2004

TR-2004017: Towards a Formal Concept Analysis Approach to Exploring Communities on the World Wide Web

Jayson E. Rome

Robert M. Haralick

Towards a Formal Concept Analysis Approach to Exploring Communities on the World Wide Web

Jayson E. Rome and Robert M. Haralick

Department of Computer Science
The City University of New York, New York NY 10016, USA

Abstract. An interesting problem associated with the World Wide Web (Web) is the definition and delineation of so called Web communities. The Web can be characterized as a directed graph whose nodes represent Web pages and whose edges represent hyperlinks. An authority is a page that is linked to by high quality hubs, while a hub is a page that links to high quality authorities. A Web community is a highly interconnected aggregate of hubs and authorities. We define a community core to be a maximally connected bipartite subgraph of the Web graph.
We observe that a web subgraph can be viewed as a formal context and that web communities can be modeled by formal concepts. Additionally, the notions of hub and authority are captured by the extent and intent, respectively, of a concept. Though Formal Concept Analysis (FCA) has previously been applied to the Web, none of the FCA based approaches that we are aware of consider the link structure of the Web pages. We utilize notions from FCA to explore the community structure of the Web graph. We discuss the problem of utilizing this structure to locate and organize communities in the form of a knowledge base built from the resulting concept lattice and discuss methods to reduce the complexity of the knowledge base by coalescing similar Web communities. We present preliminary experimental results obtained from real Web data that demonstrate the usefulness of FCA for improving Web search.

1 Introduction

Traditional techniques for information retrieval involve text based search and various indexing methods. The presence of hyperlinks between documents presents challenges and opportunities that traditional information retrieval techniques have not had to deal with. By viewing the set of n pages on the World Wide Web as nodes V and links (similarity, association) between pages as directed edges E of a directed graph $\Gamma = (V, E)$, the graph of n nodes can be stored in an $n \times n$ matrix. A nonzero entry in the $(i, j)^{th}$ position of the matrix indicates an edge (possibly weighted or labelled) from node i to node j. A hyperlink implies some form of endorsement, or conferral of authority, by citing document to the cited document. A large portion of the current research in improving web

* To Appear : International Conference on Formal Concept Analysis (ICFCA 2005), Lens France, February 14-18, 2005

search is concerned with utilizing the hyperlinked nature of the web. Kleinberg's HITS algorithm [29], and various extensions [15], [5], [9], [10], and the Google PageRank algorithm [7], [35] demonstrate the success of link based ranking in refining Web search. Henzinger's recent survey [24] enumerates the following open algorithmic challenges for Web search researchers:

- Finding techniques to generate random samples of the Web in order to determine statistical properties of the Web,
- Modeling the web to explain observed properties,
- Detecting duplicates and near duplicates to improve search efficiency,
- Analyzing temporal trends in data streams that result from user access logs,
- Finding and analyzing dense bipartite subgraphs, or Web communities,
- Finding eigenvector-induced partitionings of directed graphs in order to cluster the Web graph.

1.1 Hubs and Authorities

Consider the problem of finding "definitive" or "authoritative" sources in the mass of information available on the web. The user should be provided with relevant pages of the highest quality. The hyperlink structure of the web contains a tremendous amount of latent information in that the creation of a link from page a to page b in some way represents a's endorsement of b. Purely text based search methods fail to find authoritative sources. For example if one uses the query "operating systems," there is no guarantee that Windows, Linux, Apple or any other operating system vendor will be among the pages returned because these pages may not explicitly contain the query terms. These pages are, however, relevant and of high quality and should in fact be returned. We can define these pages to be *authorities* because they are linked to by a large number of other pages. We can define a *hub* to be a page with a large collection of links to related pages. A good hub should point to many good authorities and a good authority should be pointed to by good hubs [29].

1.2 Web Communities

An interesting problem associated with the Web is the definition and delineation of so called Web communities [30], [31], [21], [17], [18], [16]. A *web community* is loosely defined to be a collection of content creators that share a common interest or topic and manifests itself as a highly interconnected aggregate or subgraph [30]. Kumar et al define a web community as being "characterized by dense directed bipartite subgraphs [31]." Figure 1 illustrates a simple community centered around a densely interconnected set of hubs and authorities. The World Wide Web contains many thousand explicitly defined communities and many more that are implicitly defined or are emerging [31]. The systematic extraction of emerging communities is useful for many reasons, including communities provide high quality information to interested users, they represent the sociology of the web and they can be used for target advertising [30]. In addition, community linkage can be used to find association between seemingly unconnected topics.

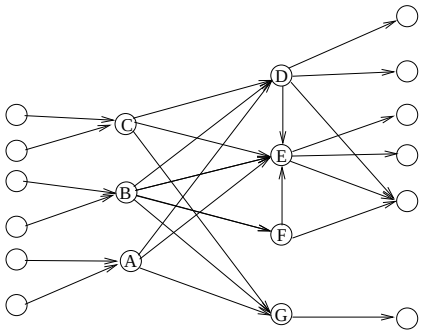

Fig. 1. A web community is characterized by a maximally complete bipartite subgraph, or diclique. In this example, nodes $\{A, B, C\}$ are hubs and nodes $\{D, E, G\}$ are authorities for a Web community.

1.3 Previous Work

Previous work on defining and delineating structures of the web graph can be roughly broken down into graph theoretic, spectral graph theoretic, distance based and probabilistic approaches.

Botafogo and Schneiderman [6] proposed a method for determining aggregates from the hyperlink structure of a small hypertext system by using the graph theoretic notions of biconnected components and strongly connected components. Flake, et al [17] describe a method for identifying Web communities based on solving for the maximal flow/minimal cut through the network. Kumar, et al [31] present a method for enumerating bipartite cores of a snapshot of the web graph.

Spectral graph theoretic methods [12] form the foundation for many link based approaches, including the Google Page Rank algorithm [35], [7] and Klienberg's HITS algorithm [29]. Klienberg uses non principle eigenvectors of matrices derived from the graph matrix to partition the graph into communities. Pirolli et al, [36] propose a procedure based on the paradigm of information foraging that "spreads activation" through the network in order to utilize both link and text information for the purposes of locating useful structures and aggregates. The method can be viewed in terms of a random walk on a weighted graph in which the nodes with the greatest activation are selected from the steady state distribution of the random process. He et al [23] describe a spectral graph partitioning method based on the Normalized Cut criterion [38].

Modha and Spangler [34] describe a method of clustering results returned by a search engine using a geometric technique based on a similarity measure that incorporates word and link information.

Almeida and Almeida [3] propose a Bayesian network approach to combine content information with user behavior to model interest based communities.

1.4 Knowledge Bases

Once we have found the communities we require a method for organizing and analyzing them. A *knowledge base* is a system for enumerating, indexing and annotating all occurrences of a specific subgraph on the web and organizing the information into a useful structure [30]. Knowledge bases are constructed by:

- Identifying a signature subgraph that is likely to characterize a specific phenomena.
- Devising a method for enumerating all occurrences of the signature subgraph.
- From each enumerated subgraph, reconstructing the associated element of the knowledge base.
- Annotating and indexing the elements of the knowledge base [30].

Reasons for building such knowledge bases include the fact that they can provide a better starting point than raw data for analysis and mining and can aid in navigation and searching. In addition, fine-grained structures can be used for targeted market segmentation and the time evolution of such structures can provide information regarding the sociological evolution of the web [31].

1.5 FCA for the Web

Though Formal Concept Analysis (FCA) has been applied to the Web, [14], [13], [27], [28], [4], [8], these approaches focus on the terms found in Web documents rather than links between documents. Kalfoglou et al [26] report using FCA to analyze program committee membership, evolution of research themes and research areas attributed to published papers. The techniques that they describe are also used to identify communities of practice [2] which are clusters of individuals defined on a weighted association network. Tilley et al [40] report results of applying FCA to the transitive closure of the citation graph within a set of survey papers. None of the approaches that we are aware of consider the context defined by the link structure of the Web graph.

We formally define a community to be a set of hub pages that link to a set of authority pages. We model a community as a maximally complete directed bipartite subgraph, or diclique [22]. Our model is similar to the bipartite cores used in [31] except that we require the cores to be maximal. This definition suffers from being too strong in that it allows no exceptions, and at the same time being too weak in that it allows communities with few members as well as communities that are defined by a single page. Our current approach to addressing these problems is to apply a post processing step in which similar concepts are coalesced together.

In most applications of FCA the sets G and M are disjoint. However, if we take both the set of objects and the set of attributes to be a set of web pages so that $G = M$, and the link matrix of the web pages to be the incidence relation I, then concepts of the context (G, M, I) correspond directly to communities of the subgraph $\Gamma = (V, E)$. For a given concept $C = (A, B)$, the extent A corresponds to the set of hub pages and the intent B corresponds to the set of authorities.

1.6 Concept Coalescing

Given a set of concepts, the concept lattice provides a convenient hierarchical description. Contexts constructed from the Web can be very large in terms of the number of nodes and dense in terms of the number of edges. It is well known that the size of the lattice grows with the size of the relation [8].

For the purposes of our investigations we are interested in reducing the complexity of the lattice by merging concepts that are in some sense similar. One approach is to look for an algebraic decomposition of the lattice [39]. Funk, et al [19] present algorithms for horizontal, subdirect and subtensorial decompositions.

In addition, homomorphisms that results from a congruence relation can be used to reduce the complexity of the lattice by coalescing of concepts, while preserving much of the underlying structure. A complete congruence relation of a concept lattice $\underline{\mathfrak{B}}(G, M, I)$ is an equivalence relation θ on $\underline{\mathfrak{B}}(G, M, I)$ such that $x_t \theta y_t$ for $t \in T$ implies $\left(\bigwedge_{t\in T} x_t\right) \theta \left(\bigwedge_{t\in T} y_t\right)$ and $\left(\bigvee_{t\in T} x_t\right) \theta \left(\bigvee_{t\in T} y_t\right)$ [20]. Define $[x]\theta = \{y \in \underline{\mathfrak{B}}(G, M, I) | x\theta y\}$ to be the equivalence class of θ that contains element x and $\underline{\mathfrak{B}}(G, M, I)/\theta := \{[x]\theta | x \in \underline{\mathfrak{B}}(G, M, I)\}$ to be the factor lattice for $\underline{\mathfrak{B}}(G, M, I)$ and congruence θ [20]. The set of all complete congruences forms a complete lattice. Ganter and Wille describe a method to construct the congruence lattice for a formal context [20].

Another approach is to utilize notions from association rule mining in order to relax the restrictions on what constitutes a concept. The confidence is a measure of the strength of a rule , while support is a measure of the statistical significance of a rule [1].

Needed Links Often the definition of the binary relation I is prone to error, and this is especially true when dealing with the web. These errors could be an error of omission, in which a pair (g, m) that should be in the relation is left out, or an error of commission, in which the pair (g, m) has erroneously been included in the relation. Omission tends to have more dramatic effect than commission [22] in that it is often easier to detect a spurious association from a given set of associations than to find a previously unknown association.

In order to collapse the lattice by the coalescing procedure we used in our experiments, we need to introduce edges to the original relation. The ability to isolate needed links automatically can be of great benefit in the case of the Web due to the fact that the Web is a volatile and dynamic environment. Additionally many communities that are forming or emerging may not know that they are, or that they should be, part of another community.

In the remainder of the paper we show an example of how we use notions from Formal Concept Analysis to explore the algebraic structure of Web communities, describe a prototype system for building web community knowledge bases using FCA and present preliminary experimental results obtained from real Web data that demonstrate the usefulness of FCA for improving Web search.

x	$I(x)$	$I^{-1}(x)$
1	2,6,9	10,11,12
2	10,11	1,3,5,7
3	2,6,9	10,11,12
4	6,9	0
5	2,6	0
6	10,12	1,3,4,5,7
7	2,6,9	0
8	10,12	0
9	10,11,12	1,3,4,7
10	1,3	2,6,8,9
11	1,3	2,9
12	1,3	6,9

Table 1. The incidence relation I

2 A Simple Example

We shall take an example first presented by Haralick [22] that is sufficiently complex to illustrate the manipulation of communities. We are given the incidence relation I, shown in tabular form in table 1 and graphical form in figure 2, which we can take to be the link structure of some subgraph of the Web. The communities of this relation are graphically enumerated in figure 3. Even with only 12 nodes and 28 links, finding these communities by inspection is nontrivial. The concept lattice, shown in figure 4, reveals the underlying structure in a relatively straightforward manner. Viewing the lattice structure is only useful for a small number of communities and for larger lattices we require a method of collapsing, or coalescing similar communities.

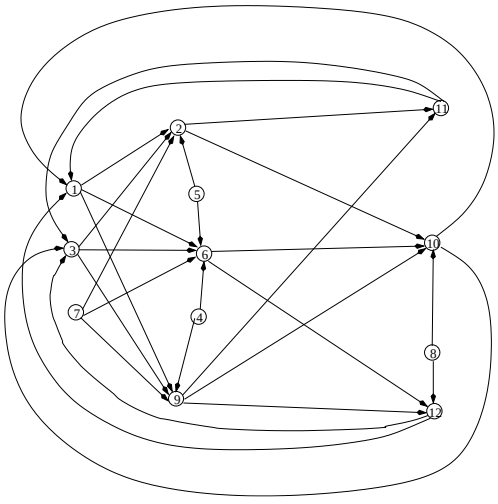

Fig. 2. The graph of binary relation I

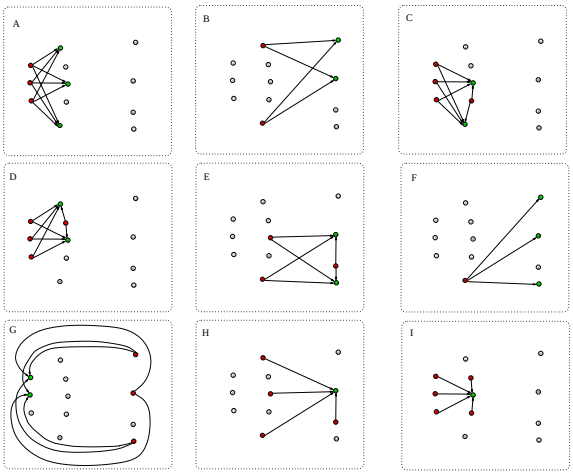

Fig. 3. A graphical enumeration of the concepts (community cores) of relation I

Coalescing finds "needed links," i.e. those links that need to be added to the relation I such that the concepts of the new coalesced relation $\hat{I}$, shown in figure 6 are themselves concepts. For example, let us again consider relation I. In order to coalesce concepts A, C, D and the concept labelled I, we need to introduce edges $(4, 2)$ and $(5, 9)$ to relation I in order to make the bipartite graph defined by $\{1, 3, 4, 5, 7\}, \{2, 6, 9\}$ complete. To coalesce concepts B, E, F, and H, we need to add $(2, 12)$, $(6, 11)$ and $(8, 11)$ to the original relation to make the bipartite graph defined by $\{2, 6, 8, 9\}, \{10, 11, 12\}$ complete.

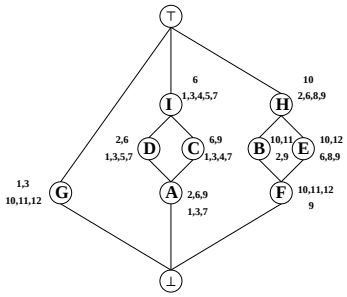

Fig. 4. The concept lattice for incidence relation I with intent (top) and extent (bottom) for each concept written out in full.

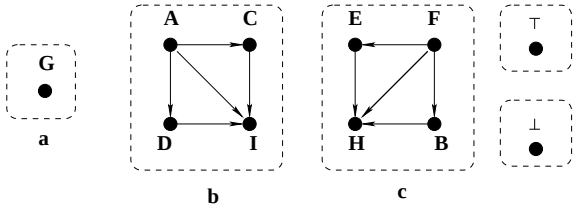

Fig. 5. A graphical representation of the relationship between concepts of I. The coalesced concepts (equivalence classes) are shown with dashed boxes.

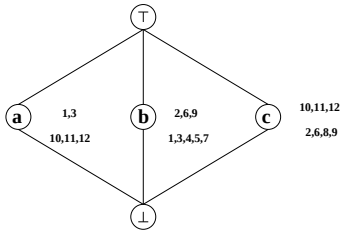

Fig. 6. The concept lattice for the coalesced binary relation $\hat{I}$ with intent and extent for each concept written out in full.

3 Experimental Setup

We built a prototype system, whose architecture is shown in figure 7, to empirically verify the effectiveness of our approach. We chose to use live data from the web, rather than a test collection like the TREC WT10g dataset or the crawls available from the Internet Archive, because it is free, easily available and supported by a large set of software tools and search services. A simple Web crawler is used to query a search engine and retrieve a set of urls that make up the nodes of the subgraph. For our experiments we used the text based search engine AltaVista. The top 20 results of the query are used to form a root set. A base set is constructed by adding those pages that point to any page in the root set and those pages which are pointed to by a page in the root set. Inbound links to page `URL` were obtained from the search engine using the `link:URL` command. This procedure can be repeated recursively up to a depth k, though the number of pages increases drastically with k. For each crawled page an index is created, the html source is stored and an entry is added into a graph that stores the link information. Text processing is used to create a descriptive feature vector for each page. These vectors are used to create summaries for each community in the knowledge base. The object set G and the attribute set M are the pages in the base set. The graph of the base set is constructed and used as the incidence relation I. The concepts and concept lattice $\underline{\mathfrak{B}}(G, M, I)$ of the context (G, M, I) are computed and the concepts are coalesced to produce a new relation

$\hat{I}$ and a new concept lattice $\underline{\mathfrak{B}}(G, M, \hat{I})$. Finally, a knowledge base is created by combining the concept lattice with the feature vectors.

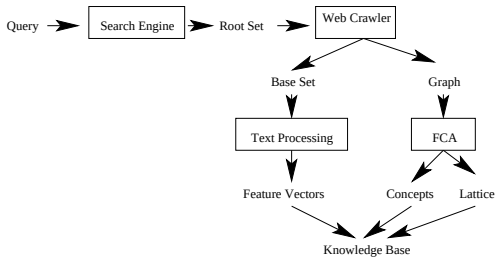

Fig. 7. System flow diagram for the prototype.

A variety of algorithms are available to compute concepts and concept lattices [20], [33], [8], [32]. Given the lattice structure of the concepts a separate procedure is used to coalesce the concepts and form a new concept lattice. Additional research needs to be done to define criterion by which an "optimal" coalescing procedure can be determined.

For the experiments presented here, we utilized a coalescing procedure similar to the horizontal decomposition described in [19]. This procedure was chosen because it is straightforward to compute and is easily understood intuitively. Recall that a lattice L is said to be *horizontally decomposable* if it can be expressed as a horizontal sum $L = \{\top, \bot\} \bigcup \sum_{i=1}^{N} L_i \setminus \{\top_i, \bot_i\}$, where the summands $L_i \cap L_j = \emptyset$ for $i \neq j$ are lattices.

We define a relation $R \subseteq \mathfrak{B}(G, M, I) \setminus \{\top, \bot\} \times \mathfrak{B}(G, M, I) \setminus \{\top, \bot\}$ as $R = \{(a, b) | \exists c, d \ni c \wedge d = a \text{ and } c \vee d = b\}$, for $a, b, c, d \in \mathfrak{B}(G, M, I) \setminus \{\top, \bot\}$. One can see that R is reflexive because $a \wedge a = a$ and $a \vee a = a$ imply $(a, a) \in R$. We define another relation $S \subseteq \mathfrak{B}(G, M, I) \times \mathfrak{B}(G, M, I)$ as $S = (R \cup R^{-1})^* \cup (\top, \top) \cup (\bot, \bot)$, where $*$ indicates the transitive closure. By construction, S is reflexive, symmetric and transitive and is therefore an equivalence relation and forms a partition of the set $\mathfrak{B}(G, M, I)$ into disjoint equivalence classes. All concepts within a given equivalence class $[x]S$ are merged into a single concept $C_{[x]S} = (\bigcup_{C_i \in [x]S} A_i, \bigcup_{C_i \in [x]S} B_i)$, where $C_i = (A_i, B_i) \in [x]S$ are the concepts in the partition. Finally a new relation $\hat{I}$ is constructed from I by $\hat{I} = I \bigcup_{x \in \mathfrak{B}(G,M,I)} \cup_{C_i \in [x]S} A_i \times \cup_{C_i \in [x]S} B_i$

We apply this approach by constructing a directed graph whose nodes are the concepts of $\mathfrak{B}(G, M, I) \setminus \{\top, \bot\}$. For every pair of concepts we connect the supremum and the infimum by a directed edge. Concepts $\top$ and $\bot$ are then added to the graph. For the relation I the graph is shown in figure 5. The connected components of this graph, shown by the dashed boxes, correspond to the equivalence classes of the equivalence relation S, and therefore represent the coalesced concepts.

3.1 The Knowledge Base

The hierarchy that results from the lattice structure forms a knowledge base that contains information about the relationships between various communities. The knowledge base is created by:

- Identifying a maximally complete bipartite subgraph (concept) to model a community core.
- Using FCA to enumerate all community cores (concepts).
- From each core, computing a representative description in the form of a feature vector.
- Using the concept lattice and the community representatives for annotating and indexing the elements of the knowledge base.

4 Experimental Results

To verify the effectiveness of the approach we performed several experiments on real Web data. Experiments were performed by posting a query to a text based search engine (in this case Alta Vista) and the base set was grown to depth $k = 1$. A summary of the results is given in table 2. The number of pages returned for a given query can be larger than what is realistically computable. Due to computational complexity, some preprocessing may be required to reduce the context to a reasonable size. The simplest method is to remove those nodes whose edge degree is below some threshold τ_I. For each query the resulting concept lattice is shown. There is clearly structure in the lattice as indicated by the number of concepts found and the density of the concept lattice.

Query	$\|G\| = \|M\|$	Number of Communities	Number Coalesced
formal concept analysis	382	43	10
support vector machine	442	51	7
ronald rivest	631	31	14
sustainable energy resources	2256	72	10
jaguar	253	41	10

Table 2. Summary of experimental results.

4.1 Community Evaluation

Community representatives are needed for the evaluation of the quality and content of a given community and can be used for annotating and indexing the knowledge base. We expect that the communities found will have some cohesiveness in terms of content. Standard approaches to clustering of text documents

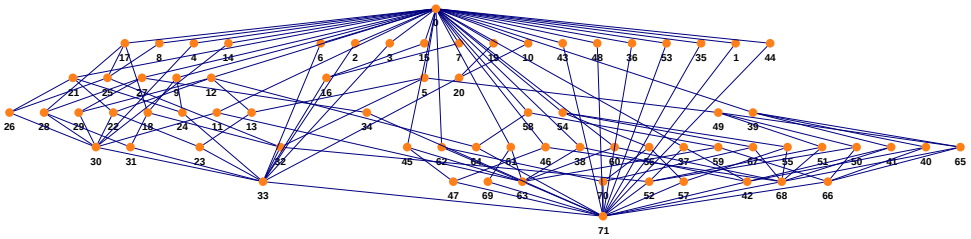

Fig. 8. The community knowledge base in the form of the concept lattice for the subgraph resulting from query *sustainable energy resources*. The base set was constructed to depth 1.

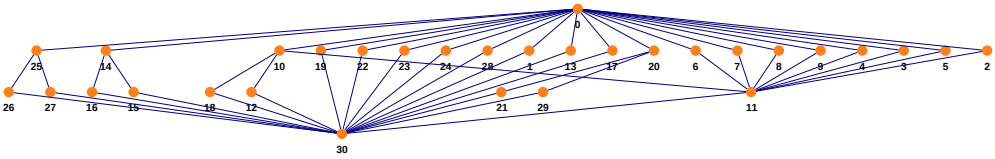

Fig. 9. The community knowledge base in the form of the concept lattice for the subgraph resulting from the query *ronald rivest*. The base set was constructed to depth 1.

involve expressing each document in terms of a feature vector and then grouping the documents into clusters based on some measure of similarity. Hotho and Stumme describe a system that uses text based clustering as a preprocessing step prior to conceptual clustering using FCA [25]. To evaluate the quality of the communities found we use the text features in a post-processing step to create descriptions of the concepts found by FCA.

Documents were processed by first extracting the text from the html documents and discarding all html markup commands. The text was broken up into a list of single words, or tokens. Stop words were removed from the token list, using a standard list of stop words. Each token was then stemmed to its root word using the Porter Stemming algorithm [37]. For each document, terms were constructed from the token list by considering all sets of words up to size s, for the purposes of the experiments described here $s = 3$. Thus, for example, the sentence "`The quick brown fox jumps over the lazy dog`" becomes "`quick brown fox jump over lazi dog`" after stemming and stop word removal and gives the features: {`quick`, `quick-brown`, `quick-brown-fox`, `brown`, `brown-fox`, `brown-fox-jump`, `...`} This procedure captures a great deal of the word interaction and semantic content of the documents. For each term t in each document d the *term frequency - inverse document frequency* $tfidf(d,t)$ is computed so that:

$$tfidf(d,t) = tf(d,t) \times log\left(\frac{|V|}{|V_t|}\right)$$

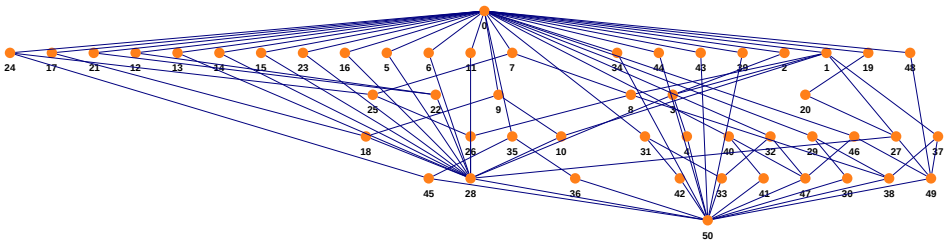

Fig. 10. The community knowledge base in the form of the concept lattice for subgraph resulting from the query *support vector machine*. The base set was constructed to depth 1.

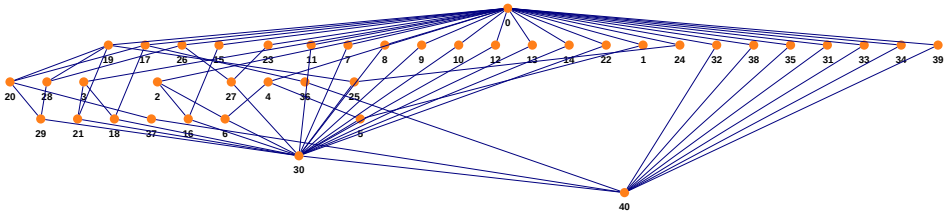

Fig. 11. The community knowledge base in the form of the concept lattice for subgraph resulting from the query *jaguar*. The base set was constructed to depth 1.

where $tf(d,t)$ is the frequency of term t in document d, V is the set of Web documents and V_t is the number of documents in which term t occurs [25].

In the prototype system community representatives are determined by computing a set of mean vectors over members of the given community. A mean vector for the object set and a mean vector for the attribute set as well as a combined mean that considers pages in both the object and attribute sets are all computed. For each community the top n features, ranked in terms of tf-idf score, from the community representatives are used to create a description of the community.

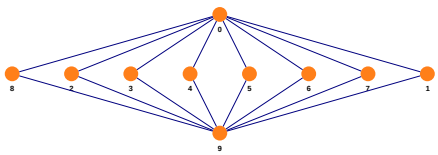

Fig. 12. The factor lattice for concept lattice resulting from the query *jaguar*.

URL
http://www.jaguar.com
http://www.jaguarcars.com
http://www.jaguar.co.uk
http://www.apple.com/macosx
http://www.jag-lovers.org
http://www.jagweb.com
http://ds.dial.pipex.com/agarman/jaguar.htm
http://www.jaguar.com.au
http://www.jaguar-racing.com/uk/flash
http://www.psgvb.com/Products/jaguar.html
http://www.jaguarmodels.com
http://www.jaguar.ca
http://www.digiserve.com/eescape/showpage.phtml?page=a2
http://www.jaguar.is
http://hem.passagen.se/isvar/jaguar_server/jserver.html
http://www.primenet.com/ brendel/jaguar.html
http://www.jec.org.uk
http://www.bluelion.org/jaguar.htm
http://www.oneworldjourneys.com/jaguar

Table 3. The urls returned from the search engine for the query *jaguar*.

4.2 Query : *jaguar*

In this section we will look at results for the query "jaguar," which has become a frequently used query for evaluating web search [11], [29], [8]. There are many reasonable answers for the query "jaguar," including Jaguar Automobiles, Animals, the Jaguar Operating System and the Atari Jaguar Game system. The urls returned from the search engine are listed in table 3. This query is an illustrative example because we expect to find many disjoint communities with interconnections between communities, documents with varied quality and with text drawn from a large vocabulary with wide variation. The concept lattice for the query is shown in figure 11 and the coalesced lattice in figure 12. Examining the top

O	OA	A
jag-lov	triumph	triumph
th-image	british	british
xj	mg	mg
find	tr	tr
cmpage	car	car
jag-lovers	usa	usa
ord	british-car	british-car
brochures	restor	restor
archives	club	club
forums	Mini	Mini

O	OA	A
atari	atari	action
tagid	action	atari
lynx	game	bit
sid	bit	game
tag	processor	processor
game	telegam	telegam
title	padport	padport
crs	hz	hz
akamai	jaguar	jaguar
referrer	arcade	arcade

Table 4. The top 10 features for *jaguar* Community 28 ($|O| = 2$ and $|A| = 2$) and Community 34 ($|O| = 1$ and $|A| = 34$)

ranked features for the representatives of each community gives us an indication of the semantic content of the community. For tables 4 through 5 O indicates the mean for the object set, A indicates the mean for the attribute set and OA indicates the mean for the combined sets. For example, the top ranked features

for concept 28, shown in table 4, indicate that the community is focused on automobiles, concept 34 in table 4 is concerned with the Atari game system, while concept 38 in table 5 deals with the Macintosh operating system. After applying

O	OA	A
mac	mac	mac
tagid	tagid	panther
tag	tag	os
os	os	tagid
sid	sid	tag
apple	apple	apple
mac-os	mac-os	sid
crs	panther	ll
akamai	crs	crs
contentgroup-wtl	akamai	akamai

O	OA	A
species	maya	maya
bluelion	cat	coat
geovisit	coat	rosett
previou	bluelion	leopard
lion	leopard	cat
image	rosett	captiv
skip	species	civil
suitabl	geovisit	jaguar
cat	captiv	differ
wild	speci	speci

Table 5. The top 10 features for *jaguar* Community 38 ($|O| = 1$ and $|A| = 29$) and CoConcept 2 ($|O| = 158$ and $|A| = 51$)

the coalescing procedure we observe that many communities remain unchanged while many communities are merged together. In the factor lattice, concept 34 gets mapped uniquely to coconcept 5 while concepts 28 and 38 get merged with many other concepts to form coconcept 2. Looking at the top ranked features for coconcept 2 shown in table 5 we see that information about the Macintosh operating system and the Jaguar automobile have been diluted by the information about the animal. So while coalescing can be a useful tool for reducing the complexity of the concept lattice, it should not be done blindly. The equivalence relation that we used in the experiments described here is very coarse. What we require is a finer coalescing procedure that considers some measure of goodness of a given congruence to select the "best" equivalence relation based on the specified criterion.

5 Conclusions

We have demonstrated the utility of Formal Concept Analysis in the problem domain of defining and delineating communities on the Web. A formal concept can be used to model a Web community, the extent of the concept corresponding to the set of hubs and the intent of the concept corresponding to the set of authorities. The lattice of communities can be used to investigate the relationships between various communities as well as provide a method of coalescing communities that are similar. The size of the contexts involved when dealing with the Web require some preprocessing. Coalescing is a powerful tool that can be used to greatly simplify the concept lattice as well as isolate needed links, though it needs to be done carefully. Additional research needs to be done to determine a criterion that can be used to determine an optimal coalescing procedure.

Special Thanks The authors would like to thank the anonymous reviewers for their insightful comments and suggestions.

References

[1] R. Agrawal, T. Imielinski, and A. N. Swami, *Mining Association Rules between Sets of Items in Large Databases*, Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data (Washington, D.C.), 1993.

[2] H. Alani, S. Dasmahapatra, K. O'Hara, and N. Shadbolt, *Identifying communities of practice through ontology network analysis*, IEEE Intelligent Systems **18** (2003), no. 2.

[3] R. B. Almeida and V. A. Almeida, *A community-aware search engine*, WWW2004, 2004.

[4] B. Berendt, A. Hotho, and G. Stumme, *Towards semantic web mining*, First International Semantic Web Conference (Sardinia, Italy), June 2002.

[5] K. Bharat and M. R. Henzinger, *Improved Algorithms for Topic Distillation in a Hyperlinked Environment*, Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval (Melbourne Australia), August 1998, pp. 104–111.

[6] R. A. Botafogo and B. Shneiderman, *Identifying Aggregates in Hypertext Structures*, Third ACM Conference on Hypertext (San Antonio, TX), 1991, pp. 63–74.

[7] S. Brin and L. Page, *The anatomy of a large-scale hypertextual Web search engine*, Computer Networks and ISDN Systems **30** (1998), no. 1–7, 107–117.

[8] C. Carpineto and G. Romano, *Concept Data Analysis : Theory and Applications*, Wiley, 2004.

[9] S. Chakrabarti, B. Dom, P. Raghavan, S. Rajagopalan, D. Gibson, and J. M. Kleinberg, *Automatic resource compilation by analyzing hyperlink structure and associated text*, Computer Networks and ISDN Systems **30** (1998), no. 1–7, 65–74.

[10] S. Chakrabarti, B. E. Dom, D. Gibson, S. Ravi Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins, *Experiments in topic distillation*, ACM SIGIR Workshop on Hypertext Information Retrieval on the Web (Melbourne, Australia), 1998.

[11] C. Chekuri, M. Goldwasser, P. Raghavan, and E. Upfal, *Web Search Using Automatic Classification*, Proceedings of WWW-96, 6th International Conference on the World Wide Web (San Jose, US), 1996.

[12] F. R. K. Chung, *Spectral Graph Theory, vol. 92 of CBMS*, American Mathematical Society, Providence, RI, 1997.

[13] R. J. Cole and P. W. Eklund, *Analyzing an Email Collection Using Formal Concept Analysis*, PKDD, 1999.

[14] ———, *Browsing Semi-structured Web Texts Using Formal Concept Analysis*, Lecture Notes in Computer Science **2120** (2001).

[15] J. Dean and M. R. Henzinger, *Finding related pages in world wide web*, Proceedings of the Eighth International World Wide Web Conference, 1999.

[16] A. Deshpande, R. Huang, Raman, T. Riggs, D. Song, and L. Subramanian, *A study of the structure of the web*, Tech. Report 284, EECS Computer Science Division, University of California, Berkeley, Berkely, CA, 1999.

[17] G. W. Flake, S. R. Lawrence, and C. L. Giles, *Efficient Identification of Web Communities*, Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (Boston, MA), August 20–23 2000, pp. 150–160.

[18] G. W. Flake, S. R. Lawrence, C. L. Giles, and F. M. Coetzee, *Self-Organization and Identification of Web Communities*, IEEE Computer **33** (2002), no. 3.

[19] P. Funk, A. Lewien, and G. Snelting, *Algorithms for Concept Lattice Decomposition and their Application*, Tech. report, TU Braunschweig, FB Informatik, 1998.

[20] B. Ganter and R. Wille, *Formal Concept Analysis : Mathematical Foundations*, Springer Verlag, Berlin – Heidelberg – New York, 1999.
[21] D. Gibson, J. M. Kleinberg, and P. Raghavan, *Inferring Web Communities from Link Topology*, Proceedings of the Ninth ACM Conference on Hypertext and Hypermedia, 1998, pp. 225–234.
[22] R. M. Haralick, *The diclique representation and decomposition of binary relations*, Journal of the ACM **21** (1974), no. 3, 356–366.
[23] X. He, C. H. Q. Ding, H. Zha, and H. D. Simon, *Automatic Topic Identification Using Webpage Clustering*, ICDM, 2001, pp. 195–202.
[24] M. R. Henzinger, *Algorithmic Challenges in Web Search Engines*, Internet Mathematics **1** (2004), no. 1, 115–126.
[25] A. Hotho and G. Stumme, *Conceptual Clustering of Text Clusters*, Proceedings FGML Workshop (Hannover), 2002.
[26] Y. Kalfoglou, S. Dasmahaptra, and J. Chen-Burger, *Fca in knowledge technologies: experiences and opportunities*, 2nd International Conference on Formal Concept Analysis (ICFCA'04), 2004.
[27] M. Kim and P. Compton, *A Web-based Browsing Mechanism based on Conceptual Structure*, The 9th International Conference on Conceptual Structures (ICCS 2001) (G. W. Mineau, ed.), 2001.
[28] ———, *Formal Concept Analysis for Domain-Specific Document Retrieval Analysis*, AI 2001: Advances in Artificial Intelligence: Australian Joint Conference on Artificial Intelligence (Adelaide, Australia), December 2001.
[29] J. M. Kleinberg, *Authoritative sources in a hyperlinked environment*, Journal of the ACM **46** (1999), no. 5, 604–632.
[30] S. Ravi Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins, *Extracting Large-Scale Knowledge Bases from the Web*, The VLDB Journal, 1999, pp. 639–650.
[31] ———, *Trawling the Web for Emerging Cyber-Communities*, WWW8 / Computer Networks **31** (1999), no. 11-16, 1481–1493.
[32] S.O. Kuznetsov and S. A. Obedkov, *Comparing Performance of Algorithms for Generating Concept Lattices*, ICCS'01 International Workshop on Concept Lattices-based KDD,.
[33] C. Lindig, *Fast Concept Analysis*, Working with Conceptual Structures - Contributions to ICCS 2000.
[34] D. S. Modha and W. S. Spangler, *Clustering Hypertext with Applications to Web Searching*, Proceedings of the eleventh ACM Conference on Hypertext and Hypermedia (San Antonio, TX USA), ACM Press, New York, US, 2000, pp. 143–152.
[35] L. Page, S. Brin, R. Motwani, and T. Winograd, *The PageRank Citation Ranking: Bringing Order to the Web*, Tech. Report Stanford Digital Libraries Working Paper SIDL-WP-1999-0120, Stanford University, 1999.
[36] P. Pirolli, J. E. Pitkow, and R. Rao, *Silk from a Sow's Ear: Extracting Usable Structures from the Web*, Proceedings of the ACM Conference on Human Factors in Computing Systems, CHI, ACM Press, 1996.
[37] M. F. Porter, *An algorithm for suffix stripping*, Program **14** (1980).
[38] J. Shi and J. Malik, *Normalized cuts and image segmentation*, IEEE Transactions on Pattern Analysis and Machine Intelligence **22** (2000), no. 8.
[39] G. Snelting and F. Tip, *Reengineering Class Hierarchies Using Concept Analysis*, Proceedings of the ACM SIGSOFT Symposium on the Foundations of Software Engineering, November 1998.
[40] T. A. Tilley, R. J. Cole, P. Becker, and P.W. Eklund, *A survey of formal concept analysis support for software engineering activities*, 1st International Conference on Formal Concept Analysis (ICFCA'03), 2003.