A Hybrid Browsing Mechanism Using Conceptual Scales

Mihye Kim[1] and Paul Compton[2]

[1] Department of Computer Science Education,
Catholic University of Daegu, 712-702, South Korea
mihyekim@cu.ac.kr
[2] School of Computer Science and Engineering,
University of New South Wales, Sydney 2052, Australia
compton@cse.unsw.edu.au

Abstract. A Web-based document management and retrieval system has been developed aimed at small communities in specialized domains and based on free annotation of documents by users. In the proposed approach, the main search mechanism is based on browsing a concept lattice of Formal Concept Analysis (FCA) formulated with a set of keywords with which users annotated the documents. In this paper, we extend our search mechanism by combining the lattice-based browsing structure with conceptual scales of FCA for ontological domain attributes. Our experience with a prototype suggests that conceptual scaling helps users not only to get more specific search results, but also to search relevant documents by the interrelationship between the keywords of documents and ontological attributes.

Keywords: Conceptual scaling, Browsing mechanism, Formal concept analysis.

1 Introduction

Formal Concept Analysis (FCA) was developed by Rudolf Wille in 1982 [17]. FCA is a theory of data analysis which identifies conceptual structures among data based on the philosophical understanding of a ‘concept’ as a unit of thought comprising its extension and intension as a way of modeling a domain. The extension of a concept is formed by all objects to which the concept applies and the intension consists of all attributes existing in those objects. This results in a lattice structure, where each node is specified by a set of objects and the attributes they share. The mathematical formulae of FCA can be considered as a machine learning algorithm which can facilitate automatic document clustering. A key difference between FCA techniques and general clustering algorithms in Information Retrieval is that the mathematical formulae of FCA produce a concept lattice which provides all possible generalization and specialization relationships between object sets and attribute sets. This means that a concept lattice can represent conceptual hierarchies, which are inherent in the data of a particular domain.

A. Hoffmann et al. (Eds.): PKAW 2006, LNAI 4303, pp. 132 – 143, 2006.

FCA has been successfully applied to a wide range of applications. A variety of methods for data analysis and knowledge discovery in databases have also been proposed based on FCA. A number of researchers have proposed an FCA lattice structure for document retrieval [2], [10], [14]. Several researchers have also studied the lattice-based information retrieval with graphically represented lattices for specific domains such as flight information, e-mail management and real estate advertisements [4], [5], [7]. Recently, FCA has been also applied to ontology engineering for structuring and building of ontologies [3], [15].

We also proposed a theoretical framework for a Web-based document management and retrieval system aimed at small communities in specialized domains based on FCA [13]. This approach allowed users themselves to freely annotate their documents. Any relevant documents can be managed by annotating with any terms the users or authors prefer. A number of annotation support tools were proposed not only to allow users to find appropriate annotations for their documents but also to be able to evolve a terminological domain ontology. This resulted in the automatic generation of a lattice-based browsing system which holds hierarchical inheritance relationships among the evolved terms (concepts) in the lattice structure. Document retrieval is based on navigating this lattice structure.

Experiments were conducted in the domain of annotating researchers' home pages according to their research interests in the School of Computer Science and Engineering, University of New South Wales (UNSW). The goal was a system to assist prospective students and potential collaborators in finding research (i.e., staff and student home pages) relevant to their interests. Results indicated that the annotation tools provided a good level of assistance so that documents were easily organized and a lattice-based browsing structure that evolves in an *ad hoc* fashion provided good efficiency in retrieval performance. It was also clear from the results that there is an advantage in lattice-based browsing over hierarchical browsing. The findings suggested that the concept lattice of FCA, supported by annotation techniques was a useful way of supporting the flexible open management of documents required by individuals, small communities and in specialized domains.

In our approach, the main search mechanism is based on browsing a concept lattice of FCA formulated with a set of keywords with which users annotated documents. This concept lattice is reformulated dynamically and incrementally by the addition of a new document with a set of keywords or by refining the existing keywords of the documents. The concept structure can fit into a predetermined terminological ontology used for browsing in information retrieval.

In this paper, we extend our search mechanism by combining the lattice-based browsing structure with conceptual scales of FCA for ontological information. The purpose of this is to allow a user to get more specific search results and to reduce the complexity of the visualization of the browsing structure. The more fundamental purpose of this is to support a hybrid browsing mechanism by combining a structure with keywords and a structure with ontological attributes. This is to allow a user to search relevant documents by the interrelationship between the keywords of documents and ontological domain attributes. The properties such as *author*, *title* and *publication year* can be ontological attributes in a domain relevant to papers. The ideal would be to support both approaches simultaneously because the organization of background knowledge, not only with the vocabularies in taxonomies but also with ontological structures in the form of properties, would be useful for navigating document. For example, a user may want to find papers which are related to "knowledge acquisition" at first. Then, the user wants to see recently published papers only among the search result (i.e., "publication year" ≥ 2005).

This paper is organized as follows. Section 2 gives a brief description of FCA. Section 3 presents a formal framework of conceptual scaling to combine the lattice-based browsing structure with conceptual scales of FCA for ontological attributes. Section 4 describes a system implemented on the Web to demonstrate the value of conceptual scaling. We then conclude with a discussion of possible future directions of the research presented in this paper.

2 Formal Concept Analysis

2.1 Basic Notions of Formal Concept Analysis

FCA starts with a formal context which is a binary relation between a set of objects and a set of attributes. It was defined for the document retrieval system that we proposed in the paper [13] as follows: A *formal context* is a triple $C = (D, K, I)$ where D is a set of documents (objects), K is a set of keywords (attributes) and I is a binary relation which indicates whether k is a keyword of a document d. If k is a keyword of d, it is written dIk or $(d, k) \in I$.

For the domain of research interests used for experiments in the previous work [13] and used in this paper, a document corresponds to a home page and a set of keywords is a set of research topics. That is, D is the set of home pages and K is the set of research topics for a context (D, K, I). However, the word *documents* and *keywords* are also used interchangeably to denote *home pages* (or simply *pages*) and *research topics* (or simply *topics*), respectively in this paper.

From the formal context, formal concepts and a concept lattice are formulated. A formal concept consists of a pair with its extent and intent. The extension of a concept is formed by all objects to which the concept applies and the intension consists of all attributes existing in those objects. These generate a conceptual hierarchy for the domain by finding all possible formal concepts which reflect a certain relationship between attributes and objects. The resulting subconcept-superconcept relationships between formal concepts are expressed in a concept lattice which can be seen as a semantic net providing "hierarchical conceptual clustering of the objects… and a representation of all implications between the attributes" [18, pp.493]. The implicit and explicit representation of the data allows a meaningful and comprehensible interpretation of the information. This lattice is used as the browsing structure. Fig. 1 shows an example of a lattice and a data structure for organizing documents in the lattice. More detailed formulae and explanations of FCA can be found in [9], [13].

2.2 Conceptual Scaling

Conceptual scaling has been introduced in order to deal with many-valued attributes [8], [9]. Usually more than one attribute exists in an application domain and each attribute may have a range of values so that there is a need to handle many values in a context.

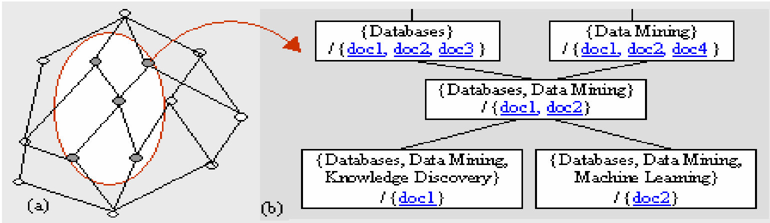

Fig. 1. An example of a browsing structure. (a) Lattice structure. (b) Indexing of the lattice.

In addition, often there is a need to analyze (or interpret) concepts in regard to interrelationships between attributes in a domain. This is the main motivation for conceptual scaling.

For instance, the domain of a "used car market" consists of a number of attributes such as *price*, *year built*, *maker*, *color*, *transmission* and others, and each attribute with a set of values. Such attributes can be considered all together in a context named with a many-valued context. Then, when one is interested in analyzing "used cars" regarding an interrelationship between certain attributes in the many-valued context, they can combine the attributes of interest into a concept lattice. This means that each attribute, or a combination of more than one attribute of the many-valued context, can be transformed into a one-valued context. The derived one-valued context is called a conceptual scale. Then, if one is interested in analyzing the interrelationship between attributes, s/he can choose and combine the conceptual scales which contain the required attributes. This process is called conceptual scaling. A case for the use of this can be seen with TOSCANA[11] and [4]. Conceptual scaling is also used with one-valued contexts in order to reduce the complexity of the visualization [5], [16]. In this case, scales are applied for grouped vertical slices of a large context.

3 Formal Framework of Conceptual Scaling

There are two ways in which we use conceptual scales. Firstly, ontological attributes can be used where readily available (e.g., person, academic position, research group and so on). These correspond to the more structured ontological properties used systems such as Ontoshare[6] and CREAM[12]. The key point of our approach is flexible evolving ontological information but there is no problem with using more fixed information if available. We have included such information for interest and completeness in conceptual scaling. Secondly, a user or a system manager can also group a set of keywords used for the annotation of documents. The groupings are then used for conceptual scaling.

The main difference between our approach and conceptual scaling in TOSCANA is that in our approach all the existing ontological attributes are scaled up together in the nested structure. On the other hand, in the TOSCANA system only one attribute (i.e., a scale) can be combined into the outer structure of an attribute at a time.

3.1 Conceptual Scaling for Ontological Attributes

A many-valued context for ontological attributes is defined as a formal context *C* = (*D*, *M*, *W*, *I*) where *D* is a set of documents, *M* a set of attributes, *W* a set of attribute values. *I* is a ternary relation between *D*, *M* and *W* which indicates that an document *d* has the attribute value *w* for the attribute *m*. We formulate a concept lattice with a set of documents and their keywords. This lattice structure is the main browsing space, but is also an outer structure. Other attributes in a many-valued context are then scaled into a nested structure of the outer structure at retrieval time.

Table 1 is an example of a many-valued context in the domain of research interests. Researchers can be the objects of the context as they are the instances of the home pages. The attributes in the many-valued context can be represented in a partially ordered hierarchy as shown in Fig. 2. The attribute "*position*" in Table 1 is located as a subset of the attribute "*person*" in the hierarchy.

Table 1. An example of the many-valued context for the domain of research interests

	Research group	Sub-group of AI	Person	Position
Researcher1	Artificial intelligence	Knowledge Acquisition	Academic staff	Professor
Researcher2	Computer systems	-	Research staff	Research associate
Researcher3	Networks	-	Academic staff	Associate professor
Researcher4	Databases	-	Academic staff	Senior lecturer
Researcher5	Software engineering	-	Research student	Ph.D. student

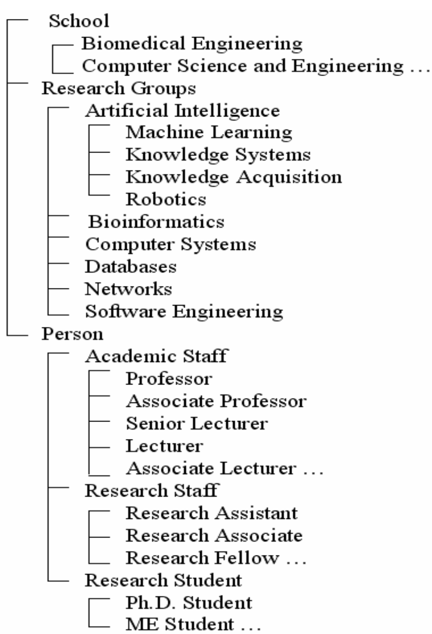

Fig. 2. Partially ordered multi-valued attributes for the domain of research interests

To explain this in a more formal way, the following definition is provided. For example, the has-value relation $\Re$ on the attributes "*person*" and "*position*" is: $\Re$ = {(academic staff, professor), (academic staff, associate professor), …, (research staff, research assistant), …, (research student, Ph.D. student), (research student, ME student)} from Fig. 2. This hierarchy of the many-valued context with the relation $\Re$ is scaled into a nested structure using pop-up and pull-down menus.

Definition 1. Let S_p be a super-attribute and S_c be a sub-attribute. There is a binary relation $\Re$ called the "has-value" relation on S_p and S_c such that $(p, c) \in \Re$ where $p \in S_p$ and $c \in S_c$ if and only if c is a sub-attribute value of p.

Fig. 3 shows examples of inner browsing structures corresponding to concepts of the outer lattice. A nested structure is constructed dynamically from the extent (home pages) of a corresponding concept of the outer lattice incorporating the ontological hierarchy. When a user assigns a set of topics for their page, the page is also automatically annotated with the values of the attributes in the many-valued context. A default home page for individual researchers is provided on the School Web site and as well as every researcher has a login account at the School. We make use of this login account when a user annotates their home page. This provides the default home page address of the user.

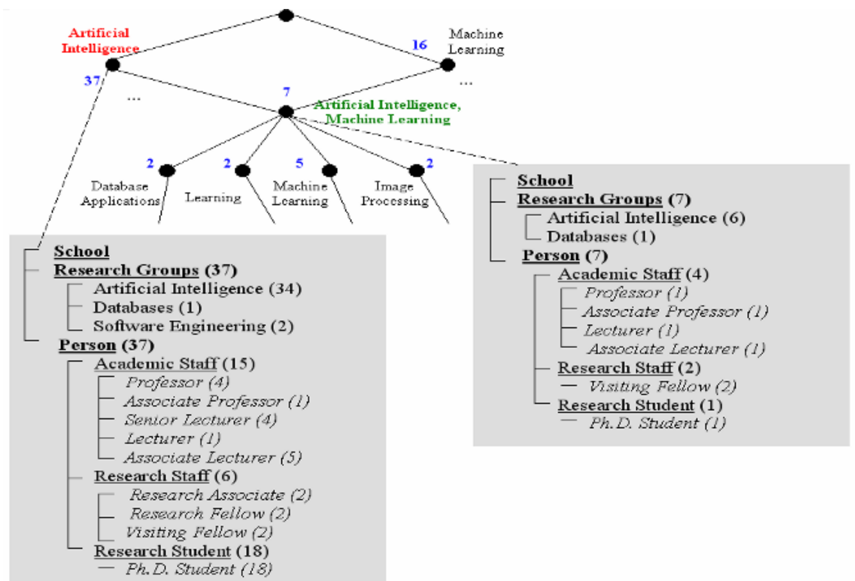

Fig. 3. Examples of nested structures corresponding to concepts. This shows the outer structure of the concepts "artificial intelligence" and "artificial intelligence, machine learning" constructed from a set of home pages and their topics. Numbers in the lattice and in brackets indicate the number of pages corresponding to the concept of the lattice and the attribute value, respectively. The nested structure is presented in a hierarchy deploying all embedded inner structures. The structure is implemented using pop-up and pull-down menus as shown in Fig. 4.

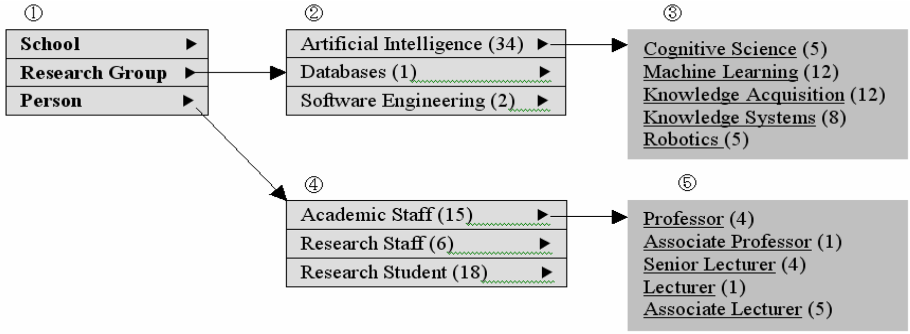

Fig. 4. An example of pop-up and pull-down menus for the nested structures of a concept

The page is an HTML file in a standard format including the basic information about the researcher such as their first name, last name, e-mail address, position and others. The system parses the HTML file and extracts the values for the pre-defined attributes. From the attributes and their extracted values, we formulate a nested structure for a concept of the lattice at retrieval time. Note that the attributes which do not exist in the default home page can be used for conceptual scaling. The user will need to provide the values of those attributes when they assign a set of keyword for their document. For this case, a simple interface to click selection of values or a series of text boxes to be filled is given to the user.

A user can navigate recursively among the nested attributes observing the interrelationship between the attributes and the outer structure. By selecting one of the nested items, the user can moderate the cardinality of the display. Again, the structure with the most obvious attributes can be partly equivalent to the ontological structure of the domain and consequently is considered as an ontological browser which is integrated into the lattice structure with the keywords set.

Fig. 4 shows an example of pop-up and pull-down menus for the nested structure of the concept "artificial intelligence" in Fig. 3. The menu of ① appears when a user clicks on the concept "artificial intelligence". Each item of menu ① is equivalent to a scale in the many-valued context. Suppose that the user selects the attribute *Person* in menu ①, the system then will display a sub-menu of the attribute as shown in menu ④.

3.2 Conceptual Scaling for Grouping Keywords

Conceptual scaling is also applied to group relevant values in the keyword sets used for the annotation of documents. The groupings are determined as required, and their scales are derived on the fly when a user's query is associated with the groupings. This means that the relevant group name(s) is included into the nested structure dynamically at run time. Table 2 shows examples of groupings for scales in the one-valued context for the attribute '*keyword*'. To deal with grouping for scales, the following definition is provided:

Definition 2. Let a formal context $C = (D, K, I)$ be given. A set $G \subseteq K$ is a set of grouping names (generic terms) of C if and only if for each keyword $k \in K$, either $k \in G$ or there exists some generic term $\kappa \in G$ such that k is a sub-term of κ. We define $S = K \setminus G$ and a relation $gen \subseteq G \times S$ such that $(g, s) \in gen$ if and only if s is a sub-term of g.

Table 2. Examples of grouping for scales in the one-valued context for the attribute '*keyword*'

Grouping (generic) names	The members of the grouping names
RDR	FRDR, MCRDR, NRDR, SCRDR
Sisyphus	Sisyphus-I, Sisyphus-II, Sisyphus-III, Sisyphus-IV, Sisyphus-V
Knowledge acquisition	Knowledge acquisition methodologies, Knowledge acquisition tools, Incremental knowledge acquisition, Automatic knowledge acquisition, Web based knowledge acquisition, …
Computer programming	Concurrent programming, Functional programming, Logic programming, Object oriented programming, …
Programming languages	Concurrent languages, Knowledge representaion languages, Logic languages, Object oriented languages, …
Databases	Deductive databases, Distributed databases, Mobile databases, Multimedia databases, Object oriented databases, Relational databases, Spatial databases, Semistructural databases
…	…

Then, when a user's query is $qry \in G$, a *sub-formal context* $C' = (D', K', I')$ of (D, K, I) is formulated where $K' = \{k \in K \mid k = qry$ or $(qry, k) \in gen\}$, $D' = \{d \in D \mid \exists k \in K'$ and $dIk\}$ and $I' = \{ (d, k) \in D' \times K' \mid (d, k) \in I\} \cup \{ (d, qry) \mid d \in D'$ and $qry \in K' \cap G\}$. For instance, suppose that there are groupings as shown in Table 2 and a user's query "*databases*". The query *databas*es $\in G$ so that a sub-context C' is constructed to include a scale of the grouping name *databases* and build a lattice of C'. The user can then navigate this lattice of C'.

Fig. 5 shows an example of a scale with the grouping name "*databases*". The grouping name is embedded into an item of the nested structure along with other scales from the many-valued context in the previous section. There are 12 documents with the concept "Databases" in the lattice, and the node (Databases, 12) embeds the scales as shown in menu ①. The scale "Databases" was derived from the groupings in the one-valued context, while other scales (items) were derived from the many-valued context (i.e., ontological attributes). A user can read that there is one document related to "deductive databases", and two documents with "multimedia databases" etc. By selecting an item of sub-menu ②, the user can moderate the retrieved documents which are only associated with the selected sub-term.

A knowledge engineer/user can set up or change the groupings using a supported tool (i.e., ontology editor) whenever it is required. When a grouping name with a set of sub-terms is added, the system gets the set of documents that are associated with at least one of the sub-terms of the grouping name. Then, the context C is refined to have a binary relation between the grouping term and the documents related to the sub-terms of the grouping term. Next, the lattice of C is reformulated when any change in C is made. If a grouping name is changed, it is replaced with the changed one in the context C and its lattice.

In the case of removal of a grouping in the hierarchy, no change is made in the context C. With this mechanism, the outer lattice can always embed a node which can assemble all documents associated with the sub-terms of a grouping. That is, the.

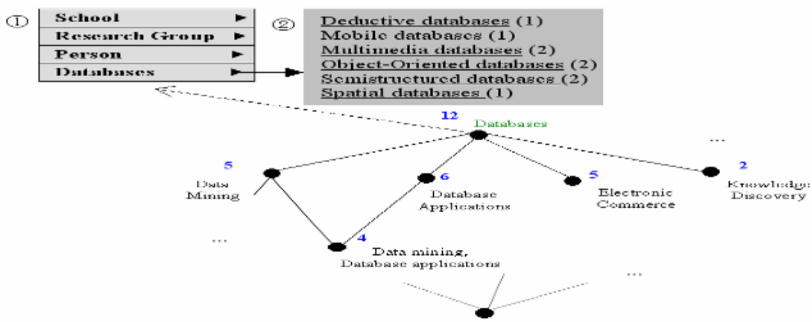

Fig. 5. A conceptual scale for the grouping name "*databases*"

groupings play the role of intermediate nodes in the lattice to scale the relevant values. Groupings can be formed with more than one level of hierarchy. This means that a sub-term of a grouping can be a grouping of other sub-terms.

4 Implementation

To examine the value of conceptual scaling, a prototype has been implemented with a test domain for research topics in the School of Computer Science and Engineering, UNSW. There are around 150 research staff and students in the School who generally have homepages indicating their research projects. The aim here was to allow staff and students to freely annotate their pages so that they would be found appropriately within the evolving lattice of research topics.

Fig. 6 shows an example of conceptual scaling for ontological attributes. It shows examples of inner browsing structures corresponding to the concept "*Artificial Intelligence*" of the outer lattice. We scale up ontological attributes into an inner nested structure. The nested structure is constructed dynamically and associated with the current concept of the outer structure. In other words, the nested attribute values are extracted from the result pages. A nested pop-up menu appears when the user clicks on the "nested" icon in the front of the current node. If the user clicks on one of the attributes items, the results will be changed according to the selection. The user can navigate recursively among the nested attributes.

For instance, we suppose that the user selects the attributes items *Position* → *Academic Staff* → *Professor*. The result then will be changed accordingly. The user can see that there are four researchers whose research topic is "*Artificial Intelligence*" and whose position is *Professor*. Numbers in brackets indicate the number of documents (i.e., homepages) corresponding to the attribute value.

As well, a knowledge engineer can arrange related terms by accessing a tool which allows him or her to set up hierarchical grouping related terms under a common name as described in Section 3.2. Then, when a user's query is related to the grouping(s), the grouping name is included into the inner structure on the fly. Fig. 7 shows an example of conceptual scaling for the grouping "*Databases*". Other items (i.e., *School, Research Groups, Position*) are derived from the ontological attributes. There are 12 documents with the concept "*Databases*" in the lattice.

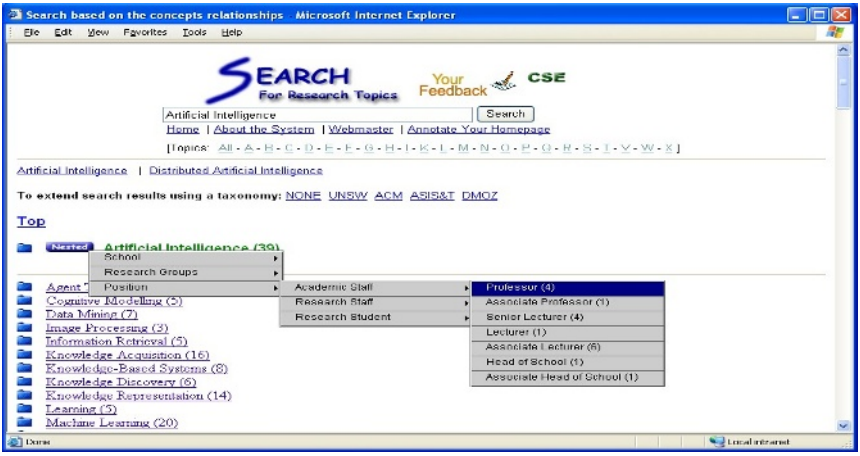

Fig. 6. An example of conceptual scaling for ontological attributes

Fig. 7. An example of conceptual scaling for the grouping "*Databases*"

The user can read that there is one document related to "*Mobile databases*", two documents with "*Multimedia databases*" and so on. By selecting one of the grouping members, the user can moderate the retrieved documents which are only associated with the selected sub-term.

5 Discussion and Conclusion

Having completed a prototype implementation of the presented approach, it seems clear that conceptual scaling facilitates users to get more specific results and to search relevant documents by the interrelationship between the keywords of documents and the domain attributes.

Another purpose of conceptual scaling in our approach was to support a hybrid browsing mechanism by connecting an outer structure with keyword sets of documents (terminological ontology) and an inner nested structure with ontological attributes (ontological structure). The ideal would be to support both approaches simultaneously because the organization of background knowledge, with the vocabularies in taxonomies as well as with ontological structures in the form of properties, would be useful for navigating information.

More fundamentally, conceptual scaling is to deal with multiple Boolean attributes which hold multiple inheritance relations within a one-valued context of FCA. The essence of conceptual scaling is to impose on this a single inheritance hierarchy or equivalently some of the Boolean attributes are reorganized as being mutually exclusive values of some unnamed attributes. Either way there is recognition that a group of Boolean attributes are mutually exclusive. In conceptual scaling, one selects one of the mutually exclusive attributes from a set and a sub-lattice containing these values is shown. A number of attribute selections can be made at the same time to give the sub-lattice. Existing attributes can be used as the parent of a group of mutually exclusive attributes or new names for the grouping can be created.

We had previously carried out user studies on the general usefulness of evolving ad hoc lattices [13]. A next step is to evaluate the usefulness of our approach to conceptual scaling and its scalability with large data sets. The user interface for conceptual scaling also needs to be improved. Users may want to find documents from ontological attributes first, then scale up their search result with keyword sets (i.e., the opposite of the current interface) or interchangeably.

Further work would be related to the extension of our approach regarding ontologies. We have adapted conceptual scaling of FCA to scale up the browsing structure derived from the keywords of documents with ontological information where readily available such as *person*, *academic position* and *research group*. However, ideally we would derive conceptual scales from an existing ontology, which is imported from standards or constructed for the system. The use of these scales could be automated if the document was appropriately marked up according to the ontology. This would give us a system that was flexible and open, but also had the type of ontological commitment represented by the KA^2 initiative[1] and CREAM[12]. It will be interesting to examine the trade-offs in allowing such requirements to emerge rather than anticipating them and also the relative costs in marking up documents rather than providing information to a server.

It will be also essential to use one of ontology representation languages such as RDF, OIL and OWL as standards instead of the proprietary text formats used currently both for the concept lattice and the ontological attributes.

References

1. Benjamins, V. R., Fensel, D., Decker, S., Perez, A. G.: $(KA)^2$: Building Ontologies for the Internet: a Mid-term Report, International journal of human computer studies (1999) 51(3): 687-712.
2. Carpineto, C., Romano, G.: Information retrieval through hybrid navigation of lattice representations, International Journal of Human-Computer Studies (1996) 45:553-578.

3. Cimiano, P., Hotho, A., Stumme, G., Tane, J.: Conceptual Knowledge Processing with Formal Concept Analysis and Ontologies, Proceedings of the Second International Conference on Formal Concept Analysis (ICFCA 04), Springer-Verlag (2004) 189-207.
4. Cole, R., Eklund, P.: Browsing Semi-structured Web texts using Formal Concept Analysis, Conceptual Structures: Broadening the Base, Proceedings of the 9th International Conference on Conceptual Structures (ICCS 2001), Springer-Verlag (2001) 290-303.
5. Cole, R., Stumme, G.: CEM - A Conceptual Email Manager, Conceptual Structures: Logical, Linguistic, and Computational Issues, Proceedings of the 8th International Conference on Conceptual Structures (ICCS 2000), Springer-Verlag (2000) 438-452.
6. Davies, J., Duke, A., Sure, Y.: OntoShare – A Knowledge Environment for Virtual Communities of Practice, Proceedings of the Second International Conference on Knowledge Capture (K-CAP 2003), ACM, New York (2003) 20-27.
7. Eklund, P., Groh, B., Stumme, G., Wille, R.: A Contextual-Logic Extension of TOSCANA, Conceptual Structures: Logical, Linguistic, and Computational Issues, Proceedings of the 8th International Conference on Conceptual Structures (ICCS 2000), Darmstadt, Springer-Verlag (2000) 453-467.
8. Ganter, B., Wille, R.: Conceptual Scaling, In: F. Roberts (eds.): Application of Combinatorics and Graph Theory to the Biological and Social Sciences, Springer-Verlag (1989) 139-167.
9. Ganter, B., Wille, R.: Formal Concept Analysis: Mathematical Foundations, Springer-Verlag, Heidelberg (1999).
10. Godin, R., Missaoui, R., April, A.: Experimental comparison of navigation in a Galois lattice with conventional information retrieval methods, International Journal of Man-Machine Studies (1993) 38 :747-767.
11. Groh, G., Strahringer, S., Wille, R.: TOSCANA-Systems Based on Thesauri, Conceptual Structures: Theory, Tools and Applications, Proceedings of the 6th International Conference on Conceptual Structures (ICCS'98), Springer-Verlag (1998) 127-138.
12. Handschuh, S., Staab, S.: CREAM – CREAting Methadata for the Semantic Web, Computer Networks (2003) 242:579-598.
13. Kim, M., Compton, P.: Evolutionary Document Management and Retrieval for Specialised Domains on the Web, International journal of human computer studies (2004) 60(2): 201-241.
14. Priss, U.: Lattice-based Information Retrieval, Knowledge Organisation (2000) 27(3):132-142.
15. Quan, T.T., Hui, S.C., Fong, A.C.M., Cao, T.H.: Automatic Generation of Ontology for Scholarly Semantic Web, The Semantic Web – ISWC 2004: Proceedings of the Third International Semantic Web Conference, Hiroshima, Springer-Verlag (2004) 726-740.
16. Stumme, G.: Hierarchies of Conceptual Scales, 12th Banff Knowledge Acquisition, Modelling and Management (KAW'99), Banff, Canada, SRDG Publication, University of Calgary (1999) 5.5.1-18.
17. Wille, R.: Restructuring lattice theory: an approach based on hierarchies of concepts, In: Ivan Rival (eds.), Ordered sets, Reidel, Dordrecht-Boston (1982) 445-470.
18. Wille, R.: Concept lattices and conceptual knowledge systems, Computers and Mathematics with Applications (1992) 23:493-515.