identified then this

[4] http://www.convera.com

might be chosen over an entity with the same name. In this case the topic Java and the company entity Microsoft might be chosen and as such it might be appropriate to initially show documents from the topic where annotations of the company have been identified. Similarly a query for 'Bill Gates Microsoft' would identify the person and company from the knowledge base. Here, as well as showing documents where annotations of both occur, it might be appropriate to show how they are related, which in this case is via the JobPosition instance relating the person to the company.

The display of entity results could also be improved with the use of user profiles. Where a user searches using a term that closely matches two or more entities there is a need for disambiguation i.e. predicting which of the entities the user is searching for by matching the results against the profile. For example, if a user was searching for documents written by an author called Davies (of which there are many different separate instances) the correct author can be chosen by matching the topics of the documents the individual authors have written against topics in the user's profile.

A second area for development is concerned with adopting a reduced configuration, faceted browsing approach such as that offered by /facet and described in the previous section.

Finally, there is a need for better and optimised integration between ontology management and full-text search. Currently the two are separate components and must be queried separately either sequentially (as in Squirrel) or in parallel with a subsequent intersection step. This has inherent performance issues which could be addressed by a combined approach that is able to employ optimisation techniques.

Acknowledgements. The work described in this paper was developed as part of the SEKT project, under EU funding (IST IP 2003506826). Further project details may be found at http://www.sekt-project.com/.

References

1. Bontcheva, K. 'Generating Tailored Textual Summaries from Ontologies'. Proceeding of the 2nd European Semantic Web Conference, Crete, Greece, June (2005), p. 531-545
2. Choi, F. Y. Y., :'Advances in domain independent linear text segmentation'. In Proceedings of NAACL, Seattle, USA, April, (2000), p. 26-33.
3. Christiansson, P., Heuristic Evaluation - A System Checklist. (2000), Xerox Corporation.
4. Fortuna, B.., Grobelnik, M. & Mladenic, D. :'Semi-automatic Construction of Topic Ontology', Semantics, Web and Mining, Joint International Workshop, EWMF 2005 and KDO 2005, Porto, Portugal, October 3-7, (2005)
5. Glover, T. and Davies, J.: 'Integrating Device Independence and User Profiles on the Web'. BT Technology Journal Vol 23. No. 3 July (2005).
6. Hildebrand, M., van Ossenbruggen, J. & Hardman, L. :'/facet: A Browser for Heterogeneous Semantic Web Repositories', Proceedings of the 5th International Semantic Web Conference, Athens, USA, November (2006).
7. Hustadt, U., Motik, B., Sattler, U. : Reducing SHIQ Description Logic to Disjunctive Datalog Programs. In Dubois, D., Welty, C., Williams, M.A., eds.: Proc. of the 9th Int. Conf. on Knowledge Representation and Reasoning (KR2004), Menlo Park, California, USA, AAAI Press (2004) 152–162

8. Jansen, B. J., Spink, A., and Saracevic, T. : ‘Real life, real users, and real needs: A study and analysis of user queries on the web’, Information Processing and Management. 36(2), (2000), 207-227.
9. Kiryakov, A., Ognyanov, D. & Manov, D. :‘OWLIM – a Pragmatic Semantic Repository for OWL’, In Proc. of Int. Workshop on Scalable Semantic Web Knowledge Base Systems (SSWS 2005), WISE 2005, 20 Nov, New York City, USA (2005).
10. Neilsen, J. and R. Molich, Heuristic Evaluation of User Interfaces, in Proceedings of the SIGCHI conference on Human factors in computing systems. (1992): Monterey, California, United States. p. 373 – 380.
11. 11.Popov, B., Kiryakov, A., Ognyanoff, D., Manov, D. and Kirilov, A. : ‘KIM - a semantic platform for information extaction and retrieval’, Journal of Natural Language Engineering, Vol. 10, Issue 3-4, Sep 2004, pp. 375-392, Cambridge University Press.
12. Reiter, E. & Dale, R.: ‘Building Natural Language Generation Systems’. Cambridge University Press, Cambridge, (2000).
13. Starz, J., Kettler, B., Haglich, P., Losco, J., Edwards, G. and Hoffman, . :‘The Concept Object Web for Knowledge Management’ Proceedings of the 4th International Semantic Web Conference, Galway, Ireland, November (2005).
14. Wharton, C., et al., Applying Cognitive Walkthroughs to more Complex User Interfaces: Experiences, Issues, and Recommendations, in Proceedings of the SIGCHI conference on Human factors in computing systems. (1992): Monterey, California, United States.
15. Xue, G., Zeng, H., Chan, Z., Ma, W., Zhang, H., and Lu, C. : ‘Implicit link analysis for small Web search’, SIGIR’03, July 28-August 1, (2003), Toronto, Canada.