Measurements and Community Detection on the World Wide Web: The Case of the Greek Web

John Garofalakis[1,2], Christos Vavatsikos[3]

[1] Computer Engineering & Informatics Department,
University of Patras
[2] Research Academic Computer Technology Institute
garofala@ceid.upatras.gr
[3] Engineering & Computer Science Department,
Concordia University, Montreal
c_vavats@encs.concordia.ca

Abstract

The information contained in the Web pages of a national domain reflects several cultural and social characteristics of the country, maintaining at the same time a topic plurality as its contents are created by several different entities (authors, organizations, etc). The content creators often share a common interest which manifests itself as a set of related web pages, forming a Web community. The study of such communities can give useful insight into the intellectual and sociological properties of a country's Web.

In this paper we utilize the Web community enumeration techniques of [Kumar et al, 1999b]. As a case study we apply these techniques to an instance of the Greek Web consisting of 4.2 million Web pages and extract a significant number of interesting communities. Additionally, we make an in-depth analysis of several characteristics of the Greek Web (regarding its structure, its content and the most widely used technologies) and we present a visualization tool for making use of the results.

Keywords: Greek Web characterization, Web communities, Link analysis, Web mining

1. Introduction

1.1 Web Properties

The Web can be modeled as a graph whose nodes correspond to pages (or sites) and whose arcs correspond to hyperlinks between pages (or sites). The Web graph is not a random graph. The existence of a link implies a certain relation between the interlinked pages and the most popular pages attract the largest number of links. The Web graph has been found to have the properties of a scale-free network [Barabasi et al, 1999]. Scale-free networks are characterized by a certain uneven distribution of links that follows the Power Law: the probability a page p has i links is proportional

to i-θ, for a small value of θ. This is the same distribution found by G. K. Zipf for the frequency of words in texts [Zipf, 1949].

Another important property of the Web graph is its self-similarity [Dill et al, 2002]: small unified subsets of the Web have characteristics similar to the global Web's. This observation motivates our study of the Greek Web domain which is a small subset of the global Web unified by its context.

The study of the Web graph's properties can yield useful information which can be used for the improvement of crawling and retrieval techniques. On a more focused view, the study of a national Web domain can provide valuable insight into the cultural and social characteristics of a country.

1.2 Web Communities

The Web contains numerous communities. We adopt the following loose definition: A Web Community is a group of web content creators sharing a common interest which manifests itself as a set of web pages [Kumar et al, 1999b]. The study of such communities can give us a view of the sociology of the Web, its current topic trends and its content evolution, and also provide quality information sources on focused topics. Due to the Web's immense size and its continuous evolution, it is impossible to track down such communities by a manual effort.

Automatic Web community detection is an interesting application in the field of Web Mining and several related studies have been published from 1998 to today (see section 2.2). It is usually based on structure analysis and more specifically on the enumeration of certain types of subgraphs which are considered characteristic "signatures" of communities (e.g. Bipartite Cores, Cliques, Rings, etc), while in some cases the content of Web pages is used as well.

A very important development in the field of Web Mining (and its application in community detection) was the HITS algorithm [Kleinberg, 1998]. HITS analyzes the link structure of a set of pages (root set) that are considered relevant to a topic and outputs a list of "authority" and "hub" pages. Good authorities are pointed frequently by good hubs and good hubs point frequently to good authorities, thus creating a mutual reinforcement relationship. The two sets of pages can be considered as a community on the specific topic.

1.3 Guided Tour of the Paper

The rest of this paper is organized as follows: Section 2 presents the related work on the subjects of national Web domain characterization and Web community detection. Section 3 describes the main issues that arise during the process of collecting an instance of a country's Web. Section 4 presents the measurements done on the Greek Web regarding the language, structure, content and technologies of the Web pages.

Section 5 describes the methodology of community detection and the results of its application on the Greek Web. Section 6 presents our visualization tools and Section 7 concludes this work.

2. Related Work

2.1 Characterization of National Web Domains

Several studies of national Web domains have been published in the last few years. A good presentation and comparison of 12 national Web characterization studies can be found in [Beaza-Yates et al, 2005a]. The same authors have also published an in-depth study of the characteristics of the Spanish Web [Baeza-Yates et al, 2005b]. The Greek Web has been a subject of study in [Efthimiadis and Castillo, 2004] and [Baeza-Yates et al, 2004]. The first paper analyzes 3.8 millions of pages (of which about 63% is found to be written in Greek) and presents the distributions of various structure and content properties. It also studies the bow-tie structure of the graph and presents the sizes of the connected components. The second paper compares the Greek and Chilean Web between which a strong similarity is observed. An attempt to archive the Greek Web has been made by Lampos et al [Lampos et al, 2004], who collected 300,000 web pages and classified them into semantically coherent clusters in order to facilitate the searching of the archive.

2.2 Community Detection

The studies on Web community detection differ in their method, their motivation and purpose, and the unit of interaction (Web page or Web site). Some of them do not explicitly use the concept of Web community but they have a similar purpose. We divide the related work in two general categories, depending on the method used: (1) studies based on the HITS algorithm and its variations and (2) studies following a different graph-theoretic approach to the problem.

2.2.1 HITS-based Studies

Bharat and Henzinger [Bharat and Henzinger, 1998] suggest improvements for HITS in its application to the problem of Topic Distillation in the Web: given a typical user query, find quality documents related to the query topic. HITS can have poor performance in cases where the initial set of pages isn't well connected or the links are created automatically, or when the expanded root set includes many irrelevant documents. The performance gets improved (by at least 45% of precision at 10) by adding content analysis to the pure link analysis and also by reducing the influence of a single host in the computation of authority and hub weights.

Chakrabarti et al [Chakrabarti et al, 1998] describe the design, prototyping and evaluation of ARC, a system for automatically compiling a list of authoritative Web

resources on any (sufficiently broad) topic. The goal of this project is the automatic construction of a taxonomy like the manually built Yahoo! and Infoseek. The system uses an improved version of the HITS algorithm augmented with content analysis. A user study showed that ARC's resource lists are as good as the manually constructed ones. A high-level node in the resulting taxonomy can be considered as a community on the corresponding topic.

Dean et al [Dean et al, 1999] present a different application of HITS, aiming at the detection of pages that are related to a specific input page. A related page addresses the same topic as the original one. They present two algorithms that rely solely on connectivity information. The original page along with the related pages can be viewed as a community on the topic of the original page.

[Nomura et al, 2002] presents some problems of HITS, which in theory extracts topic-bound communities but in practice fails. For the understanding of the problem the authors developed a visualization tool that gives a graphical representation of the extraction process. It was observed that a large and densely linked set of unrelated Web pages in the base set (obtained with the expansion of the root set) impeded the extraction. They propose two solutions which utilize only the structural information of the Web: 1) the projection method, which projects eigenvectors on the root subspace so that most elements in the root set will be relevant to the original topic, and 2) the base-set downsizing method, which filters out the pages without links to multiple pages in the root set. These methods are shown to be robust for broader types of topics and low in computation cost.

2.2.2 Graph-theoretic Approaches

Pirolli and Pitkow [Pitkow and Pirolli, 1997] present a technique of clustering relevant web documents by exploiting their topological structure. The topology reflects the organization of a community and its knowledge base and can be extracted by doing a co-citation analysis on the web documents. The technique is a less restricted version of the HITS algorithm, which also captures the notion of co-citation in the computation of hub and authority scores.

Terveen and Hill [Terveen and Hill, 1998] consider the Web site as an ideal unit of interaction and present the notion of a clan-graph which groups relevant sites together. They also present a visualization method (Auditorium View) that reveals important structural and content properties of sites within a clan graph. Their technique for constructing clan-graphs comes from the fields of sociometrics and co-citation analysis. Beginning with a seed set of strongly connected relevant web sites, a clan graph is constructed by adding sites at a short distance from the seeds. After constructing a clan graph, they analyze it to extract additional structure and aid user comprehension.

Kumar et al [Kumar et al, 1999b] develop algorithms for the enumeration of specific subgraphs that are considered signatures of communities in the Web. An example of such a structure is a bipartite core, i.e. a small complete bipartite graph. The existence of bipartite cores corresponds not only to explicitly defined communities (such as Web Directories, News-Groups, Web-Rings, etc) but also to emerging and implicitly defined ones. The application of the enumeration process on a 1997 Alexa Web crawl (comprising of approximately 200 million pages) produced 130,000 bipartite cores which showed a strong thematic coherency. These cores were fed as root sets into the HITS algorithm and the top authorities and hubs for each core were regarded as communities. 25% of them were not represented in Yahoo!, meaning that they were emerging and had not been recognized yet.

The same authors [Kumar et al, 1999a] augment the previously described enumeration method with more subgraph signatures of tightly-focused topic communities (such as Cliques, Rings, Taxonomy Trees and Stars) in order to create knowledge bases of the Web. They build Campfire, a knowledge base of 100,000 communities, by enumerating certain subgraphs, expanding them into communities and indexing them with the application of an index term extraction algorithm on each community. The quality of the communities in the knowledge base was evaluated by manual inspection of a random sample and was found to be fairly high.

Flake et al [Flake et al, 2000] use a maximum flow/minimum cut framework to identify community members in the Web. A focused crawler that crawls to a fixed depth can approximate community membership of a node. A community is defined as a set of sites that have more links towards members of the community than towards non-members. By exploiting the properties of the web graph the problem of examining community membership for a node is reduced to the s-t maximum flow problem which can be solved in polynomial time.

Reddy et al [Reddy and Kitsuregawa, 2002] propose an approach to extract community structures in the Web that can be modeled as Dense Bipartite Graphs (DBGs). A higher level community can be considered as a DBG over a set of low-level communities, thus creating a community hierarchy. The authors extracted a three level hierarchy by using connectivity information of a TREC data set.

Toyoda and Kitsuregawa [Toyoda and Kitsuregawa, 2001] suggest a method of creating a Web community chart that interconnects relevant communities. The method is based on the Companion algorithm [Dean et al, 1999] which uses only connectivity information. A community is defined as a set of Web pages satisfying the Symmetric Derivation Relationship: each page derives the other as a related page when applying the Companion algorithm. Two communities are connected when a member of one derives a member of the other as relevant. The purpose of this project is the creation of a community chart where a user can navigate over relevant pages and communities.

In recent work [Gibson et al, 2005], Gibson et al. present a very efficient method for discovering large and dense subgraphs of massive graphs, based on a recursive application of fingerprinting via shingles. They applied this method on a host-graph of the World Wide Web containing over 50 million nodes and 11 billion edges and measured the subgraph distribution and also their evolution over time. They discovered several hundreds of giant dense subgraphs (containing over 1000 hosts) and 64 thousand dense subgraphs with less than 100 hosts. Many of these subgraphs were found to be link spam, thus the proposed method can be used as a first step in spam detection. Moreover, these types of dense subgraphs can correspond to high-quality communities that can be employed in search engines and improve their performance.

3. Web Crawling

In order to study the characteristics of a national Web domain we need to collect an instance of it, i.e. as many of the Web documents belonging to the domain as possible. We obtained our collection of the Greek Web by running the Larbin Web Crawler [Ailleret, 2004]. Larbin is a very fast and customizable crawler written in C++ and its code is under the GPL.

Some serious issues arise during the process of document collection. Larbin was designed to handle many of them but for some others it was necessary to make several modifications and additions to its code. The most important problems and the methods for handling them are described in the following paragraphs. A list of published crawler architectures along with an in-depth study of Web crawling issues can be found in [Castillo, 2004].

3.1 Network Issues

The crawler has limited available bandwidth and storage capacity so it must be designed effectively. It must also be able to handle the retrieval of large numbers of documents. It should not overload the Web and DNS servers with multiple requests and it should follow the Robots Exclusion Protocol [Koster, 1996]. Koster [Koster, 1993] suggests a waiting period of at least 30 seconds between successive requests to the same server (identified by its IP address) and also the obligatory identification of the crawler by setting a value to the User-Agent field in the HTTP request header.

3.2 HTTP Issues

A general crawler downloads only HTML and XML content and usually avoids retrieving files of different content type (e.g. mp3, zip, avi etc). This can be done by setting the appropriate MIME type to the Accept field in the HTTP request header and also by checking the Content-Type field of the server HTTP response header.

Additionally, a list of popular non-HTML extensions can be used to filter out URLs pointing to files of invalid content type.

The crawler must also be able to handle the HTTP status codes returned by the server according to the RFC 2616 protocol [W3C, 1999]. In the case of a 40X status code (client error) the crawler should mark the URL as invalid. Some servers make abuse of the 404 (page not found) status code by returning a normal page (status code 200) which includes an error message and usually some links to other pages. These pages are called "soft-404" and can mislead the crawler into considering them as normal. Identification of such pages can be done with a method described in [Bar-Yosef et al, 2004]. In the case of a 30X status code (page moved / redirection) the crawler should extract the redirection URL, identify it with the original one, and then follow it. It should also avoid getting trapped in an infinite loop by stopping following redirection URLs after a maximum number of successive redirections.

3.3 Web Content Issues

A general crawler retrieves pages only from the "publicly indexable portion" of the Web [Lawrence and Giles, 1998], neglecting a huge portion called the "hidden Web" [Raghavan and Garcia-Molina, 2001]. The latter consists of pages that a human user can access but an automatic agent cannot, like pages that require authorization or pages that are generated dynamically as the result of search actions.

While the information in the indexable Web is finite, the number of pages can be considered infinite due to the existence of dynamic pages. These include links that are generated automatically and are pointing to other dynamic pages. The crawler should restrict the number of dynamic pages it downloads from each site by either: downloading only static pages, setting a fixed upper limit to the number of dynamic pages, or downloading dynamic pages to a certain depth. The latter is the approach we followed by setting a maximum depth of 3 links for dynamic pages and 5 links for static pages for each Web site.

Cho et al [Cho et al, 2000] found that in the Web exists a very large volume of duplicate content that reaches 30% of its total size. By downloading duplicate pages, the crawler wastes bandwidth and storage space. Moreover, duplicate pages can skew the statistic measures in the Web characterization and affect the results of community detection. It is important for the crawler to be able to detect a duplicate page and identify it with the original, without following any links from it. The detection is usually done by applying a hash function on the HTML content of each page. Hashing is an efficient way of duplicate detection but it may yield false positives. Larbin implements such a mechanism.

To avoid wasting bandwidth by downloading too large files, the crawler should also set an upper limit on the pages size. We used a limit of 200 KB which is quite adequate for HTML documents.

Another important problem can be posed by pages using frames, i.e. pages consisting of multiple HTML files. A user visiting a frameset page sees one single page divided into sections corresponding to the frame pages. The crawler should detect frameset pages, download their frames pages and merge them into one single page.

Finally, a problem arises with the use of Javascript for the creation of navigational menus. Since a general crawler does not include a Javascript parser it should try to extract URLs by applying certain heuristics on the Javascript code. A similar problem arises with authors who publish the content using Flash objects. In order to extract links and content from such objects, a Flash decompiler is needed. Our crawler used a simple heuristic of extracting URLs from embedded Javascript code but did not extract links and content from Flash objects.

4. Measuring the Greek Web

Examining if a web page belongs to the national Web of a country can be a difficult task. IANA, the organization that controls the assignment of addresses and symbolic names on the Internet (http://www.iana.org/), reserves certain suffixes for each country, such as .es for Spain, .it for Italy and .gr for Greece. A first approach is to define the Greek Web as the set of Web pages whose host name is registered under the .gr domain. It is noticed though that many Web sites retain English versions of their pages as well. We define the Greek Web as follows: The Greek Web is the set of Web pages belonging to the .gr domain and some of their textual content is written in Greek language. This definition neglects a large number of Greek pages from other domains, such as .com, .net, .org but significantly reduces the crawler's total load and running time.

We obtained a sample of the Greek Web by running Larbin for seven days on a single PC with an AMD Athlon-64 3000+ processor, 1 GiB of RAM and 250 GiB of disk storage, under Suse Linux 10 64-bit. The crawling process took place in January 2006. The crawler started from the page http://www.in.gr/ (a popular Web portal) and attempted to download approximately 10 million pages. 7 million pages returned HTTP status code 200 (OK). Of these, 5.5 million were downloaded and stored while 1.5 million were duplicates. Our language identification heuristic showed that 75% of these pages were written in Greek. This resulting set of 4.2 million Web pages is the instance of the Greek Web whose properties we are analyzing in the following paragraphs.

4.1 URL Properties

The page depth measures the organization of pages in directories and sub-directories on the Web server. It is equal to the number of slashes in the URL minus one (thus a page in the root directory has depth 0). The average URL depth was found equal to 1.1 and 94% of the pages are in depth less than or equal to 3. The average URL length

(including the http:// protocol specification) for the Greek Web was found to be 65 characters.

The URL of a page indicates if the page is static or dynamic. A page can be identified as dynamic by the existence of one of the characters ?, =, & in its URL or by its file extension (php, asp, jsp or cfm). Approximately 75% of the downloaded pages were found to be dynamic while 25% were static (plain HTML). The numeric prevalence of dynamic pages over static ones can be explained by the fact that they are usually generated automatically and do not require manual effort.

4.2 Structure Properties

The Web can be viewed as a directed graph, whose nodes correspond to Web pages and whose arcs to hyperlinks between pages. Multiple arcs between two pages are merged into one and self-loops are ignored. The in-links and out-links distributions follow the power law for θ=1.76 and θ=3.32 respectively. The average in-degree and out-degree was found to be 29.3.

Besides the regular http links, pages can contain links to ftp (prefix ftp://) or e-mail addresses (prefix mailto:). We found a total number of 11,262 ftp links and 2,662,223 e-mail addresses. Both distributions follow the power law with θ=1.29 and θ=2.88 respectively.

4.3 Content Properties

The average size of HTML pages was found 31.6 KiB, while the average text size was 4.4 KiB. The size distributions are skewed, especially at the beginning of the measures, and can be approximated by two separate power law lines. The page size distributions follow the power law for θ_1=0.16 and θ_2=3.28. For the text size distributions we found θ_1=0.86 and θ_2=3.28.

The crawler discovered approximately one million links (which may contain duplicates) to documents with non-HTML content. We divide them into four categories, depending on their file extension: Images (jpg, gif, bmp, png, etc), Text Documents (txt, doc, pdf, ps, etc), Videos (avi, mov, mpeg, etc) and Sounds (wav, mp3, asf, wma etc). We found that documents and images comprise the largest part (91%) of the Greek Web's content.

4.4 Technologies

The file extension of a dynamic Web page can give us information about the server side programming language used to create it. Plain HTML pages have the extensions .htm, .html, .xhtml, .dhtm, dhtml extensions and comprise the 19.59% of total pages. PHP was found to be the most popular programming language in the Greek Web (42.89% of total pages).

Next we study the use of some popular client-side programming languages and tools such as Javascript, Cascading Style Sheets, Flash, Java and RSS. In Figure 1 we can see that CSS and Javascript are two widely used technologies in the Greek Web.

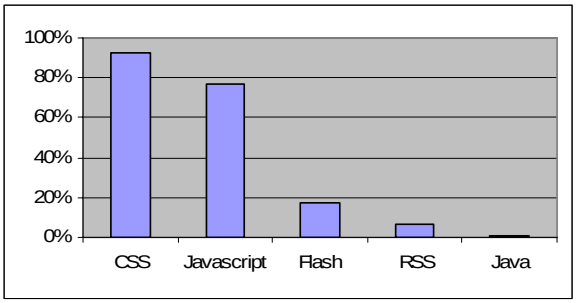

Figure 1. *Most popular technologies used in the Greek Web.*

5. Community Detection

For the detection of communities on the Greek Web we used a method based on the algorithm of Kumar et al [Kumar et al, 1999b]. We use graph structures derived from the basic hub-authority linkage patterns as the "signature" of a community and systematically locate such structures in the Web graph. It is considered that thematically cohesive Web communities contain a strongly connected core of hubs and authorities, a pattern characteristic of not only established but of emergent communities as well. On a graph-theoretic level this pattern corresponds to a small complete directed bipartite graph, which can be partitioned into two sets of nodes (Fans and Centers) such that every link in the graph is directed from a node in Fans to a node in Centers (Figure 2), although links between fans or between centers can exist as well.

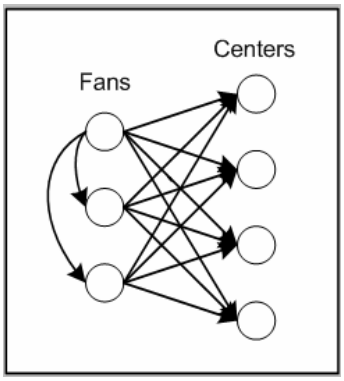

Figure 2. *A (3,4) Bipartite Core.*

5.1 Core Enumeration

Our Web graph consisted of 4.2 million nodes and 120 million edges. We consider a (3, 4) bipartite core as the smallest community "signature", since smaller subgraphs may not qualify as meaningful structures. We tried to enumerate all the (3, j) bipartite subgraphs, with $j \geq 4$ and the 3 fans belonging to different hosts (non-nepotistic cores). The latter is a very important restriction which can reduce the search space significantly. [Kumar et al, 1999b] apply this restriction as a filter after the enumeration. We apply it during the enumeration process, thus significantly reducing the number of potential and of generated cores. As the exhaustive search of the solution space is infeasible and has exponential running time, we first reduce the graph's size by applying a series of pruning steps:

Step 1. Initially, we filter out nodes that are not good candidates for fans or centers in a bipartite core. We consider as potential fans only the pages that have links to at least 3 pages of different Web sites. We also ignore very popular pages (the 200 pages with the highest PageRank) as potential centers because they tend to infiltrate many of the resulting communities without actually being closely related to the other members. For this step, [Kumar et al, 1999b] used the number of in-links and pruned the pages that have in-degree more than 50. We believe that PageRank provides us with a better and less crude measure of web page popularity than a threshold on in-degree. This step reduced the graph edges number to 35 million.

Step 2. Next we apply the iterative pruning algorithm. We iteratively prune each fan (center) whose out-degree (in-degree) is smaller than 4 (3), along with its outgoing (incoming) edges, since it cannot be part of any (3, 4) core. This step significantly reduced the number of edges to 3.5 million.

Step 3. In this step we apply the Inclusion-Exclusion algorithm: we examine the nodes that barely qualify as members of a (3, 4) core, i.e. the fans that have out-degree exactly equal to 4 and the centers that have in-degree exactly equal to 3. It is easy to check if such nodes are members of a core by examining the size of their neighbors' intersection. If this is the case, the algorithm outputs the core; otherwise the node (along with its incoming or outgoing edges) gets pruned. This algorithm generated 5 thousand (3, 4) cores, while 2.7 million edges were left in the remaining graph.

Step 4. After the Inclusion-Exclusion step we explicitly enumerate all the (3, j) cores using the Apriori algorithm [Agrawal and Srikanth, 1994]: starting with all (1, j) cores (i.e. the set of all nodes with out-degree $\geq$ j) we construct all (2, j) cores by examining every fan which also cites any center in a (1, j) core. Then we construct all (3, j) cores by examining every fan which also cites any center in a (2, j) core. During this final step we prune each edge that is found to belong to a (3,4) core so it cannot be part of another core. This measure (which had not been taken in [Kumar et al, 1999b])

produces cores with distinct edges and reduces the total number of generated cores, as well as the total time of computation. This last step generated 12 thousand (3, j) community cores, for $j \geq 4$.

5.2 Community Extraction & Processing

We expand each of the 15 thousand enumerated (3, j) cores into a community by adding all the nodes pointed by each of its fans, all the nodes pointing to at least 2 of its centers, and all the edges existing between them in the Web graph. The expanded graph is then fed to the HITS algorithm (with the improvements suggested in [Bharat and Henzinger, 1998]), which calculates an authority score and a hub score for each page of the graph. The extracted community consists of the set of the top 20 authorities and top 20 hubs.

After the extraction step we eliminate duplicate or near duplicate communities. For this purpose we consider a community as a set of pages and use the Jaccard coefficient $simJ = |A \cap B|/|A \cup B|$ as a comparison metric. Two communities A and B are merged into one when $simJ > 0.75$, $A \subset B$ or $B \subset A$. This step generated 5 thousand discrete communities showing that there was significant overlap between the initial sets.

The Jaccard coefficient was also used to cluster the resulted communities into groups. We consider that a community A belongs to group G if the Jaccard coefficient between A and any other community B contained in G is greater than a threshold m. We used $m = 0.4$ which produced 373 groups of communities. This grouping can facilitate the community browsing experience for a user as we will see in section 6.3.

5.3 Community Characterization

After the enumeration of cores, their expansion to communities and the organization into groups, we need to characterize each community's content. For this purpose, we use a method similar to [Kumar et al, 1999a]: We extract index terms (i.e. descriptive keywords) from the Web pages of each community. For this purpose we extract keywords describing the top 10 authorities of each community from the following sources: (1) the title of each page, (2) the Meta keywords defined in the <HEAD> tag, and (3) the anchor text of each page. The anchor text of a page A is the text in the vicinity of the href tags at the hyperlinks pointing to A and it has been found to be a very reliable source of descriptive terms. We used a window of 50 characters around the href tag and a maximum of 5 hyperlinks pointing to each authority page.

Afterwards, we filter the extracted terms with the help of a stop-word list in order to remove very common words (e.g. articles, pronouns, and words like "html", "homepage", etc). The list was constructed by parsing the text of a large random sample of 10,000 pages and taking the 500 most frequent terms as stop-words.

Next, we calculate the frequency of each term across the set of the 10 authorities and return the top 10 most frequent words as descriptive for the community. The frequency calculation is based on each word's stem, disregarding the suffix (if there is one). To achieve this we use a Greek stemming algorithm [Ntais, 2006].

5.4 Community Evaluation

For the evaluation of the communities generated by the system we took a random sample of 150 communities, each belonging to a different group. We then manually examined each one's top 10 authority pages, since authorities contain the most significant content. The evaluation was based on the following two criteria: (1) Community Reliability, which is high when there is strong thematic coherency between the members, and (2) Community Quality, which is high when most of the members belong to different hosts (non-nepotistic communities) and their topic is clear and focused. Each community was graded in the scale 0 – 10. The evaluation gave an average of 6.3 / 10 which is fairly high for the result of a purely automated process.

We observed that the community reliability and quality decreases in the following cases: (1) for communities whose members are popular pages of too general content (e.g. Portals, Newspapers, etc), (2) for communities whose pages are connected through advertising links or by the fact that they have been built and are hosted by the same hosting company, and (3) for communities deriving from the strong interconnection between dynamic pages of on-line shops. Communities of the first and third kind can be filtered by restricting the maximum number of authorities (besides the number of hubs) that can belong to the same site. The improved version of HITS is lowering the scores of the pages in the second case. In general, the quality and reliability improves if we focus only on the pages with the highest authority and hub scores and ignore the rest.

5.5 Community Graph

In order to relate the detected communities of the Greek Web we constructed the Community Graph as follows: a community C1 links to a community C2 when there exist at least k links from the top m authority or hub pages of C1 pointing to the top m authority or hub pages of C2, with the interlinked pages belonging to different hosts. The values of k and m define the link density of the Community Graph. Using a low value for m, we consider links between pages of higher quality, while using a high value for k we demand many links to exist between these pages for a link between the communities to exist. We used the values m=5 and k=3 after experimentation.

6. Visualization

Besides studying aggregate properties of the Greek Web, which are mainly of statistical interest, the results of our analysis can be better exploited and understood by the use of a visualization interface to the data. For this purpose we constructed two interconnected visualization interfaces: the Web Graph interface and the Community Graph interface. These interfaces have the form of dynamic Web pages which interact with a database where all the collected data is stored.

6.1 Web Graph Interface

The Web Graph interface allows a user to see a small subset of the Web Graph around a specific page/node, which consists of the set of its in and out neighbors. The user can locate the specific page by searching for a string in its URL. Besides viewing the page's attributes (e.g. size, number of images, character set, etc), the user can also navigate the Web Graph by following one link at a time, forwards or backwards. The in and out links can be ordered by PageRank or alphabetically and can be further filtered to internal or external links only.

6.2 Community Graph Interface

The Community Graph interface provides a tool for the visualization of the Greek Web's communities, their members, the topics they address and the relations existing between them. The interface is similar to the previously described Web Graph interface but in this case the unit of interaction is the Web Community. The initial page of the interface lists the 373 groups of communities along with the most frequent keywords describing their content. A user can start navigating the Community Graph either by selecting a group of interest, or by searching for a specific keyword in the describing terms. For each community the user can see the URLs and titles of the top authority and hub pages (along with the 10 keywords describing the community's content) and also navigate the Community Graph by following in or out links to other communities. Following this process, the user can gain valuable insight regarding the topic trends and the sociological properties of the Greek Web. An instance of the interface can be seen in Figure 1 of the Appendix.

7. Conclusions & Future Work

We studied a large collection of over 5 million documents of the Greek Web, contained in more than 50 thousand hosts of the .gr domain. We found that most of the properties of Web pages and sites are following the power law, verifying the observation that the Web is a self-similar and scale-free network. During the crawling and analysis process we encountered numerous broken links, many isolated sites and pages, a high degree of duplicate content and a few cases of Web spam. These

characteristics decrease the overall quality of the Greek Web but they are quite common in most Web samples.

We detected 5 thousand existing communities and grouped them into 373 clusters, based on their member overlap. By using the community graph visualization intreface we observed that a large fraction of the communities were governmental (such as of ministries, prefectures, municipalities, organizations and institutes) and academic (such as of universities, technical educational institutions and schools). The most popular general topics across the rest of the Greek Web communities were education, sports, finance, politics, tourism and culture. Many communities were of commercial content and had been generated due to the existence of numerous advertising links pointing to their pages.

Many issues have risen in this work and remain open for study. The collection of a larger sample of the Greek Web (including Greek pages outside the .gr domain) would yield a more representative sample. The inability of the crawler to extract links from Javascript code and Flash objects is a serious problem which decreases the completeness of the sample, especially if we consider the wide use of these technologies. Thus the inclusion of a Javascript parser and of a Flash decompiler seems essential. The method of community detection can be expanded with the use of additional "signature" graph types besides bipartite cores (such as Cliques or Rings). Moreover, more sophisticated techniques that can extract semantic content rather than frequent keywords would improve the communities' characterization. The annotation and organization of the communities into a hierarchical taxonomy would facilitate the browsing and visualization of the information. Finally, the collection of successive instances of the Greek Web and the study of changes between them would provide useful insight into the way the Greek Web is evolving in time.

References

Agrawal R., Srikanth R. (1994). Fast Algorithms for mining Association rules. In Proceedings of VLDB, Sept 1994, Santiago, Chile.

Ailleret S. (2004). Larbin crawler. http://larbin.sourceforge.net/index-eng.html. GPL software.

Baeza-Yates R., Castillo C., Efthimiadis E. (2004). Comparing the characteristics of the Chilean and the Greek Web. Technical report, Universidad de Chile.

Baeza-Yates R., Castillo C., Efthimiadis E. (2005a). Characterization of National Web Domains.

Baeza-Yates R., Castillo C., Lopez V. (2005b). Characteristics of the Web of Spain. Tech. rep., Universitat Pompeu Fabra.

Barabasi A., Albert R., (1999). Emergence of scaling in random networks, Science, 286(509).

Barford P., Bestavros A., Bradley A., Crovella M. E. (1999). Changes in Web client access patterns: Characteristics and caching implications in WWW. Special Issue on Characterization and Performance Evaluation.

Bar-Yossef Z., Broder A., Kumar R., Tomkins A (2004). Sic transit Gloria telae: towards an understanding of the web's decay. In Proceedings of the 13th conference on World Wide Web, pages 328 – 337, New York, USA.

Bharat K., Henzinger M. (1998). Improved Algorithms for Topic Distillation in a Hyperlinked Environment. Proceedings of SIGIR-98, 21st ACM International Conference on Research and Development in Information Retrieval.

Broder A., Kumar R., Maghoul F., Raghavan P., Rajagopalan S., Stata R., Tomkins A., and Wiener J. (2000). Graph structure in the Web: Experiments and models. In Proceedings of the Ninth Conference on World Wide Web. ACM Press, Amsterdam, Netherlands, 309-320.

Castillo C. (2004). Effective Web Crawling. PhD Thesis.

Chakrabarti S., Dom B., Raghavan P., Rajagopalan S., Gibson D., Kleinberg J. (1998). Automatic resource compilation by analyzing hyperlink structure and associated text. Computer Networks and ISDN Systems, 30(1--7):65--74.

Cho J., Shivakumar N., Garcia-Molina H. (2000). Finding replicated web collections. In SIGMOD.

Dean J, Henzinger M. (1999). Finding Related Pages in the World Wide Web. Computer Networks (Amsterdam, Netherlands: 1999)

Dill S., Kumar R., Mccurley K. S., Rajagopalan S., Sivakumar D., Tomkins, A. (2002). Self-similarity in the web. ACM Trans. Inter. Tech., 2(3):205–223.

Efthimiadis E., Castillo C. (2004). Charting the Greek Web. ASIST Conference (Poster), Providence, Rhode Island, USA.

Faloutsos M., Faloutsos P., Faloutsos C. (1999). On power law relationships of the internet topology, ACM SIGCOMM.

Flake G. W., Lawrence S., Giles C. L. (2000). Efficient identification of Web communities.KDD 2000:150-160

Gyongyi Z., Garcia-Molina H. (2005). Web spam taxonomy. In First International Workshop on Adversarial Information Retrieval on the Web.

Kleinberg J. (1999) Authoritative sources in a hyperlinked environment. ACM 46:604–632.

Koster M. (1993). Guidelines for robots writers. http://www.robotstxt.org/wc/guidelines.html.

Koster M. (1996). A standard for robot exclusion. http://www.robotstxt.org/wc/exclusion.html.

Kumar R., Raghavan P., Rajagopalan S., Tomkins A. (1999a). Extracting Large-Scale Knowledge Bases from the Web. VLDB 1999: 639-650

Kumar R., Raghavan P., Rajagopalan S., Tomkins A. (1999b). Trawling the Web for Emerging Cyber-Communities. Computer Networks (Amsterdam, Netherlands)

Lampos C., Eirinaki M., Jevtuchova D., Vazirgiannis M. (2004). Archiving the Greek Web. In Proceedings of the 4th International Web Archiving Workshop (IWAW04), September 2004, Bath, UK

Lawrence S., Giles C. L. (1998). Searching the World Wide Web. Science,:98–100.

Nomura S, Oyama S., Hayamizu T., Ishida T. (2002). Analysis and Improvement of HITS Algorithm for Detecting Web Communities.

Ntais G. (2006). Development of a Stemmer for the greek Language, Master thesis.

Page L., Brin S., Motwani R., and Winograd T. (1998). The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project.

Pirolli P., Pitkow J., and Rao R. (1996). Silk from a sow's ear: Extracting usable structures from the Web. Proc. ACM SIGCHI.

Pitkow J. and Pirolli P. (1997). Life, death, and lawfulness on the electronic frontier, Proc. ACM SIGCHI.

Raghavan S. and Garcia-Molina H. (2001). Crawling the hidden web. In Proceedings of the Twenty-seventh International Conference on Very Large Databases (VLDB), pages 129–138, Rome, Italy.

Reddy K., Kitsuregawa M. (2001). An Approach to Relate the Web Communities through Bipartite Graphs. WISE (1) 2001: 301-310.

Terveen L. G., Hill W. C. (1998a). Evaluating Emergent Collaboration on the Web. CSCW 1998: 355-362.

Terveen L. G., Hill W. C. (1998b). Finding and visualizing intersite clan graphs. In Proceedings of the ACM SIGCHI '98 Conference on Human Factors in Computing Systems, pages 448--455, Los Angeles, Ca.

Toyoda M., Kitsuregawa M. (2001). Creating a Web community chart for navigating related communities. Hypertext 2001: 103-112.

Toyoda M., Kitsuregawa M. (2003). Extracting evolution of web communities from a series of web archives. Hypertext 2003: 28-37.

W3C, 1999. http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html. RFC 2616 protocol.

Zipf G. K. (1949). Human Behavior and the Principle of Least Effort. Addison-Wesley.

Appendix: *Visualization Interface*

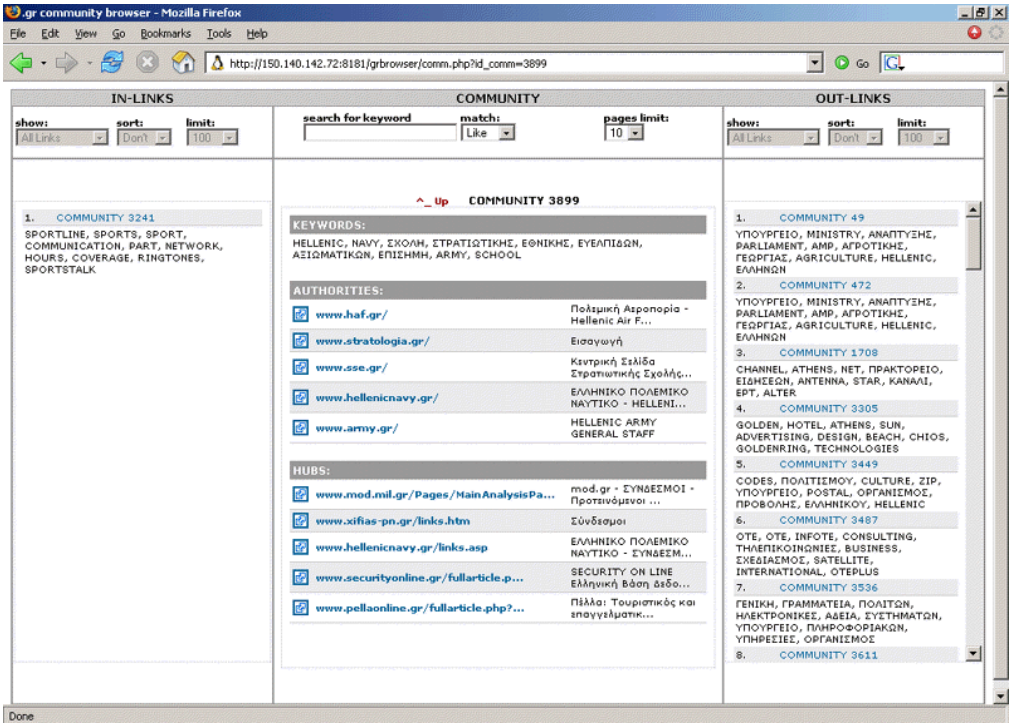

Figure 1. *Navigating the Community Graph. The focus is on a community of military schools.*