Semantic Web Site Usage Analysis: The ORGAN system

John Garofalakis[1,2], Theodoula Giannakoudi[1,2] and Evangelos Sakkopoulos[1,2]

[1] *RA Computer Technology Institute*
Internet and Multimedia Technologies RU 5 and WestGate
N. Kazantzaki str. 26504, Greece

[2] *University of Patras*
Computer Engineering & Informatics Dept
26500 Patras, Greece

E-mail: gianakot@westgate.gr, {garofala, sakkopul}@cti.gr

ABSTRACT

In this work, a new information acquisition system is proposed and implemented, called ORGAN (Ontology-oRiented usaGe ANalysis system). ORGAN aims to enhance and ease log analysis by use of semantic knowledge. It is an application for log analysis taking advantage of semantic web technologies. ORGAN is able to offer typical statistical analysis of web usage logs taking into consideration at the same time site's underlying semantics.

Categories and Subject Descriptors

H.5.3 [**Information Systems**]: Group and Organization Interfaces: *Evaluation/methodology, Web-based interaction*

H.3.3 **[Information Search and Retrieval**]: *Information filtering-Query formulation*

General Terms

Management, Measurement, Performance, Design, Experimentation, Human Factors, Verification.

Keywords

Web Usage Mining, Knowledge Acquisition, Web site content semantic characterization, OWL ontology

1. INTRODUCTION

In this work, a new information acquisition system is proposed and implemented, called *ORGAN*. *Ontology-oRiented usaGe ANalysis system* aims to enhance and ease log analysis by use of semantic knowledge. ORGAN integrates a number of roles:

a) it facilitates traditional web usage analysis,

b) it assists the detection of domain knowledge and its assignment on a well-known domain ontology for the web site at hand and, finally,

c) it combines both of them in order to answer combined semantically enhanced queries about the web site usage.

The web site semantic attributes have resulted from the web site pages applying data mining techniques and have been annotated by an OWL ontology that depicts the domain knowledge. ORGAN constitutes an integrated and standalone system, which is available for semantic log analysis and can be utilized for any web site when combining the appropriate domain ontology (academic sites, e-shops, commercial sites etc).

2. MOTIVATION & RELATED WORK

User visits' analysis is the first step in any kind of web site evaluation procedure either when it involves re-design and reorganization or not. Generally, the process of discovering useful information from Web logs is called log mining [6] (or web usage mining). Log mining includes straightforward statistics methods, such as page access frequency, as well as more sophisticated forms of analysis, such as association rule mining, sequential pattern mining, clustering, etc. Our intention is to make a step forward beyond the available statistical analysis reports and to provide higher fidelity in the analysis results using the domain semantic underlying knowledge.

A web usage analysis tool milestone has been set in the first years of WWW conference by Pitkow and Bharat with the WebViz tool [8] where a first web usage analysis tool was presented by researchers to the global IT community. However, their log analysis could not result in the exact user paths and a client based enhancement, WebQuilt, was presented in [5] that introduced the notion of client side logging. Unfortunately such an idea could not be and it is not widely adopted. On the other hand, ORGAN is an application that tries to enhance the web usage analysis by the use of the underlying latent semantic topics in a website. The aim is to enable the site analyst to detect possible interconnected visits in "related" sections of his/her site. However, the standard analysis and mining is unable to find such relationships. To overcome the above problems, one can utilize existing taxonomies, such as hierarchical clustering of content and site directory or even enhance them further in order to enable semantically enhanced log analysis queries.

An older solution that tried to take advantage of site context and taxonomies in order to provide web site personalization is the SEWeP system [3]. However, our system involves a domain ontology in the web pages annotation process, instead of a site-specific terms taxonomy. The XML taxonomy used in [3] was especially designed for the certain web site, confining its applicability only to that web site. On the contrary, in ORGAN any OWL ontology is supported in order to be used for the web site at hand, suffice it to define relevant instances for the ontology classes. In this way, the internationally available and reliable knowledge is easily localized to the specific domain of interest. Considering the variety of ontology management tools that are available, the instances creation task is accounted as trivial.

3. ORGAN ARCHITECTURE

In this section, the ORGAN modules will be introduced. Our aim is to outline a roadmap of the available functionality and ease the readership. ORGAN analyses the usage of a web site with linchpin the web site's semantic features, as they are expressed through an OWL ontology, relevant to the thematic field of the web site. ORGAN utilizes knowledge, which is mined from three different data recourses: a) the web site content, b) the usage data and c) an ontology representative of the web site theme.

The web site content provides the semantics of its web pages and its structure, as well. The usage data contain information about the average number of hits from individual users visits to every web page in proportion to the whole set of distinct users visits. The ontology is used for the assignment of the web site content to

standard attributes of the specific field, so as to ensure homogeneous annotation of the web pages. In addition, the ontology provides adequate apprehension of the correlations between the classes and, by extension, the instances, which pertain the site, offering the possibility to the end-user to form queries relevant to web pages subject with combinative and sophisticated criteria.

ORGAN consists of two different functional parts. The first part deals with the processing of the input data (web site content and usage data), so as to build the appropriate knowledge base, that will be used from the query mechanism of the subsequent part as a repository. The initial component-which is utilized only once for a certain web site and it will be re-used only if the web site content changes- includes two independent sub-components. In the first sub-component, the functionalities that take place serially are: the web site parsing, the extraction of keywords and the assignment of the keywords to the ontology classes and instances. In the second sub-component, the log files pre-processing and the sessions' extraction are carried out. In short, the modules dealing with the raw data processing are:

- ORGAN Indexer, process extracts keywords from the web page body and from the URLs out of the web site's domain to which this page provides links.
- ORGAN cross-reference, which extracts keywords from web site pages popular cross-referrers
- ORGAN Translation, which translates greek words in the case of non English written pages
- ORGAN MetaData-Assigner, which applies appropriate mechanisms of word and phrases assignment, so as to characterize semantically the web site pages - represented by keywords sets- with ontology terms.
- ORGAN LogPro, which processes raw usage data and extracts user sessions.

The second ORGAN part, which is called ORGAN Analyzer module, constitutes the system interface for the end-user. It enables a combined analysis of semantically enriched statistical queries on the pre-processed raw data.

The system architecture is presented in the following figure:

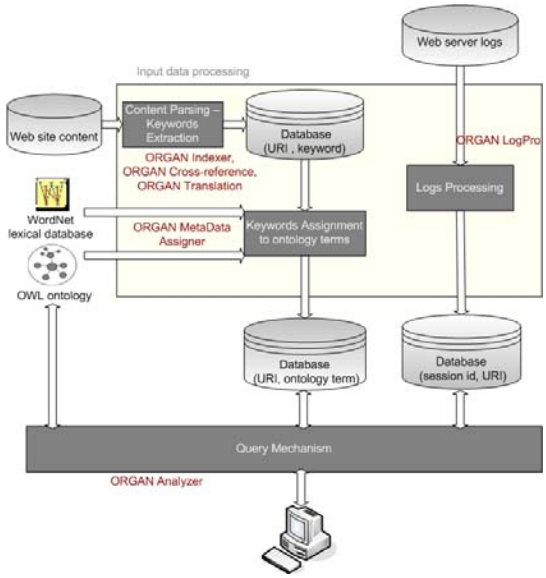

Figure 1. ORGAN Architecture

As it is shown in the figure above, the web site content processing must follow a serial execution, whereas the log files pre-processing may be carried out independently.

Next, the different modules that constitute the system are discussed in detail.

4. SITE CONTENT MANIPULATION

4.1 Keywords Extraction

During the initial web site content analysis, ORGAN follows a methodology similar to the one proposed in [3]. In this work, keywords for every web page were extracted from the web page itself, the web pages it references and the web pages that reference it. However, we refined this methodology in order to take into consideration an attribute that affects the keywords value, the web site structure itself. The web sites tested in our system and more contemporary web sites, as well, give access barely to every web site page from any node of the site. As a result, using the site pages that are referenced by the current web page to determine the page's content is not efficient in this case. It leads to the extraction of keywords that characterize the web site content in general, but not the content of the particular web page. The utilization of the web page's content, the relative content of web pages out of the site domain that are linked by the web page and the anchor text areas of web pages that reference it seem to serve the system purpose well and characterize the web page adequately.

Consequently, the set of keywords for every web site page is extracted as follows:

1. keywords are extracted from the web page's content
2. more keywords are extracted from the web pages that are referenced by the certain page and don't belong to the web site's domain
3. additional keywords are extracted from the pages, which cross-reference the specific page

Experimentation and fine tuning with ORGAN using the different web site datasets resulted in the choice of fifty keywords for each set extracted. Further fine tuning can be performed as ORGAN can be reconfigured at any keyword list length.

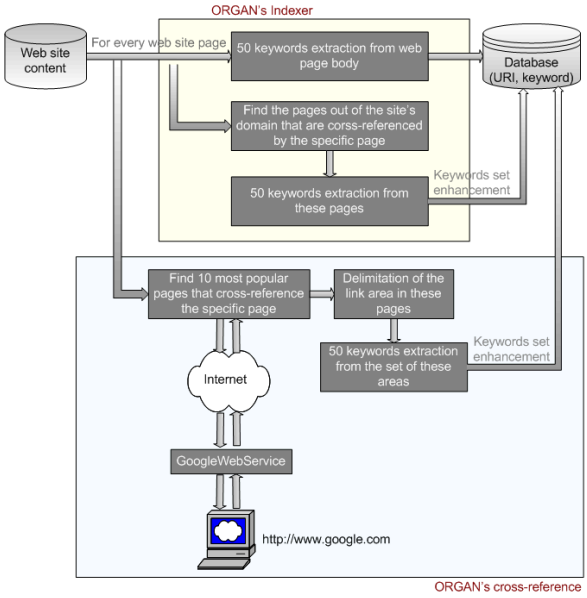

Figure 2. Keywords extraction modules architecture

The steps of the keywords extraction process are presented in Figure 3. Two distinct modules carry these tasks: the ORGAN Indexer and the ORGAN cross-reference.

4.1.1 ORGAN Indexer

In this module, the web site is parsed and for every web page the HTML tags are cleaned, the stop-words (very common words, numbers, symbols, articles) are removed, since they are considered not to contribute to the semantic denotation of the web page's content, and at last, the top 50 most frequent keywords are extracted. The set of 50 keywords that represent a web page is complemented by 50 more keywords which are extracted from the web pages that are cross-referenced by the specific page.

4.1.2 ORGAN cross-reference

Next, the second module of the keywords extraction process takes place in order to find web page references using web searching. In the case of web reference searching, Google **Error! Reference source not found.** was utilized mainly because of its web service programming interface. The first maximum 10 most popular web sites, which reference the specific page, are considered. The HTML code of each of these pages is parsed and the area around the specific link is spotted. The margins of this area are anchored 100 bytes before the link and 100 bytes after the link. The text included in that character/byte window is accounted as representative of the referenced web page subject. After removing HTML tags (and broken HTML tags from the edges) and the stop-words, the most frequent words are extracted. A set of the top 50 most frequent keywords is again kept for each web site page from the set of keywords extracted from the set of 10 web pages. Overall, after the compilation of these first two modules, appropriate keyword sets annotate every web page of the web site under analysis.

4.2 ORGAN Translation

To facilitate lexical processing, in non-English written sites web page keywords should be translated. We have used the functionality of the WordNet [7] lexical database, for words of the English vocabulary only. To achieve automation in translation we build up a web service, which posts the keywords to the Babel Fish translation engine (http://babelfish.altavista.com) and receives the translation results. In this way, the assignment process is performed. Other WordNet like solutions can be also easily incorporated such as the EuroWordNet (http://www.illc.uva.nl/EuroWordNet/) in order to achieve lexical analysis in other languages natively using a web service interface.

5. SITE CONTENT SEMANTIC CHARACTERIZATION

In this step, the content of the web site pages is related to classes and instances of an OWL ontology.

5.1 ORGAN Metadata-Assigner module

The ORGAN Metadata Assigner module uses as a criterion the semantic similarity measure between every keyword and every term of the ontology to classify the URIs to classes and instances of the ontology. Each web page is not represented by the three sets of keywords (within page keywords, referencing pages keywords and linked pages keywords) anymore, but by a very slighter terms set, which describes the web page content. Particularly, the knowledge derived from this step for a random URI is that the web page's subject concerns classes, such as "Course", "Professor" and "Sector" and especially classes instances, as "Internet Technologies", "J. Garofalakis", "Software Sector".

The calculation of the semantic similarity measure between each keyword and each ontology term was accomplished using semantic similarity measures in combination with WordNet [7]. In WordNet, English nouns, verbs, adjectives and adverbs are organized into synonym sets, each representing one underlying lexical concept. The specific measure that was applied in this system is the Wu & Palmer one. The specific measure calculates relatedness by considering the depths of the two synsets (one or more set of synonyms) in the WordNet taxonomies, along with the depth of the LCS.

$$score = \frac{2 * depth(lcs)}{(depth(s_1) + depth(s_2))}$$

This means: $score \in (0,1]$.

The score can never be zero because the depth of the LCS is never zero (the depth of the root of a taxonomy is one). The score is one if the two input synsets are the exactly same.

The assignment process is time-expensive, therefore we have implemented a caching policy to improve system response. The assignments of instances words are kept in cache, to minimize response time in case these words are met again.

As a result, the web site pages will have been assigned to relevant classes and instances of the ontology after the completion of the metadata-assigner module. All assignments will be recorded in the ORGAN local database. In the following section, we present how our tool deals with the necessary log files pre-processing procedure in order to clean the web usage trails of unnecessary non-textual information such as http errors or page icons.

6. LOG FILES PROCESSING

The preprocessing of the logs files follows two steps, as Cooley identifies them [2]: data cleaning, sessions' identification and path completion. In our case, the log files are initially cleaned from records of requests for images, requests for non informational files, such as D-HTML parts (i.e. cascade stylesheets (css) & scripting code) and requests that were not successfully responded or were submitted by search spiders & robots.

In the following, the log files are parsed and user sessions are extracted. Sessions include the set of distinct pages that were visited. For the sessions determination, three aspects are taken into consideration: the user IP, the user agent and the time interval between subsequent requests. So, the couple IP-agent defines a user, however if time between hits exceeds half an hour, it is considered that a new session started. Next, path completion is performed and the page references that are missed due to local browsing caching mechanisms are filled in. The information extracted from the log files about the user sessions is stored in ORGAN local database.

7. ORGAN ANALYZER

The ORGAN Analyzer is a knowledge analysis tool and query interface. Particularly, this ORGAN module acts as a query builder that mines logs having as a linchpin their semantics. It analyses the knowledge derived from three resources:

1. Web server raw log files
2. Database which contains metadata information of the site
3. OWL ontology with classes, instances and properties that correlate them

To facilitate knowledge acquisition functionalities, we have utilized the application programming interface of a very popular

ontology management tool, the Protégé [9]. Without lost of generality any other ontology management tool could be utilized, however several reasons presented in the following subsections drove our decision to build upon it.

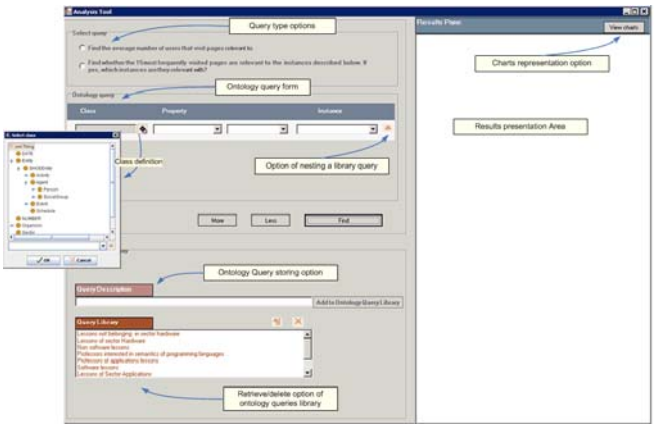

Figure 3. ORGAN Analyzer Interface

7.1 ORGAN Analyzer Interface

The ORGAN Analyzer Interface is outlined in Figure 3 and it provides access to a number of functions such as:

1 Form a semantic query (see Figure 4):

1.1 Select the statistical part queries. We call this static query part, because most web log analyzers provide a list of predefined statistical questions. However, any possible query can be built and customized to mine in a statistical sense.

1.2 Define the semantic criteria that have to be taken into consideration. We call this dynamic query part to underline the fact that any semantic notion can be chosen any time.

2 Store a query with a representative title in order to retrieve it more quickly later.

3 Recall a stored a query as a condition during the building of a new query, achieving in this way the creation of a chain of nested queries.

4 Create union queries

The queries formed through our system are of the following format:

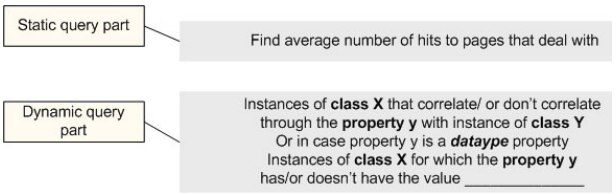

Figure 4. Query format

In particular, to submit a query, one has to select the statistical query type that needs. ORGAN supports any statistical query that can be transformed finally into a typical database SQL statement as all typical log analysis tools.

Henceforth, ORGAN delivers its new functionality compared to previous log analyzers. After selecting the query type, the instances of the pages that this query involves, can be defined in order to semantically enhance the query. The query part which is executed on the ontology may either be formed through the available form or it may be selected from a library storing favorite queries.

At this point, the capability of forming union queries has been incorporated. As a result, the user may define one or two groups of pages as criteria for the individual users' visits hits. This feature leads to the extraction of useful conclusions for the visitors' preferences. Via queries, such as "What users ratio visits web pages relevant to hardware courses and pages relevant to software courses, as well?", the user may detect the pages that don't interrelate ostensibly, but interest some users groups.

Finally, ORGAN displays the results in the right hand side of the interface pane. Intermediate results of the involved ORGAN modules can also be displayed for detailed analysis or debugging after user choice.

8. CONCLUSIONS & FUTURE STEPS

Concluding, ORGAN is an integrated tool, taking advantage of extra online service through a service-oriented architecture. WordNet-based similarity measurement, term translation, OWL ontology querying constitute a set of individual services that our system unifies, to achieve its final goal; the web site usage log semantic analysis. Future steps include keyword extraction using linguistics and incorporation of log file processing enhancement research such as the ideas presented in [1] and [4].

9. REFERENCES

[1] Christopoulou, E., Garofalakis, J., Makris, C., Panagis, Y., Sakkopoulos, E., Psaras-Chatzigeorgiou, A. and Tsakalidis, A. Techniques and Metrics for Website Reorganization. Journal of Web Engineering, 2003, 2(1-2), 90-114.

[2] Cooley, R. The use of web structure and content to identify subjectively interesting web usage patterns. ACM Transactions on Internet Technology, 2003, portal.acm.org.

[3] Eirinaki, M., Vazirgiannis, M., Varlamis, I. SEWeP: Using Site Semantics and a Taxonomy to Enhance the Web Personalization Process. In Proceedings of the 9th SIGKDD Conference, 2003.

[4] Garofalakis, J., Kappos, P. & Mourloukos, D. Web Site Optimization Using Page Popularity. IEEE Internet Computing, ,1999, 3(4): 22-29.

[5] Hong, J.I., Heer, J., Waterson, S., Landay, J.A. WebQuilt: A proxy-based approach to remote web usability testing. ACM Transactions on Information Systems, 2001, Vol 19, no 3, 263-285.

[6] Kosala, R. and Blockeel, H. Web Mining Research: A Survey. ACM SIGKDD, July 2000.

[7] Miller,GA. WordNet: A lexical database for English. Communications of the ACM, 1995, 38(11):39--41.

[8] Pitkow, J.E., Bharat, K.A. Webviz: A Tool For World-Wide Web Access Log Analysis. In Proceedings of 1st World Wide Web Conference (WWW1), Geneva, Switzerland, May 1994, 271–277. Elsevier Science BV, Amsterdam, 1994.

[9] The Protégé Ontology Editor and Knowledge Acquisition System. http://protege.stanford.edu/.