Association for Information Systems
AIS Electronic Library (AISeL)

PACIS 2003 Proceedings

Pacific Asia Conference on Information Systems (PACIS)

December 2003

Lattice-Based Information Retrieval Application

Karen Cheung
City University of Hong Kong

Douglas Vogel
City University of Hong Kong

Lattice-Based Information Retrieval Application

Karen S.K. Cheung[a] and Douglas Vogel[b]

[a] Department of Information Systems, City University of Hong Kong
83 Tat Chee Avenue, Kowloon, Hong Kong
iskaren@is.cityu.edu.hk

[b] Department of Information Systems, City University of Hong Kong
83 Tat Chee Avenue, Kowloon, Hong Kong
isdoug@cityu.edu.hk

Abstract

Lattice-based information representation has the advantage of providing efficient visual interface over textual display. However, the complexity of a lattice may grow rapidly with the size of the database. In this paper we formally draw the analogy between Vector Space retrieval. We then propose to use the idea of quotient lattice to reduce the complexity of a Term-Document Lattice. The equivalence relation required to construct the quotient lattice is obtained by performing Singular Value Decomposition on the original term-document matrix. We also discuss the design and evaluation of a prototype system to implement our proposed approach.

Keywords

Vector space model, information retrieval, concept lattice, quotient lattice, singular value decomposition

1. Introduction

The amount of information available to the public has been growing exponentially in recent decades. In the past few years, the World Wide Web has facilitated an explosion of informal information as well. Information is available everywhere, the potential for the retrieval of information is vast, and at times daunting.

Numerous studies suggest that graphical representation and display of searched results can improve information retrieval performance (Lin 1997, Bruza and Dennis 1997, Bruza and McArthur 2000, McArthur and Bruza 2000). In contrast to the conventional search methods such as keyword search and textual display of relevant documents, a graphical information display can provide a broad and concise representation of the searched results from which the searchers can quickly comprehend their relevance and importance. The graphical information display not only can relieve the cognitive overload of the searchers, but also it can improve the low precision and low recall of the searched results. In addition, a user-friendly graphical display enables those users who are lacking of precise information requirement to browse and navigate easily during the search process. Further similar findings have been found in a recent study (Lin 1997), in which a detailed survey is presented on how visualizations can enhance information retrieval by allowing searchers to browse through a graphical representation of the requested documents. It is therefore of utmost importance for an information retrieval system to equip with a good graphical user interface that organizes the information into an effective visual structure for the searchers to browse through during the information retrieval process.

A number of researchers have proposed the use of lattice for graphical organization and visualization structuring in the construction of information retrieval systems (Godin, Saunders & Gecsei 1986, Carpineto & Romano 1995a,1995b,1996,1998, Cole & Eklund 1996,1999, Godin, Missaoui & April 1993, Priss1997,2000a, 2000b,2000c). Lattice is a network-like classification structure that can be generated automatically from a term-document indexing relationship. Such a network structure outperforms hierarchical classification structure since the former enables many paths to a particular node while the latter restricts each node to possess only one parent. Hence lattice navigation provides an alternate browsing-based approach that can overcome the weakness of hierarchical classification browsing. However, the lattice navigation approach has its inherent problem—the curse of dimension, namely, the lattice representation of a document collection is too large to fit in a screen even for small databases. To visualize large structures, researchers have developed interfaces that allow multiple local and global views (Crouch 1990, Wille 1989). However, the method has the disadvantage that the searchers need to map different graphical representation.

Carpineto and Romano (Carpineto & Romano 1996a,1996b,2000) extended the work to adopt a variant of the fisheye view technique (Furnas 1986) to show individual nodes of the lattice on a standalone symbolic lisp machine. More recently, Kim and Compton (Kim 2001a,2001b) have developed a FCA-based browsing mechanism for dynamic clustering and for document retrieval. However, the browsing mechanism is a web-based system using a hypertext representation of the links to a node, but without providing a graphical display of the overall lattice graph. Despite these many research efforts, the display and comprehension of the lattice associated with a large database remains an open problem.

We propose in this paper an approach that incorporates the advantages of both the Vector Space Model and Concept Lattice to resolve the dimension problem of information retrieval using lattice. We introduce a formal model (the Term-Document Lattice Model) to represent a database as a lattice making use of the notion of formal concept. Then, we apply the idea of quotient lattice to reduce the complexity of a Term-Document Lattice. In the reduction process we use the method of Singular Value Decomposition (SVD) to define the equivalence relation required for the construction of quotient lattices. We propose to develop a prototype system to implement our idea. A usability study will be conducted to test and verify our proposed approach.

The remaining sections are organized as follows. In Section 2 we review the underlying principle of Vector Space Model, and Concept Lattice. The use of equivalence relationship on lattice reduction will be discussed in Section 3. We use an information retrieval example in Section 4 to show how Singular Value Decomposition can be applied to obtain a reduced Term-Document Lattice for solving the dimension problem in lattice-based information retrieval. Section 5 discusses the prototype system, and Section 6 presents the conclusion and future work.

2. Vector Space Model and Concept Lattice

The Vector Space Model (Salton et al. 1975) is well known in information retrieval. Its main idea is to represent the database as an $m \times n$ term-document matrix A (Table 1) and the query an $m \times 1$ vector q. Then matrix analysis can be performed on A and q to explore the relationships between the query and the documents. In particular the angle between the query and a document (both are $m \times 1$ vectors) is computed as a measure of how close is the two objects. Another important extension is the use of Singular Value Decomposition to obtain a low-rank approximation of A before the angle computations. This method is known as Latent Semantic Indexing (LSI) (Deerwester, Dumais & Harshman 1990) for tackling the synonymy and polysemy problems.

Term \ Document	D1	D2	D3	D4
a	1	1	0	0
b	1	0	1	0
c	0	1	0	1
d	0	0	1	1

Table 1. A 4 ´ 4 Term-Document Matrix A

The elements a_{ij} of the term-document matrix A and the elements of the query vector q may assume only one of the two values 1 and 0, with 1 (respectively 0) indicating the presence (respectively absence) of the ith term in the jth document (or in the query). Hence A is an incidence matrix. In other cases positive real numbers may be assigned to these elements to reflect the relative importance of a term in a document (or the query).

Concept Lattices stem from Formal Concept Analysis (FCA)—an area of applied mathematics that brings mathematical methods into the field of data analysis and knowledge processing. Formal Concept Analysis was first introduced by Rudolf Wille in 1980 (Davey and Priestley 1990, Ganter and Wille 1996, Grätzer 1998). FCA is based on the philosophical understanding of the world in terms of objects and attributes. It is assumed that a relation exists to connect objects to the attributes they possess. **Formal context** and **formal concept** are the fundamental notions of Formal Concept Analysis.

Definition 1 A formal context is defined as a triple (G,M,I) consisting of two sets, G and M, and a binary relation $I \subseteq G \times M$. The elements of G and M are called **objects** and **attributes** respectively. I is a relation defined between G and M. To represent an object g is in a relation I with an attribute m, we write gIm or $(g,m) \in I$.

The information presented in Table 2 gives a (limited) formal context for a crude classification of people. Here the objects are girl, woman, boy, and man. The attributes are female, juvenile, adult, and male.

Object / Attribute	girl	woman	boy	man
female	x	x		
juvenile	x		x	
adult		x		x
male			x	x

Table 2. An Example of a Formal Context

Definition 2 Let (G,M,I) be a formal context. A formal concept of (G,M,I) can be defined as an **ordered pair** (A,B) where $A \subseteq G$, $B \subseteq M$, $A' = B$, $B' = A$. We call A the **extent** and B the **intent** of the concept (A,B).

Definition 3 For $A \subseteq G$, and $B \subseteq M$, we define

$$A' := \{m \in M \mid (g,m) \in I \text{ for all } g \in A\}$$

denotes the set of attributes common to all the objects in A. Similarly,

$$B' := \{g \in G \mid (g,m) \in I \text{ for all } m \in B\}$$

denotes the set of objects possessing the attributes in B.

Definition 4 The formal concepts of a given formal context can be ordered by the **generalization-specialization** relation. If (A_1,B_1) and (A_2,B_2) are formal concepts of a formal context, then (A_1,B_1) is called a **subconcept** of (A_2, B_2) if and only if A_1 is a subset of A_2 (or equivalently, if and only if B_2 is a subset of B_1). Then (A_2,B_2) is called a **superconcept** of

(A_1,B_1), and we write (A_1,B_1) ≤ (A_2,B_2), with ≤ denotes the **hierarchical order** (or simply **order**) of the formal concepts. The set of all concepts of (G,M,I) ordered in this way is denoted by $\underline{B}(G,M,I)$and is defined as the **concept lattice** of the formal context (G,M,I).

Graphically, a concept lattice is visualized by a **Hasse diagram** (or line diagram) with **nodes** representing formal concepts and **edges** representing the subconcept-superconcept relations between formal concepts. Concept lattice allows the investigation and interpretation of relationships between concepts, objects, and attributes. Figure 1 is the concept lattice generated from the formal context in Table 2.

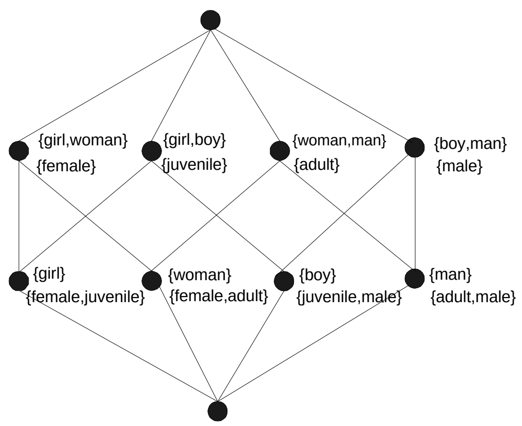


Figure 1. Concept Lattice for the Formal Context in Table 2

We interpret the lattice shown in Figure 1 in the following ways:

- Each node of the lattice is a pair composing of a subset of the objects (the upper line) and a subset of the attributes (the lower line). In each pair, the subset of attributes contains only the attributes shared by the subset of objects, and similarly, the subset of objects contains just the objects sharing the subset of attributes.
- There is an edge between two nodes if and only if they are comparable and there is no other intermediate concept in the lattice. In other words, each node is linked to its maximally specific more general nodes and to its maximally general more specific nodes.
- The ascending paths represent the subconcept/superconcept relation. The top concept contains all objects and is defined by their common attributes (possibly none), whereas the bottom concept is defined by the set of all attributes and contains no objects.

Putting Table 1 and Table 2 side-by-side we observe immediately that mathematically a term-document matrix is structurally identical to a formal context if the ones (respectively the zeros) in the former are identified with the crosses (respectively the empty boxes) of the latter. In this case, the documents and terms in the term-document matrix become objects and attributes in the formal context, respectively. Furthermore, a lattice structure is induced by the term-document matrix when the latter is considered as a formal context. We call such a lattice **Term-Document Lattice** (**TDL**). We can draw Hasse diagrams for Term-Document Lattices as concept lattices.

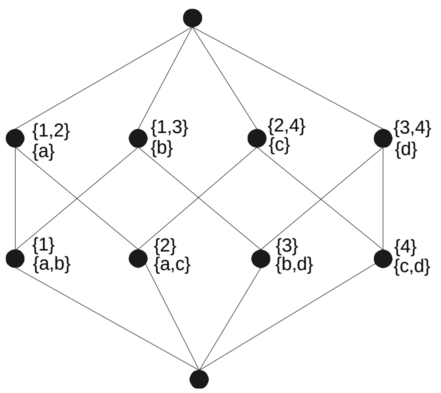
3. Lattice Reduction Using Equivalence Relationship

We introduce in this section two examples of transforming a lattice to another lattice of reduced complexity using equivalence relationship between nodes of the lattice. Given a small-sized 4×4 term-document matrix A and its associated lattice L shown in Figure 2.

Term	Document
a	D1 = *a*,*b*
b	D2 = *a*,*c*
c	D3 = *b*,*d*
d	D4 = *c*,*d*

4×4 Term-Document Matrix A

Term \ Document	D1	D2	D3	D4
a	1	1	0	0
b	1	0	1	0
c	0	1	0	1
d	0	0	1	1



Lattice *L*

Figure 2. A 4´ 4 Term-Document Matrix and Its Associated Lattice

Example 1

As discussed before, we can regard the 4×4 term-document matrix A as a formal context. Suppose that the terms *a* and *b* are considered as equivalent, we may define a new term *a*'. Furthermore, we obtain a new formal context from the old one, in which the terms *a* and *b* are replaced by *a*' (Table 3). We see, from Table 3, that although Document 2 contains *a* but not *b*, it does contain the new term *a*' because *a* and *b* are equivalent. Similarly, Document 3 contains *b* but not *a*. However, it contains *a*' because *a* and *b* are equivalent. Note that the number of rows is reduced from four to three. Associated with the new table is a new lattice K_1 as shown in Figure 3. We find that the nodes labeled with *a* and *b* collapse into a single node labeled with *a*' and the complexity of the lattice is reduced since the number of nodes and edges decrease. We also observe that K_1 retains the structure of a lattice. Mathematically, we say that K_1 is **homomorphic** to the original lattice *L*. In particular, we consider the new lattice K_1 is a **quotient lattice** of the original lattice *L*.

Term \ Document	D1	D2	D3	D4
a'	x	x	x	
c		x		x
d			x	x

Table 3. Reduced Formal Context for Example 1

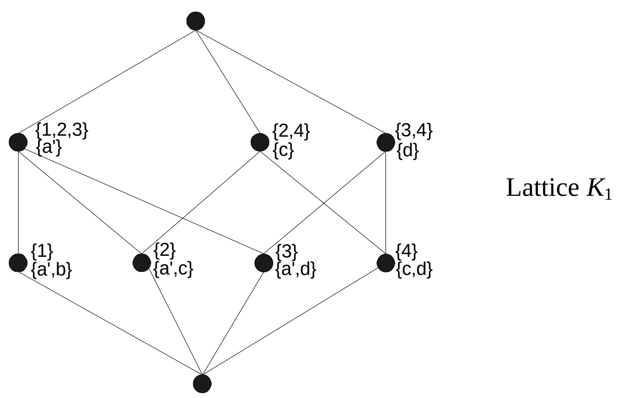


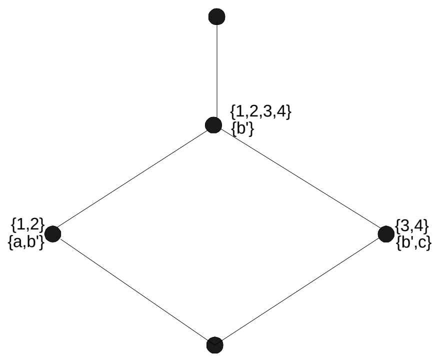
Figure 3. Concept Lattice for the Reduced Formal Context in Table 3

Example 2

In this example we will see a more significant reduction of the complexity of the lattice *L*. Suppose the terms *b* and *c* are now considered as equivalent, then we assign a new term *b*'. Table 4 shows the new formal context. Though Document 2 does not contain *b*, as *b* and *c* are equivalent, Document 2 in Table 4 does contain the new term *b*'. Similarly, Document 3 contains *b* but not *c*. However, it contains *b*' because *b* and *c* are equivalent. Again, the number of rows is reduced from four to three. Figure 4 shows the corresponding new lattice K_2 from which a significant reduction of complexity is observed.

Term \ Document	D1	D2	D3	D4
a	x	x		
b'	x	x	x	x
d			x	x

Table 4. Reduced Formal Context for Example 2



Lattice K_2

Figure 4. Concept Lattice for the Reduced Formal Context in Table 4

4. Singular Value Decomposition and Quotient Lattice

We examine in this section the use of Singular Value Decomposition to define the equivalence relation required for the construction of quotient lattices. Of particular emphasis is the significant reduction of the complexity of the Term-Document Lattice that can be achieved using this construction.

Singular Value Decomposition is one of the various matrix decomposition techniques arising from numerical linear algebra. SVD reduces both the column space and the row space of the term-document matrix to lower dimensional spaces to address the errors or uncertainties associated with the Vector Space Model (due to problems like synonymy and polysemy). The main idea of SVD is to project the very high dimensional documents and query vectors into a low dimensional space. In this new space it is reasoned that the underlying structure of the document collection is revealed thus enhancing retrieval performance. The details of SVD can be found in the standard text such as (Deerwester *et al.* 1990, Dumais, Berry & O'Brien 1995).

Our central idea is to equate clustering (using SVD) with collapse of rows of a formal context, thus introducing equivalence relations between certain nodes of the Term-Document Lattice. In the parlance of lattices, drawing equivalence relations between nodes of two lattices produces a quotient lattice with lower complexity and hence a reduced Term-Document Lattice. We use in the following a concrete information retrieval example to illustrate the idea.

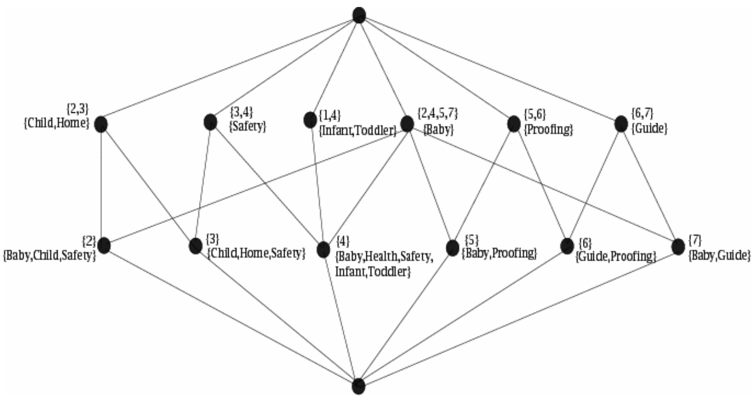
Example 3

In Berry *et al*. (1999) the use of SVD to reveal the latent relationship between documents is demonstrated. The original term-document matrix and its associated Term-Document Lattice *L* (Figure 5) are shown as follows.

Term	Document
Bab(y,ies,y's)	D1 = Infant & Toddler First Aid
Child(ren's)	D2 = Babies & Children's Room (For Your Home)
Guide	D3 = Child Safety at Home
Health	D4 = Your Baby's Health and Safety: From Infant to Toddler
Home	D5 = Baby Proofing Basics
Infant	D6 = Your Guide to Easy Rust Proofing
Proofing	D7 = Beanie Babies Collector's Guide
Safety	
Toddler	

9×7 Term-Document Matrix A

Term \ Document	D1	D2	D3	D4	D5	D6	D7
Bab(y,ies,y's)	0	1	0	1	1	0	1
Child(ren's)	0	1	1	0	0	0	0
Guide	0	0	0	0	0	1	1
Health	0	0	0	1	0	0	0
Home	0	1	1	0	0	0	0
Infant	1	0	0	1	0	0	0
Proofing	0	0	0	0	1	1	0
Safety	0	0	1	1	0	0	0
Toddler	1	0	0	1	0	0	0



Lattice *L*

Figure 5. Term-Document Lattice L for the 9 ´ 7 Term-Document Matrix A

After applying SVD to the 9×7 term-document matrix, we obtain a simplified formal context (Table 5), and the associated Term-Document Lattice (*J*) as shown in Figure 6.

Term \ Document	D1'	D5	D6	D7
Bab(y,ies,y's)	x	x		x
Child(ren's)	x			
Guide			x	x
Health	x			
Home	x			
Infant	x			
Proofing		x	x	
Safety	x			
Toddler	x			

Table 5. A Simplified Formal Context after Performing SVD

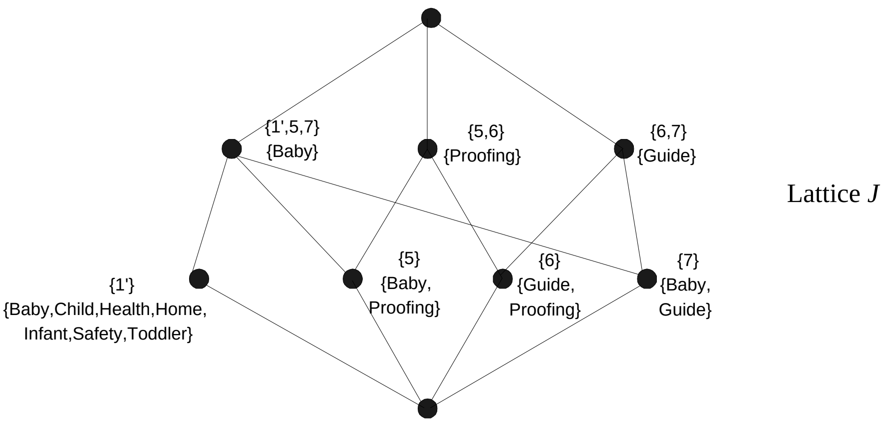
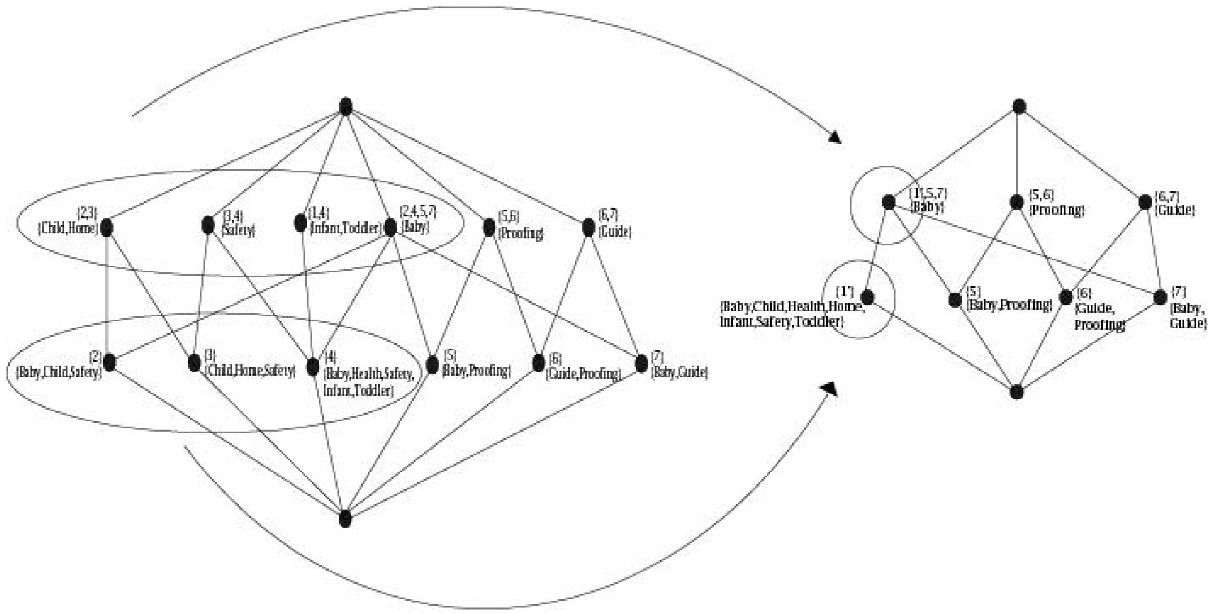


Figure 6. Term-Document Lattice J for the Simplified Formal Context in Table 5

We see that the nodes labeled with Document 1,2,3, and 4 of lattice *L* collapse into a single node labeled with Document 1'. The complexity of the lattice is reduced, and we obtain a transformed lattice *J*, which is a quotient lattice of lattice *L*. As shown in Figure 7, the congruences on *L* for the equivalence relations are depicted as shown by placing loops around the respective blocks of elements. We see that *L* and *J* are homomorphic with arrows indicating the homomorphism. Note that the equivalence relations are induced by collapsing columns of the 9×7 term-document matrix *A*.



Lattice *L* Lattice *J*

Figure 7. The Homomorphism Between the Original TDL and the Reduced TDL

5. Prototype System

We will design and develop a prototype system to implement our idea of applying Singular Value Decomposition to the Term-Document Lattice and then generate a reduced Term-Document Lattice for information retrieval.

5.1 Components of the Prototype System

The prototype system will consist of the following components:

1. Visualization Module

   The visualization module is the graphical user interface of the prototype system. To enable and facilitate the interaction between a searcher and an information retrieval system, we use lattice as the visual interface for system. A lattice visual interface can well adapted to various types of searchers with various needs. For example, more experienced users familiar with the organization of the database and locating specific information can directly specify certain combinations of terms. Casual or inexperienced users without any particular goal in mind can freely navigate browse through the lattice graph without ever submitting a query. In essence, searchers with mixed and ever-changing needs can start searching by typing in general terms, and then browse and navigate the search space from there on.

2. Browsing Module

   The browsing module is one of the interactive components of the prototoype system. The browsing mechanism we will develop is very similar to the approach of Carpineto & Romano (1996a). The browsing module takes advantage of the lattice properties. First, the fact that each node is considered to be a query (the intent) with its associated set of documents (the extent) may improve the retrieval of specific information. To be more specific, this might facilitate the searcher to recognize more useful nodes during browsing for information. Second, the network lattice structure outperforms hierarchical classification structure since the former enables many paths to a particular node while the latter restricts each node to possess only one parent. Hence lattice navigation not only provides an alternate browsing-based approach that can overcome the weakness of hierarchical classification browsing, but also facilitates recovery from bad decision making while traversing the lattice hierarchy that, in turn, might save searcher's searching time. Third, the lattice structure provides gradual enlargement or refinement of a query, and allows retrieval to be performed in an interactively way. For example, to follow the edges going upwards from a query corresponds to a minimal enlargement, and hence produces all minimal conjunctive enlargements of the query. To follow the edges going downwards from a query corresponds to a minimal refinement, and hence produces all minimal conjunctive refinements of the query.

3. Querying Module

   The prototype system provides the searchers with interactive querying function. In pure browsing mode a searcher can navigate through the lattice interface, or selects any nodes by direct graphical manipulation, that is, pointing and clicking with the mouse on the desired node. Furthermore, the querying module allows a searcher to submit his or her queries in two ways: either the searcher specifies the new keywords from scratch, or the searcher modifies the current query, that is, the intent of the current query. The result of a query is the node of the lattice (whose intent is) equal to the query, if there is any, or one of more nodes that partially match the query. The query mode will allow a user to make large jumps to regions of interest.

5.2 Prototype Implementation and Evaluation

We will conduct a usability study to validate our prototype system. The objective of the usability study is to evaluate effectiveness of a complexity reduced Term-Document Lattice retrieval method on information retrieval task. We will compare the retrieval performance of a reduced Term-Document Lattice model with that of a non-reduced Term-Document Lattice model, and that of a textual display visual interface.

We will test 30 subjects in the usability study. The subjects will be provided with a 1-hour tutorial training session and more training will be offered if needed. We will use the CISI database in our study. In our usability study we will randomly select 20 queries from the CISI database. The 30 subjects will be asked to retrieve documents relevant to the 20 queries using the three retrieval methods mentioned above. We will use a repeated-measures design, in which each subject will search each query using each retrieval method. We will use four widely used measures to evaluate the retrieval methods. They are precision, recall, search time, and number of documents retrieved.

6. Conclusion and Future Work

Graphical user interface is one of the major topics in human computer interaction, which plays an increasingly important role in how we access, retrieve, and understand information. We have addressed in the present paper the complexity problem in lattice-based information retrieval. The significance of the problem lies on the importance of constructing an effective graphical user interface for business or any other information systems. After the introduction of the Term-Document Lattice as the retrieval model we applied the notion of quotient lattice to reduce the complexity of a Term-Document Lattice. The construction of a quotient lattice requires the setting up of an equivalence relation. To obtain this relation we have proposed the application of Singular Value Decomposition to the original term-document matrix. We also described the design and evaluation of the prototype system implementing our idea.

Our present research can be extended in several directions. For example, the initial usability study may be expanded to include more systematic tests. Another possible research venue is to further study and evaluates how the performance results of our proposed approach change when the controlling factors, such as the characteristics of searchers and the database scale, vary.

References

Bruza, PD & Dennis, S (1997), ‘Query reformulation on the internet: empirical data and the hyperindex search engine’ in *Proceedings of the RIAO97 Conference - Computer-Assisted Information Searching on Internet*.

Bruza, PD & McArthur, R (2000), ‘Interactive internet search: keyword, directory and query reformulation mechanisms compared’ in *Proceedings of the 23rd Annual ACM Conference of Research and Development in Information Retrieval (SIGIR '2000),* ACM Press.

Carpineto, C & Romano, G (1995a), 'Automatic construction of navigable concept networks characterizing text databases', in Gori, M and Soda, G, eds. *Topics in Artificial Intelligence, 4th Congress of the Italian Association for Artificial Intelligence AI*IA*. Springer-Verlag, pp. 67-78.

Carpineto, C & Romano, G (1995b), 'ULYSSES: A lattice-based multiple interaction strategy retrieval interface' in Gornostaev J, Blumenthal B and Unger C, eds. *Lecture Notes in Computer Science - Human-Computer Interaction (proceedings of a conference in Berlin in 1995).* Springer-Verlag, Berlin, pp. 91-104.

Carpineto, C & Romano, G (1996a), 'Information retrieval through hybrid navigation of lattice representations', *International Journal of Human-Computer Studies*, vol. 45, pp. 553-578.

Carpineto, C & Romano, G (1996b), 'A lattice conceptual clustering system and its application to browsing retrieval', *Machine Learning*, vol. 24, pp. 95-122.

Carpineto, C & Romano, G (1998), 'Effective reformulation of boolean queries with concept lattices' in Andreasen, T, Christiansen, H and Larsen HL, eds. *FQAS '98, LNAI 1495*, Berlin, Heidelberg, pp. 83-95.

Carpineto, C & Romano, G (2000), 'Order-theoretical ranking', *Journal of the American Society for Information Science*, vol. 51, no. 7, pp. 587-601.

Cole, R & Eklund, P (1996), 'Application of formal concept analysis to information retrieval using a hierarchically structured thesaurus' in *Proceedings of a conference in Sydney in 1996*.

Cole, R & Eklund P (1999), 'Analyzing an email collection using formal concept analysis' in *Proceedings of a conference on Knowledge and Data Discovery*, No. 1704 in LNAI. Springer-Verlag.

Crouch, CJ (1990), 'An approach to the automatic construction of global thesauri', *Information Processing and Management*, vol. 26, no. 5, pp. 629-640.

Davey, BA & Priestley, HA (1990), 2nd edn, *Introduction to Lattices and Order*, Cambridge University Press, Cambridge.

Deerwester, S, Dumais, ST & Harshman R (1990), 'Indexing Semantic Indexing (LSI) Retrieval', *Journal of the American Society for Information Science*, vol. 41, pp. 391-407.

Dumais, S, Berry, MW & O'Brien,G (1995), 'Using linear algebra for intelligent information retrieval', *SIAM Review*, vol. 37, pp. 573-595.

Furnas, G (1986), 'Generalized fisheye views' in *Proceedings of ACM CHI '86 Conference on Human Factors in Computing Systems,* Boston, MA, pp. 16-23.

Ganter, B & Wille, R (1996), *Formal Concept Analysis: Mathematical Foundations*, Springer-Verlag, Berlin.

Godin, R, Saunders E & Gecsei J (1986), 'Lattice model of browsable data spaces', *Information Sciences*, vol. 40, pp. 89-116.

Grätzer, G (1998), *General Lattice Theory*, Birkhauser Verlag, Berlin.

Kim, M & Compton, P (2001a), 'A web-based browsing mechanism based on conceptual Structures' in *Proceedings of the 9th International Conference on Conceptual Structures (ICCS 2001),* Stanford University, California, USA, pp. 47-60.

Kim, M & Compton, P (2001b), 'Incremental development of domain-specific document retrieval systems' in *Proceedings of the First International Conference on Knowledge Capture (K-CAP 2001): Workshop on Knowledge Markup and Semantic Annotation,* Victoria, BC, Canada, pp. 69-77.

Lin, X (1997), 'Map displays for information retrieval', *Journal of the American Society of Information Science*, vol. 48, pp. 40-54.

McArthur, R & Bruza, PD (2000), 'The ranking of query refinements in interactive web-based retrieval' in *Proceedings of the Information Doors Workshop (Held in conjunction with the ACM Hypertext and Digital Libraries Conferences*.

Missaoui R, Godin, R & April, A (1993), 'Experimental comparison of navigation in a golis lattice with conventional information retrieval Methods', *International Journal of Man-Machine Studies*, vol. 38, pp. 747-767.

Priss, U (1997), 'A graphical interface for document retrieval based on formal concept analysis' in Santo E, ed., *AAAI Technical Report CF-97-01*, pp. 66-70.

Priss, U (2000a), 'Faceted information representation' in Stumme G, ed., *Working with Conceptual Structures (proceedings of a conference in Aachen in 2000),* Springer-Verlag, Berlin, pp. 84-94.

Priss, U (2000b), 'Knowledge discovery in databases using formal concept analysis', *Bulletin of the American Society for Information Science,* October/November, pp. 18-20.

Priss, U (2000c), 'Lattice-based information retrieval', *Knowledge Organization*, vol. 27, no. 3, pp. 132-142.

Wille, R (1989), 'Lattices and data analysis: How to draw them using a computer', *Algorithms and Order,* ed. Rival I, Kluwer, Boston, pp. 33-58.