Automatic Selection of Noun Phrases as Document Descriptors in an FCA-Based Information Retrieval System

Juan M. Cigarrán, Anselmo Peñas, Julio Gonzalo, and Felisa Verdejo

Dept. Lenguajes y Sistemas Informáticos**
E.T.S.I. Informática, UNED, Madrid Spain
{juanci, anselmo, julio, felisa}@lsi.uned.es
http://nlp.uned.es

Abstract. Automatic attribute selection is a critical step when using Formal Concept Analysis (FCA) in a free text document retrieval framework. Optimal attributes as document descriptors should produce smaller, clearer and more browsable concept lattices with better clustering features. In this paper we focus on the automatic selection of noun phrases as document descriptors to build an FCA-based IR framework. We present three different phrase selection strategies which are evaluated using the *Lattice Distillation Factor* and the *Minimal Browsing Area* evaluation measures. Noun phrases are shown to produce lattices with good clustering properties, with the advantage (over simple terms) of being better intensional descriptors from the user's point of view.

1 Introduction

The main goal of an Information Retrieval (IR) system is to ease information access tasks over large document collections. Starting from a user's query, usually made in natural language, a classic IR system retrieves the set of documents relevant to the user needs and shows them using ranked lists (e.g. Google, Yahoo or Altavista).

The use of ranked lists, however, does not always satisfy the user's information needs. Ranked lists are best suitable when users know exactly what they are looking for and how to express it using the right words (e.g. the last driver for a specific graphics card or the papers published by any author). More generally, ranked lists can be useful when the task is to retrieve a very small number of relevant items. However, when there is a need to retrieve relevant information from many sources, or when the query involves fuzzy or polysemous terms, the use of a ranked list implies to read almost the whole list to find the maximum

** This work has been partially supported by the Spanish Ministry of Science and Technology within the following projects: TIC-2003-07158-C04 Answer Retrieval from Digital Documents, R2D2; and TIC-2003-07158-C04-02 Multilingual Answer Retrieval Systems and Evaluation, SyEMBRA.

B. Ganter and R. Godin (Eds.): ICFCA 2005, LNAI 3403, pp. 49–63, 2005.

number of relevant documents. For instance, if we ask *Google* (*www.google.com*) with the query '*jaguar*' looking for documents related with the jaguar as animal, we obtain 7.420.000 of web pages as a result. Of course, not all the retrieved pages are relevant to our needs and, based on the ranking algorithm of Google [1], pages containing the term '*jaguar*' but with different senses (i.e. jaguar as a car brand or jaguar as a Mac operating system) are mixed up in the resulting ranking, making the information access task tedious and time consuming.

As an alternative, clustering techniques organize search results allowing a quick focus on specific document groups and improving, as a consequence, the final precision of the system from a user's perspective. In this way, some commercial search engines (i.e. www.vivisimo.com) apply clustering to a small set of documents obtained as a result of a query or a filtering profile. The use of clustering as a post-search process applied only to a subset of the whole document collection makes clustering an enabling search technology halfway between browsing (i.e. as in web directories) and pure querying (i.e. as in Google or Yahoo).

We propose the use of Formal Concept Analysis (FCA) as an alternative to classic document clustering, not only considered as an information organization mechanism but also as a tool to drive the user's query refinement process. Advantages of FCA over standard document clustering algorithms are: a) FCA provides an intensional description of each document cluster that can be used for query modification or refinement, making groups more interpretable; and b) the clustering organization is a lattice, rather then a hierarchy, which is more natural when multiple classification is possible, and facilitates recovering from bad decisions while browsing the lattice to find relevant information.

The main drawbacks of FCA disappear when dealing with small contexts (i.e. with a small set of documents obtained as the result of a search process): a) FCA is computationally more costly than standard clustering, but when it is applied to small sets of documents (i.e. in the range of 50 to 500 documents) is efficient enough for online applications; and b) lattices generated by FCA usually are big, complex and hence difficult to use for practical browsing purposes. Again, this should not be a critical problem when the set of documents and descriptors are restricted in size by a previous search over the full document collection.

But the use of FCA for clustering the results of a free text search is not a straightforward application of FCA. Most Information Retrieval applications of FCA are domain-specific, and rely on thesauruses or (usually hierarchical) sets of keywords which cover the domain and are manually assigned as document descriptors [12, 6, 7, 5, 8, 16, 9]. The viability of using FCA in this scenario implies to solve some challenges related with: a) the automatic selection of suitable descriptors for context building, b) the rendering of node descriptions, c) the visualization of concept lattices obtained, and; d) the definition of suitable query refinement tasks. Most importantly, the (non-trivial) issue of how to evaluate and compare different approaches has barely been discussed in the past.

This paper is presented as a continuation of the research presented in [4], where the problem of automatic selection of descriptors was first addressed. In

that previous research we focused on the automatic selection of single terms as document descriptors. The problem with single terms is that, even if the lattice has good clustering/browsing properties, the intensional descriptions are not descriptive enough from a user's point of view. Noun phrases, however, tend to be excellent descriptors of the main concepts in a document, and are easily interpretable to users. Therefore, in this paper we will refine our proposal by using noun phrases as document descriptors. We will propose and compare three algorithms to select noun phrase document descriptors, using a shallow, efficient phrase extraction technique and the evaluation framework introduced in [4].

The paper is organized as follows. First of all we will present the information retrieval and organization system in which our evaluation framework is based; we will introduce the information organization model used and the system architecture. Then, we will describe the phrase extraction methodology and the set of phrase selection strategies presented for evaluation. Finally, we will present the evaluation experiments and discuss the results.

2 Information Retrieval and Organization System

2.1 The Information Organization Model

Using the ranked list of documents retrieved by a search engine, we generate a concept lattice to organize these search results. Lattices generated are based on a formal context $K := (G, M, I)$, where $G = \{doc_1, doc_2, \ldots, doc_n\}$ represents a subset of the retrieved documents, $M = \{desc_1, desc_2, \ldots, desc_k\}$ is a subset of document descriptors and I is the incidence relationship.

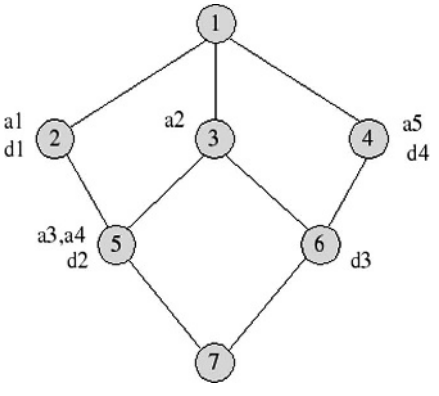

Fig. 1. Example concept lattice

This model relies on the set of concepts generated and its corresponding concept lattice, while introducing some assumptions about what concept information is going to be considered for showing, browsing or evaluation purposes.

Table 1. Formal concepts and their corresponding information nodes of the example lattice of Figure 1

Concept Node	**Information Node**
$c_1 = (\{d_1, d_2, d_3, d_4\}, \{\emptyset\})$	$n_1 = (\{\emptyset\}, \{\emptyset\})$
$c_2 = (\{d_1, d_2\}, \{a_1\})$	$n_2 = (\{d_1\}, \{a_1\})$
$c_3 = (\{d_2, d_3\}, \{a_2\})$	$n_3 = (\{\emptyset\}, \{a_2\})$
$c_4 = (\{d_4, d_3\}, \{a_5\})$	$n_4 = (\{d_4\}, \{a_5\})$
$c_5 = (\{d_2\}, \{a_1, a_2, a_3, a_4\})$	$n_5 = (\{d_2\}, \{a_1, a_2, a_3, a_4\})$
$c_6 = (\{d_3\}, \{a_2, a_5\})$	$n_6 = (\{d_3\}, \{a_2, a_5\})$
$c_7 = (\{\emptyset\}, \{a_1, a_2, a_3, a_4, a_5\})$	$n_7 = (\{\emptyset\}, \{a_1, a_2, a_3, a_4, a_5\})$

In our context, the remade formal concepts will be called *information nodes* and are defined as follows. Being A_i and B_i the extent and the intent of a generic formal concept c_i, we define its corresponding information node n_i as:

$$n_i = (AI_i, BI_i) \equiv \begin{cases} AI_i \subseteq A_i, \text{where } \forall \alpha \in AI_i \cdot \gamma(\alpha) = c_i \\ BI_i = B_i, \end{cases} \quad (1)$$

We also define a *connection node* as a information node where $AI_i = \emptyset$.

Information nodes are based on the assumption that a concept node should not display all its extent information. Working with the whole extent implies no differences between those documents which are object concepts (i.e. they are not going to appear as extent components of lower nodes) and those documents that can be specialized.

Figure 1 shows, as an example, a concept lattice where concept nodes are represented in Table 1 followed by its corresponding information nodes. Showing concept extent implies, for instance, that a user located at the top node of the lattice would be seeing the whole list of the documents retrieved at once. This situation would make our system essentially identical to a ranked list for browsing purposes. The use of information nodes overcome this problem, granting the document access only when no more specialization is possible. This model agrees with the access model used by most web directories (e.g. Open Directory Project ODP or Yahoo! Directory), where it is possible to find categories with no documents (i.e. categories that being very general do not completely describe any web page).

2.2 System Architecture

Our proposal to integrate FCA in a information retrieval and organization system is based on the architecture presented in Figure 2. It is divided in four main subsystems that solve the indexing, retrieval, lattice building and lattice representation and visualization tasks. Interactions between the subsystems are represented in the same figure and can be summarized as follows:

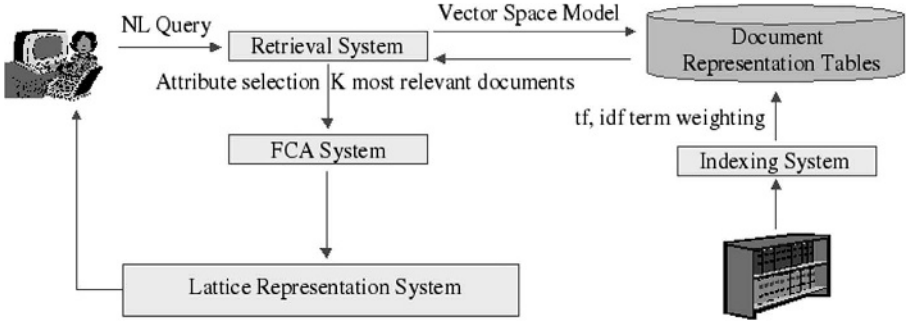

Fig. 2. System Architecture

1. The user makes a query using natural language as query language. The retrieval subsystem processes the query, removing stopwords and lemmatising meaningful query terms. The output of this step is a query representation which is used in the next step to make the retrieval process.
2. Relevant documents are retrieved by the retrieval subsystem using the query representation and the vector space model cosine measure.
3. The FCA subsystem builds a formal context using the first n most relevant documents retrieved and the k most suitable descriptors, which generates a concept lattice.
4. The Lattice representation subsystem applies the information organization model to the lattice generated, displaying a suitable visualization for user browsing purposes.

Currently, two prototypes have been developed based on this architecture, JBraindead and JBraindead2, which are used as our framework for evaluation purposes.

3 Phrase Selection

Dealing with noun phrases involves three main steps [14]: phrase extraction, phrase weighting and phrase selection.

3.1 Phrase Extraction

Phrase extraction is aimed at detecting all the possible candidate phrases that could be relevant and descriptive for the document in which they appear, and for the domains related to the document. These phrases are sequences of words that match the pattern of terminological phrases [13].

The phrase extraction procedure is divided in four main steps:

1. Document tokenisation, which identifies all possible tokens in the document collection.
2. Text segmentation according to punctuation marks.

3. Lemmatisation and part-of-speech tagging, in order to associate a base form to each word, together with its grammatical category.
4. Syntactic pattern recognition to detect the sequences of words that match a terminological phrase structure. The pattern is defined in Formula 2 as morpho-syntactic tag sequences. If the text contains a word sequence whose gramatical tags match the pattern, then a new candidate phrase has been detected.
   According to the pattern, a candidate phrase is a sequence of words that starts and ends with a noun or an adjective and might contain other nouns, adjectives, prepositions or articles in between. This pattern does not attempt to cover the possible constructions of a noun phrase, but the form of terminological expressions. The pattern is general enough to be applied to several languages, including English and Spanish [13, 15].

The result of the term detection phase is a list of terminological phrases with their collection term and document frequencies. For practical purposes, in our experiments we only consider those phrases of two or three terms (longer phrases usually have very low frequency values, which is of little help for concept clustering purposes).

$$[Noun|Adjective]\,[Noun|Adjective|Preposition|Article] * [Noun|Adjective] \tag{2}$$

3.2 Selection Strategies

Once suitable phrases are extracted, the next step is to select the subset of phrases that best characterizes the retrieved document set. Results of this phase are critical to a) reach a reasonable cardinality for the concept set; and b) reach an optimal distribution of the documents in the lattice. A good balance between the cardinality of the descriptors set and the coverage of each descriptor should provide meaningful and easy to browse concept lattices.

With this objective in mind, we introduce three different phrase selection strategies to select candidate phrases as document descriptors: a) selection of generic phrases that occur with highest document frequencies (*Generic Balanced* strategy), b) selection of those phrases that, containing at least one query term as phrase component, have the highest document frequency values *(Query Specific Balanced* strategy) , and c) selection of phrases which are terminologically relevant to describe the retrieved document set (*Terminological* strategy).

Generic Balanced Strategy. This strategy selects the k phrases with the highest document frequency covering the maximum number of documents retrieved. Noun phrases occur much less frequently than terms, so the main idea is to describe, at least with one descriptor, the maximum number of documents. This is the algorithm to select the k set of descriptors according to this principle:

– Being $D = \{doc_1, doc_2, \ldots, doc_n\}$, the set of n most relevant documents selected from the retrieved set, and $P = \{phr_1, phr_2, \ldots, phr_m\}$, the set of

m phrases that appear in the n documents. We define a set $G = \{\emptyset\}$ that will store the covered documents, and a set $S = \{\emptyset\}$, that will store the final selected phrases.

- Repeat until $|S| = k$ or $|D| = \emptyset$, where k is the number of phrases to select for the document descriptors set.
  1. From P extract the phr_i with the highest document frequency in current D. If two or more phrases should have the same document frequency, then select the phrase that appear in the most relevant document of D (i.e. documents are ranked by the search engine).
  2. Store in an empty auxiliary set (AUX) those documents, belonging to current D, where phr_i appears.
  3. Delete the selected phrase from the candidate phrases set. $P = P \backslash \{phr_i\}$.
  4. Delete the selected documents from the documents set. $D = D \setminus AUX$
  5. Add the selected phrase to the final descriptors set. $S = S \cup \{phr_i\}$
  6. Add the selected documents to the used documents set. $G = G \cup AUX$
- The S set will contain the k highest document frequency phrases with maximal document coverage.

Query Specific Balanced Strategy. This is the same strategy, but restricting the set of candidate phrases P to those phrases containing one or more query terms as phrase components. First, we directly add to the S set the k' phrases with more than one query term and with a document frequency greater than one. Then we apply the above algorithm to calculate the best $k - k'$ phrases containing one query term. The main idea of this approach is to extract query related phrases that, due to its lower document frequencies, are not selected as document descriptors by the Generic Balanced selection strategy. In addition, phrases containing query terms should be better suggestions for users.

Terminological Strategy. Here we apply the terminological formula introduced in [4], but computed on phrases instead of terms. The main motivation of this formula is to weight with higher values those phrases that appear more frequently in the retrieved document set than in the whole collection. Formula 3 reflects this behavior, where w_i is the terminological weight of phrase i, $tf_{i,ret}$ is the relative frequency of phrase i in the retrieved document set, $f_{i,ret}$ is the retrieved set document frequency of phrase i, and $tf_{i,col}$ is the relative frequency of phrase i in the whole collection minus the retrieved set.

$$w_i = 1 - \frac{1}{log_2\left(2 + \frac{tf_{i,ret} \cdot f_{i,ret} - 1}{tf_{i,col} + 1}\right)} \tag{3}$$

4 Evaluation

4.1 Information Retrieval Testbed and Evaluation Measures

The Information Retrieval testbed for our experiments has been the same as in [4]. The new JBraindead2 prototype, which was tested with a set of 47 TREC-like

topics coming from the CLEF 2001 and 2002 campaigns, and having extensive manual relevance assessments in the CLEF EFE 1994 text collection.

The main evaluation measures used were the *Lattice Distillation Factor* (LDF) and the *Minimal Browsing Area* (MBA) defined and motivated in [4]. LDF is the precision gain between the original ranked list and the minimal set of documents which should be inspected in the lattice in order to find the same amount of relevant information. The biggest the LDF, the better the lattice. The MBA is the percentage of nodes in the lattice which have to be considered with an optimal browsing strategy. The smaller the MBA, the better the lattice.

Previous research on FCA applied to IR has barely focused on evaluation issues. Two exceptions are [11, 3], where empirical tests with users were conducted. In both cases, documents were manually indexed and the lattices were built using that information. Therefore, the problem of choosing optimal indexes was not an issue. In free-text retrieval, however, selecting the indexes is one of the main research challenges. Our LDF and MBA measures estimate the quality of the lattices for browsing purposes on different index sets, permitting an initial optimization of the attribute selection process prior to experimenting with users.

4.2 Experiments

We made three main experiments to test which selection strategy had best performance values in our information organization framework. Experiments are described in the following subsections.

Table 2. Experimental results for Experiment 1 with $k = 10$ and $k = 15$ with a number of documents $n = 100$. The averaged precision of the baseline ranked list was 0.15

	GB	**QSB**	**T**
LDF(%) (k=10)	255.7	84.17	16.7
LDF(%) (k=15)	562.72	127.01	25.17
MBA(%) (k=10)	46.97	59.25	69.14
MBA(%) (k=15)	42.48	60.64	73.29
Nodes (k=10)	114.27	35.89	13.31
Nodes (k=15)	161.24	48.07	19.87
Obj. Concepts (k=10)	52.8	27.56	12.09
Obj. Concepts (k=15)	65.73	36.98	17.93
Connect. Nodes (k=10)	61.47	8.33	1.22
Connect. Nodes (k=15)	95.51	11.09	1.93

Experiment 1. We evaluated the system using only noun phrases as document descriptors. We applied the three selection strategies (i.e. generic balanced, query-specific balanced and terminological) presented to extract the k most relevant phrases. The strategies were tested with the first 100 most relevant documents retrieved and a set of descriptors with $k = 10$ and $k = 15$. Results are summarized in Table 2 and in Figures 3 and 4, where GB stands for the generic

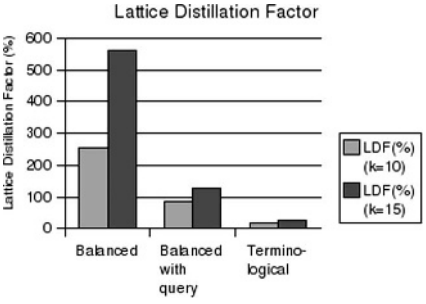

Fig. 3. Lattice Distillation Factor in Experiment 1

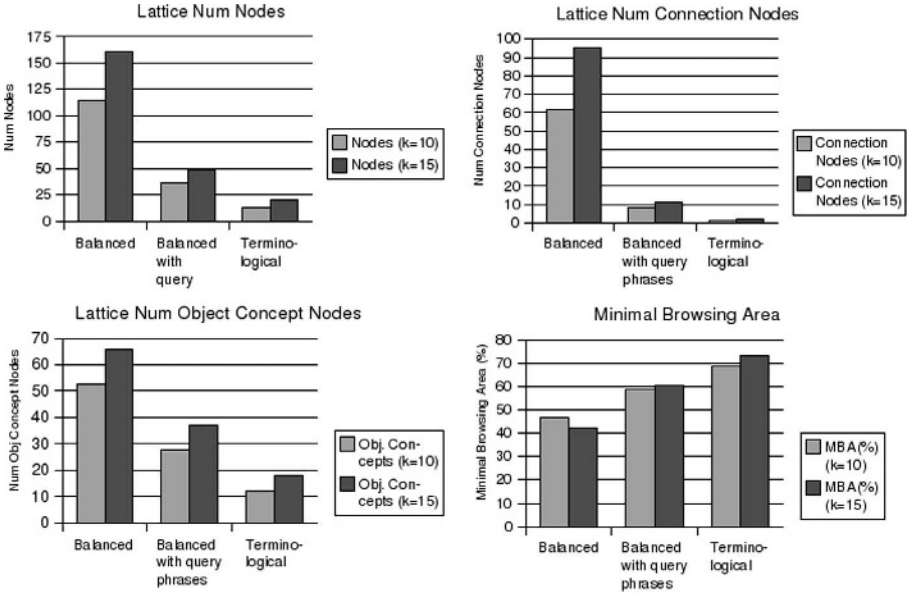

Fig. 4. Number of nodes, object concepts, connection nodes and Minimal browsing Area for Experiment 1

balanced phrase selection strategy, QSB is the query specific balanced strategy and T the terminological selection strategy.

Although the best lattice distillation factor (LDF) values are obtained using the Generic Balanced strategy (with improvements of 256% and 563% in the retrieval precision of the system), the size of the concept lattices generated makes them too complex for practical browsing purposes. In addition, minimal browsing area (MBA) values (i.e. which imply to explore a 47% and a 42% of the lattices) imply that a substantial proportion of a big lattice has to be inspected to find all relevant information. We estimate that the size (i.e. number of nodes) of a lattice should in any case be below 50% of the size of the document set; otherwise, the lattice makes document inspect even more complex than the original ranked list.

The main reason for the high LDF values in the Generic Balanced approach is the nature of the selected phrases. High document frequency phrases selected are shared by many documents, which enhances the possibility of combinations between documents. This situation generates lattices with a great number of connection nodes (i.e. nodes that are not object concepts and, as a consequence, do not have any document to be shown), which makes very easy to find the optimal ways to the relevant nodes without traversing any non-relevant node. A large set of connection nodes also explains the high values for the MBA.

Query Specific Balanced and Terminological strategies obtain lower LDF values but, in contrast, the number of nodes generated and the MBA values are more appropriate to our requirements. These two selection strategies choose more specific phrases (i.e. with lower document frequencies) than the Generic Balanced recipe, which explains the smaller size of the concept lattices obtained. At this point, two questions arise: due to the lower document frequency of the phrases selected, is the whole document space covered by the set of descriptors selected? and how many documents are generators of the top concept as object concept?. Answers to these questions explain the lower LDF values: a) a top information node with a small set of documents implies to read only a few documents at the very first node of the lattice. In this case, a low LDF value implies a poor clustering process with the relevant and non relevant documents mixed up in the same clusters, and; b) a top information node with a large set of documents implies to read too many documents at the first node of the lattice. In this situation, a low LDF value does not necessarily imply a bad clustering process, but probably a damaged LDF where lots of non relevant documents are clustered in the top node, and therefore always counted for precision purposes.

Experiment 2. In order to solve this problem, we proposed to characterize the documents which do not receive descriptors with a dummy descriptor '*other topics*'. This new context description generates a concept lattice with a top information node with an empty set of documents, which ensures a first node of the lattice with no documents to show. If the new node containing "other topics" documents does not contain relevant documents, then it will not affect negatively to the LDF measure.

Using this "other topics" strategy, we re-evaluated the three selection strategies with the first 100 most relevant documents retrieved and a set of descriptors with $k = 10$. Results are shown in Table 3 and in Figure 5, where GB represents the generic balanced phrase selection strategy, QSB the query specific balanced strategy and T the terminological selection strategy.

The results show much better LDF values for the Specific Query Balanced and the Terminological selection strategies than in the previous experiment (with improvements of 182% and 1300% respectively). The LDF value for the Generic Balanced selection strategy is also improved (237.84%). The values obtained in the previous experiment indicate that the bad clustering performance of the proposed selection strategies were due to the generation of top information nodes with too many non relevant documents (which the user is forced to read) which damage the final LDF values.

Table 3. Experimental results for Experiment 2 with $k = 10$ and with a number of documents selected of $n = 100$. The averaged precision of the baseline ranked list was 0.15

	GB	QSB	T
LDF(%) (k=10)	863.86	237.64	233.81
MBA(%) (k=10)	48.4	65.55	84.62
Nodes (k=10)	115.27	36.87	14.31
Obj. Concepts (k=10)	52.8	27.56	12.09
Connect. Nodes (k=10)	62.47	9.31	2.22

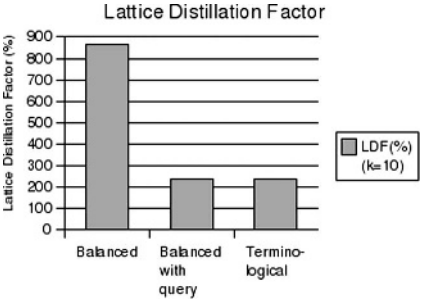

Fig. 5. Lattice Distillation Factor in Experiment 2

LDF values for the Specific Query Balanced and the Terminological selection strategies are very similar; a new question has to be asked to differentiate both: having similar LDF values, which of these strategies groups documents best?. Both strategies generate an acceptable number of nodes, but looking at the number of object concepts generated, we can see that the Specific Query Balanced strategy doubles the number of object concepts in the Terminological one. The number of object concepts is directly related with the number and size (i.e. in average) of the clusters generated and, as a consequence, the Specific Query Balanced strategy generates more clusters with a smaller size than those generated by the Terminological one. In this situation, a small set of large clusters gives a vast, poorly related view of the clustered document space where the user is not able to specialize the contents of the large relevant clusters selected. We think that this scenario is not desirable for browsing purposes and, provided that acceptable lattice sizes and similar LDF values are obtained, the selection strategy which gives the maximum number of clusters should be preferred.

Finally, there is a practical reason to select Specific Query Balanced as the preferred selection strategy: it is very simple to compute, and does not need collection statistics to be calculated. This is relevant, e.g., if the goal is to cluster the results of a web search without having collection statistics from the full web.

Experiment 3. As an additional experiment to avoid the overload of the top information nodes, we tested the effect of adding the query terms as document

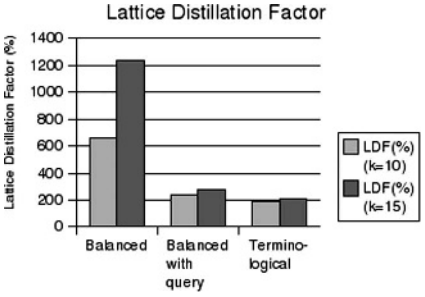

Fig. 6. Lattice Distillation Factor in Experiment 3

descriptors, in addition to noun phrases. The idea is based in the fact that query terms, that should appear as attribute concepts in the first levels of the lattice, are not only a good aid to drive the initial user navigation, but also they make a natural partition of the document space clarifying basic document relations and generating top information nodes with an empty document set.

We evaluated the three selection strategies with the first 100 most relevant documents retrieved and a set of descriptors with $k = 10$ and $k = 15$ built using the qt query terms and the $k - qt$ phrases selected. Results are summarized in Table 4 and in Figures 6 and 7, where GB represents the generic balanced phrase selection strategy, QSB the query specific balanced strategy and T the terminological selection strategy.

Although the results show that the Generic Balanced selection strategy obtains the best LDF values, the large number of nodes generated and, as a consequence, the small size of the clusters generated and the large number of connection nodes lead us to reject this selection strategy as optimal for our information organization purposes.

Query Specific Balanced and Terminological selection strategies perform lower LDF values but generate better-sized lattices. Query Specific obtains better LDF

Table 4. Experimental results for Experiment 3 with $k = 10$ and $k = 15$, with a number of documents $n = 100$. The averaged precision of the baseline ranked list was 0.15

	GB	**QSB**	**T**
LDF(%) (k=10)	664.34	235.97	192.04
LDF(%) (k=15)	1239.75	277.99	208.14
MBA(%) (k=10)	43.57	59.74	66.27
MBA(%) (k=15)	38.36	55.43	63.55
Nodes (k=10)	92.53	38.96	16.33
Nodes (k=15)	191.31	53	23.49
Obj. Concepts (k=10)	49.22	27.91	13.84
Obj. Concepts (k=15)	66.36	38.36	20.16
Connect. Nodes (k=10)	43.31	11.04	2.49
Connect. Nodes (k=15)	124.96	14.64	3.33

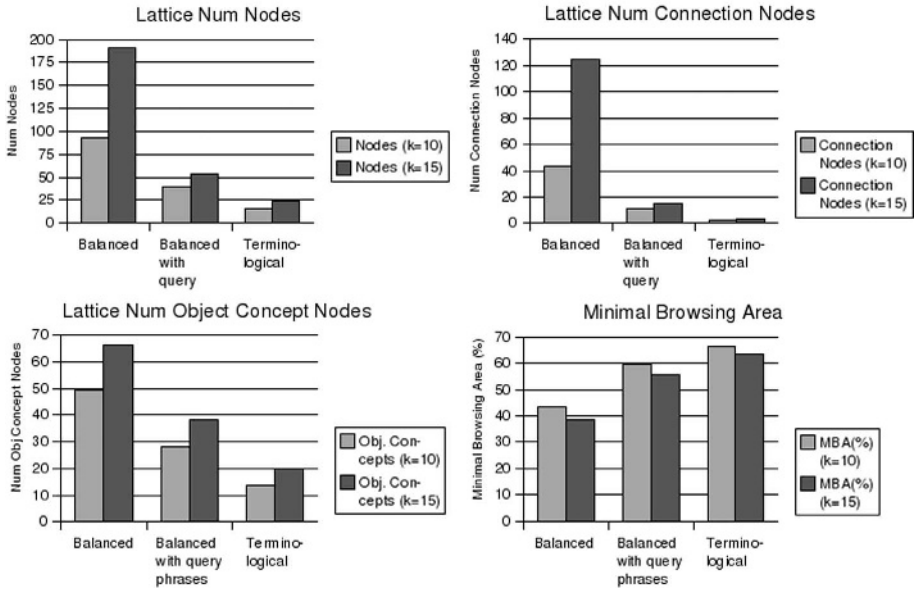

Fig. 7. Number of nodes, object concepts, connection nodes and Minimal browsing Area in Experiment 3

values than Terminological, which, in addition to our previous experiment discussion, makes this strategy better than the others in our information organization framework. Furthermore, although the growth of k does not significantly improve the LDF values, a better improvement is detected with the Query Specific Balanced selection strategy (i.e. Query Specific selection strategy improves in a 18% LDF values while, in contrast, Terminological selection strategy improves LDF only in a 8%).

5 Conclusions

Starting from the organization framework presented in [4], this paper has explored the possibility of using noun phrases as document descriptors to generate lattices for browsing search results. We have focused our research on the development of different phrase selection strategies, which have been tested using the LDF and MBA evaluation measures specifically designed for this task and introduced in [4].

The experiment results reveal a high clustering power for lattices built using all three selection strategies studied, being the Query Specific Balanced selection strategy the most suitable for user browsing purposes. The use of noun phrases, in contrast with the use of terms as document descriptors, deals with low document frequency values so the use mixed approach based on the use of query terms and phrases as document descriptors gives us the best LDF values. In addition, noun

phrases are more adequate as intensional descriptions of lattice nodes from a user's point of view.

Our current work is focused on two main objectives: a) the evaluation of the generated lattices in an interactive setting involving users, and b) the research on lattice visualization and query refinement aspects.

The information organization framework proposed illustrates the scalability of FCA to unrestricted IR settings if it is applied to organize search results, rather than trying to structure the whole document collection. In this direction, some recent efforts have also been made by other systems such as CREDO [2] (i.e. an FCA system oriented to organize web search results) or DOCCO [10] (i.e. and FCA system oriented to manage the organization and retrieval of PC stored files with different formats) with promising results. A distinctive feature of our proposal is the incorporation of a framework for the systematic evaluation and comparison of indexing strategies within this general paradigm of FCA as a tool to organize search results in generic free-text retrieval processes.

References

1. Brin S., Page L.: The anatomy of a large-scale hypertextual Web search engine. Computer Networks and ISDN Systems. 30 **1–7** 107–117 1998.
2. Carpineto C., Romano G. Concept Data Analysis. Theory and Applications. Ed. Willey ISBN: 0-470-85055-8. 2004.
3. Carpineto, C. and Romano, G. A Lattice Conceptual Clustering System and its Application to Browsing Retrieval. Machine Learning (1996) 24, 95-122.
4. Cigarrán J.M., Gonzalo J., Peñas A., Verdejo F.: Browsing Search Results via Formal Concept Analysis: Automatic Selection of Attributes. Concept Lattices, Second International Conference on Formal Concept Analysis, ICFCA 2004. Sydney, Australia. Ed. Eklund P.W. LNCS **2961** 74–87 Springer-Verlag, Berlin 2004.
5. Cole, R. J. The management and visualization of document collections using Formal Concept Analysis. Ph. D. Thesis, Griffith University. 2000.
6. Cole, R. J. and Eklund, P. W. Application of Formal Concept Analysis to Information Retrieval using a Hierarchically structured thesaurus.
7. Cole, R. J. and Eklund, P. W. A Knowledge Representation for Information Filtering Using Formal Concept Analysis. Linkoping Electronic Articles in Computer and Information Science (2000), 5 (5).
8. Cole, R. J. and Eklund, P. W. Scalability in Formal Concept Analysis. Computational Intelligence (1999), 15 (1), pp. 11-27
9. Cole, R., Eklund, P. and Amardeilh, F. Browsing Semi-structured Texts on the web using Formal Concept Analysis. Web Intelligence (2003).
10. Docco Project home page. http://tockit.sourceforge.net/docco/
11. Godin, R., Missaoui, R. and April, A. Experimental Comparision of Navigation in a Galois Lattice with Conventional Information Retrieval Methods. Int. J. Man-Machine Studies (1993) 38,747-767.
12. Godin, R., Gecsel, J. and Pichet, C. Design of a Browsing Interface for Information Retrieval. In 12th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, (Cambridge, MA, 1989), ACM SIGIR Forum, 32-39.

13. Peñas, A., Verdejo, F., Gonzalo, J. Terminology Retrieval: towards a synergy between thesaurus and free text searching. Advances in Artificial Intelligence - IBERAMIA 2002. Ed. F.J. Garijo, J.C. Riquelme, M. Toro. LNAI **2527**. Springer-Verlag 2002.
14. Peñas, A., Verdejo, F., Gonzalo, J. Corpus-based terminology extraction applied to information access. Proceedings of the Corpus Linguistics 2001, Technical Papers, Special Issue. University Centre for Computer Corpus Research on Language, Lancaster University. **13**, 458–465, 2001.
15. Peñas, A., Gonzalo, J., Verdejo, F., Cross-Language Information Access through Phrase Browsing. Applications of Natural Language to Information Systems. Proceedings of 6th International Workshop NLDB 2001, Madrid, **P-3**, 121–130. Lecture Notes in Informatics (LNI), Series of the German Informatics (GI-Edition). 2001.
16. Priss, U. Lattice-based Information Retrieval. Knowledge Organization (2000), 27 (3), p. 132-142.