Browsing Search Results Via Formal Concept Analysis: Automatic Selection of Attributes

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

Departamento de Lenguajes y Sistemas Informáticos.
E.T.S.I. Informática. Universidad Nacional de Educación a Distancia (UNED)
{juanci,julio,anselmo,felisa}@lsi.uned.es
http://nlp.uned.es

Abstract. This paper presents the JBraindead Information Retrieval System, which combines a free-text search engine with online Formal Concept Analysis to organize the results of a query. Unlike most applications of Conceptual Clustering to Information Retrieval, JBraindead is not restricted to specific domains, and does not use manually assigned descriptors for documents nor domain specific thesauruses. Given the ranked list of documents from a search, the system dynamically decides which are the most appropriate attributes for the set of documents and generates a conceptual lattice on the fly. This paper focuses on the automatic selection of attributes: first, we propose a number of measures to evaluate the quality of a conceptual lattice for the task, and then we use the proposed measures to compare a number of strategies for the automatic selection of attributes. The results show that conceptual lattices can be very useful to group relevant information in free-text search tasks. The best results are obtained with a weighting formula based on the automatic extraction of terminology for thesaurus building, as compared to an Okapi weighting formula.

1 Motivation

Clustering techniques, which are a classic Information Retrieval (IR) technique, are only now becoming an advanced feature of Web search engines. *Vivisimo* (www.vivisimo.com), for instance, performs a standard web search and then provides a hierarchical clustering of the search results in which natural language expressions label each node. The results of the search "jaguar" with Vivisimo automatically displays a taxonomy of results with nodes such as "Car" (referring to the Jaguar car brand), "Mac OS X" (also known as *Jaguar*), or "animal", which permit a fast refinement of the search results according to the user needs. If the user, for instance, expands the node "animal", results are classified in four subclusters, namely "wild life", "park zoo", "adventure amazon" and "other topics". Other examples of clustering techniques in the web include the *Altavista* (www.altavista.com) search engine, which displays a set of suggestions for query refinement that produce a similar clustering effect, and the *Google news service* (news.google.com), where news from hundreds of web servers are automatically grouped into uniform topics.

A common feature of such web services is that clustering is applied to a small set of documents, which come as a result of a query (in search engines) or filtering profile. At this level, clustering proves to be an enabling search technology halfway between browsing (as in web directories, e.g. Yahoo.com or dmoz.org) and querying (as in Google or Altavista). Pure browsing is useful for casual inspection/navigation (i.e., when the information need is vague), and querying is useful when the information need is precise (e.g. I am looking for a certain web site). Probably the majority of web searches lie somewhere between these two kinds of search needs, and hence the benefits of clustering may have a substantial impact on user satisfaction.

A natural question arises: can Formal Concept Analysis (FCA) be applied to browse search results in a free text IR system? FCA is a conceptual clustering technique that has some advantages over standard document clustering algorithms: a) it provides an intensional description of each cluster, which makes groupings more interpretable, and b) cluster organization is a lattice, rather than a hierarchy, facilitating recovery from bad decisions while exploring the hierarchy and, in general, providing a richer and more flexible way of browsing the document space than hierarchical clustering.

The idea of applying FCA only to a small subset of the document space (in our case, the results of a search) eliminates some of the problems associated to the use of FCA in Information Retrieval:

- FCA is computationally more costly than standard clustering, but both can be equally applied to small sets of documents (in the range of 50-500) efficiently enough for online applications.
- Lattices generated by FCA can be big, complex and hence difficult to use for practical browsing purposes. In particular, it can produce unmanageable structures when applied to large document collections and rich sets of indexing terms. Again, this should not be a critical problem when the set of documents is restricted in size and topic by a previous search over the full document collection.

But 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 (see section on *Related Work*). Is it viable and useful to apply FCA without such manually built knowledge?

The JBraindead system, which combines free-text searching with FCA on search results, is a prototype Information Retrieval system that serves as a testbed to investigate this research question. In this paper, we focus on the first essential aspect on the application of FCA to free-text searching: *what is the optimal strategy for the automatic selection of document attributes?*

In order to answer this question, we first need to define appropriate evaluation metrics for conceptual lattices in free-text search tasks, and then compare alternative attribute selection strategies with such metrics. Therefore, we will start by defining two different metrics related to the user task of finding relevant information: 1) a *lattice distillation factor* measuring how well the document clusters in the lattice prevent the user from accessing irrelevant documents (compared to the original ranked list returned by the search engine), and 2) a *lattice browsing complexity* measuring how many node descriptions have to be examined to reach all the relevant information. An optimal lattice will have a high distillation factor and a low browsing complexity. With these measures, we will compare two different attribute selection criteria: a standard IR weight (Okapi) measuring the discriminative importance of a term with respect to the collection, and a "terminological weight" measuring the adequacy of a term to represent the content of a retrieved subset as compared to the full collection being searched. We will also study which is the adequate number of attributes to build an optimal conceptual lattice for this kind of task.

The paper is organized as follows: Section 2 provides a brief description of the functionality and architecture of the JBraindead system. Section 3 summarizes the experimental setup and the results obtained; Section 4 reviews related work, and Section 5 offers some conclusions and discusses future work.

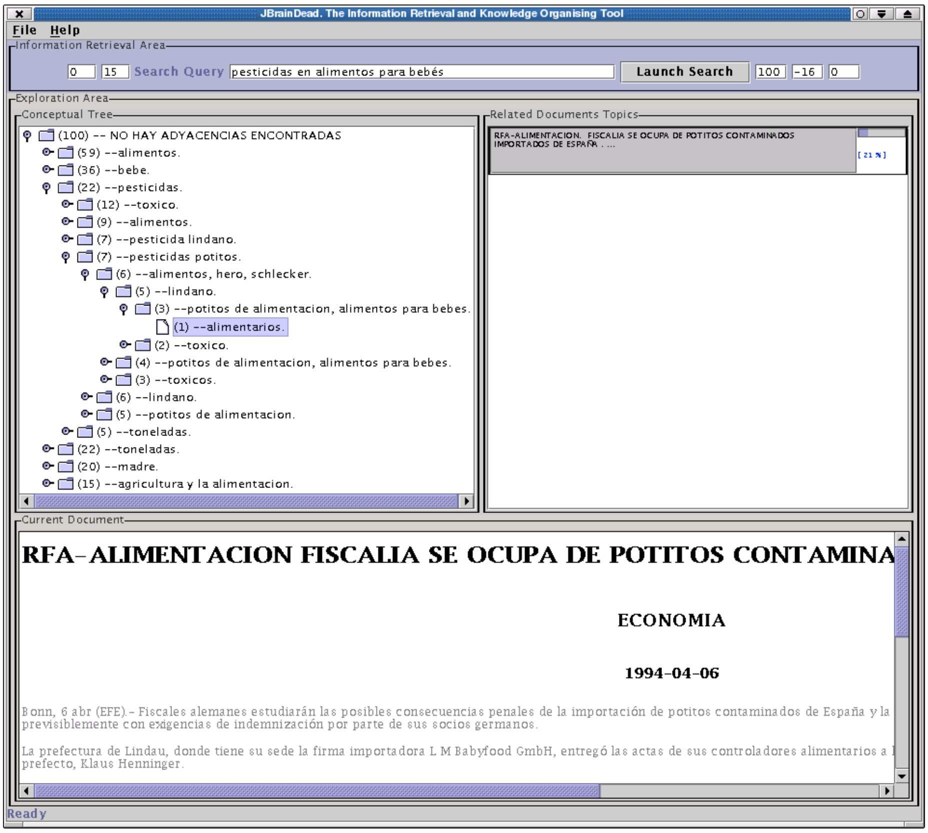
2 The JBraindead Information Retrieval and Clustering System

JBraindead is a prototype IR system that applies Formal Concept Analysis to organize and cluster the documents retrieved by a user query:

1. Free-text documents and queries are indexed and compared in a vector space model, using standard *tf*idf* weights. For a given query, a ranked list of documents is retrieved using this model.
2. The first *n* documents in the ranked list are examined to extract, from the terms in the documents, a set of *k* optimal descriptors according to some relevance weighting formula.
3. Formal Concept Analysis is applied to the set of documents (as formal objects), where the formal attributes of each document are the subset of the *k* descriptors which are contained in its text.
4. Besides the intensional characterization of each concept node, an additional description is built with the most salient phrasal expressions including one or more query terms. This additional characterization is intended to enhance node descriptions for the query-oriented browsing task that conceptual lattices play in JBraindead.
5. The resulting annotated lattice is presented to the user, which can browse the top *n* results by traversing the lattice and/or refine the query at some point. In its current implementation, query refinement can only be made as a direct query reformulation.

The core of the process lies in steps 2 and 4. Step 2 determines the attribute set for a given document set, and then, implicitly, also defines the conceptual lattice. Step 4 enriches the intensional description of concept nodes with query-related phrases, defining how the lattice nodes will be presented to the user.

Figure 1 shows the JBraindead interface for the results of the query "*pesticidas en alimentos para bebés*" (pesticides in baby food) when searching the standard CLEF collection of news in Spanish (see next section for details). Both "*pesticidas*", "*alimentos*" and "*bebé*" appear as second-level nodes in the conceptual lattice. Other attributes automatically selected by JBraindead are "*potitos*" (a kind of baby food which happened to be recipient of pesticides), "*lindano*" (the kind of toxic waste found in the baby food), "*Hero*" (a baby food brand), or "*toxicos*" (toxic). JBraindead also extracts complex node descriptions including node attributes, such as “*alimentos para bebés*” (baby food) or “*agricultura y alimentación*” (food and agriculture).



0 | 15 | Search Query: pesticidas en alimentos para bebés | 100 | -16 | 0

```
(100) -- NO HAY ADYACENCIAS ENCONTRADAS
    (59) --alimentos.
    (36) --bebe.
    (22) --pesticidas.
        (12) --toxico.
        (9) --alimentos.
        (7) --pesticida lindano.
        (7) --pesticidas potitos.
            (6) --alimentos, hero, schlecker.
                (5) --lindano.
                    (3) --potitos de alimentacion, alimentos para bebes.
                        (1) --alimentarios.
                    (2) --toxico.
                (4) --potitos de alimentacion, alimentos para bebes.
                (3) --toxicos.
            (6) --lindano.
            (5) --potitos de alimentacion.
        (5) --toneladas.
    (22) --toneladas.
    (20) --madre.
    (15) --agricultura y la alimentacion.
```

RFA-ALIMENTACION. FISCALIA SE OCUPA DE POTITOS CONTAMINADOS IMPORTADOS DE ESPAÑA . ... [21 %]

RFA-ALIMENTACION FISCALIA SE OCUPA DE POTITOS CONTAMINA

ECONOMIA

1994-04-06

Bonn, 6 abr (EFE).- Fiscales alemanes estudiarán las posibles consecuencias penales de la importación de potitos contaminados de España y la
previsiblemente con exigencias de indemnización por parte de sus socios germanos.

La prefectura de Lindau, donde tiene su sede la firma importadora L M Babyfood GmbH, entregó las actas de sus controladores alimentarios a l
prefecto, Klaus Henninger.

Fig. 1. JBraindead system results for the query “pesticidas en alimentos para bebés” (pesticides in baby food) and the CLEF EFE 1994 news collection.

3 Experiments in Attribute Selection

This section describes a set of experiments designed to find automatically optimal attributes for the set of documents retrieved for a given query in the JBraindead system. We describe a) the Information Retrieval testbed used in our experiments, b) a set of measures proposed to evaluate the quality of conceptual lattices for the purposes of grouping free-text search results, c) the experiments carried out, and, finally, we discuss the results obtained.

3.1 Information Retrieval Testbed

A manual, qualitative inspection of the lattices generated by JBraindead on the results of random queries can provide an initial feedback on the quality of the process. But for a systematic comparison of approaches, an objective measure is needed. While the final purpose of the system is to improve user searches, studies involving users are costly and should only be performed for final testing of already optimized alternatives; hence, we wanted to find an initial experimental setup in which we could tune the process of selecting document attributes before performing user studies.

The Spanish CLEF EFE-1994 collection that we have used during system development includes a set of 160 TREC-like topics (used in CLEF 2001, 2002 and 2003 evaluation campaigns) with manual relevance assessments from a rich and stable document pool [13], forming a reliable and stable test bed for document retrieval systems. Out of this set, we have used topics 41-87 coming from the CLEF 2001 and 2002 campaigns.

If we feed JBraindead with CLEF topics, we can study how the conceptual lattices group relevant and non relevant documents. The baseline is the ranked list of documents retrieved by the initial search: to discover all relevant documents in the set, the user would have to scan at least all documents until the last relevant document is identified. If the FCA process group relevant documents and the node descriptions are useful indicators of content, then browsing the lattice for relevant documents could provide the same recall while scanning only a fraction of the initial set of retrieved documents, saving time to the user and offering a structured view of the different subtopics among relevant documents.

3.2 Evaluation Measures

How can we measure quantitatively the ability of the FCA process to group relevant documents together? A couple of standard clustering measures are *purity* and *inverse purity.* Given a manual classification of the documents into a set of labels, the precision of each cluster P with respect to a label partition L (containing all documents assigned to the label), the *precision* of P is the fraction of documents in P which belong to L. The *purity* of the clustering is then defined as the (weighted) average of the maximal precision values of each cluster P, and the *inverse purity* is defined as the weighted average of the maximal precision values of each partition L over the clusters. Purity achieves a maximal value of 1 when every cluster has one single document, and inverse purity achieves a maximal value of 1 when there is only one single cluster.

Purity and inverse purity are, then, inadequate measures for the conceptual clustering generated by FCA: the cluster structure of a conceptual lattice is much richer than a plain set of labels; and, in addition, the only distinction that we can make for this experiment is between relevant and non relevant documents. What we want to measure is whether the lattice structure effectively "distillates" relevant documents together, allowing the user to locate relevant information better and faster than in a ranked list of documents. Hence we introduce here a "*lattice distillation factor*" measure which relies on a notion of *minimal browsing area* that we introduce now.

3.2.1 Lattice Distillation Factor

Let C be the set of nodes in a conceptual lattice, where documents are all marked as relevant or non-relevant for a given query. Let us assume that, when visiting a node, the user sees the documents for which the node is their object concept. We will use the term *relevant concept* to denote object concepts generated by, at least, one relevant document, and *irrelevant concept* to denote object concepts generated only by one or more irrelevant documents.

We define $C_{REL} \subseteq C$ as the subset of relevant concepts in the lattice. In order to find all relevant documents displayed in the lattice, the user has to examine, at least, the contents of all concepts in C_{REL}. We define the **minimal browsing area** (*MBA*) as the minimal part of the lattice that a user should explore, starting from the top node, to reach all the relevant concepts of C_{REL}, minimizing the number of irrelevant documents that have to be inspected to obtain all the relevant information. We can think of the precision of the MBA (ratio between relevant documents and overall number of documents in the MBA) as an upper bound on the capacity of the lattice to "distillate" relevant information from the search results. The lower bound is the precision of the original list: the user has to scan all documents retrieved to be sure that no relevant document is being missed from that list.

The **lattice distillation factor** (*LDF*) can then be defined as the potential precision gain between the lattice and the ranked list, i.e., as the percentual precision gain between the minimal browsing area and the original ranked list:

$$LDF(C) = \frac{\text{Precision}_{MBA} - \text{Precision}_{RL}}{\text{Precision}_{RL}} \cdot 100 \qquad \textbf{(1)}$$

Note that the minimal browsing area and the distillation factor can be equally applied to hierarchical clusters or any other graph grouping search results.

The only difficulty to calculate the distillation factor lies in how to find the minimal browsing area for a given lattice. In order to calculate this area, we will create an associated graph were all nodes are relevant concepts, and where the cost associated to each arc is related to the number of irrelevant documents which will be accessed when traversing the arc. Then we will calculate the minimal span tree for such graph, which will give the minimal browsing area:

1. We start with the original lattice (or any directed acyclic graph). We define the *cost* of any arc reaching a relevant o irrelevant concept node, from one of its upper neighbors, as the number of irrelevant documents that are fully characterized by the node. E.g., if we have an object concept c, such as, $\gamma d_1 \equiv \gamma d_2 \equiv \gamma d_3 \equiv c$, where d_1 and d_2 are non-relevant documents, all arcs reaching c will have a cost of 2.
2. In a top-down iterative process, we will suppress all nodes which are not relevant concepts. In each iteration, we select the node j which is closest to the top and is not a relevant concept. j is deleted and, to keep connections between ancestors and descendants of the node, we create a new arc for every pair of nodes $(u,l) \in UjXLj$, where Uj and Lj are the sets of upper and lower neighbors of j. A cost of $cost(u,l) = cost(u,j) + cost(j,l)$ is then assigned to the new arc. If we end up with more than one arc for a single pair of nodes (u,l), we select the arc with the lowest cost and suppress the others.

3. The result of the iteration above is a directed acyclic graph whose nodes are all relevant concepts. The minimal span tree of this new graph tells us which is the minimal browsing area in the original lattice.

Figure 3 shows an example of how to build the minimal browsing area and calculate the lattice distillation factor.

3.2.2 Lattice Browsing Complexity

The *distillation factor* is only concerned with the cost of reading documents. But browsing a conceptual structure has the additional cost (as compared to a ranked list of documents) of examining node descriptions and deciding whether each node is worth exploring. For instance, a lattice may lead us to ten relevant documents and save us from reading another ten irrelevant ones... but force us to traverse a thousand nodes to find the relevant information! Therefore, the number of nodes in the lattice has to be considered to measure its adequacy for searching purposes.

There might be also the case that a lattice has a high distillation factor but a poor clustering, forcing the user to consider most of the nodes in the structure. An example can be seen in Figure 2, where all the object concepts occur near the lattice bottom. Precision for the minimal browsing area is 1, and the lattice distillation factor is 100%. The clustering, however, is not good: the user has to consider (if not explore) all node descriptions to decide where the relevant information is located.

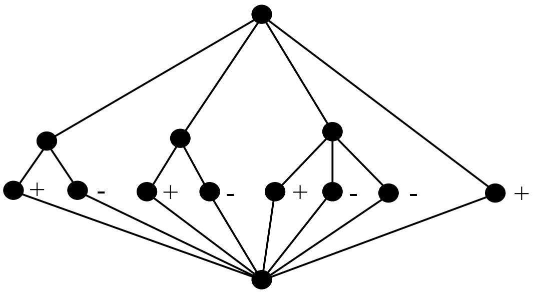
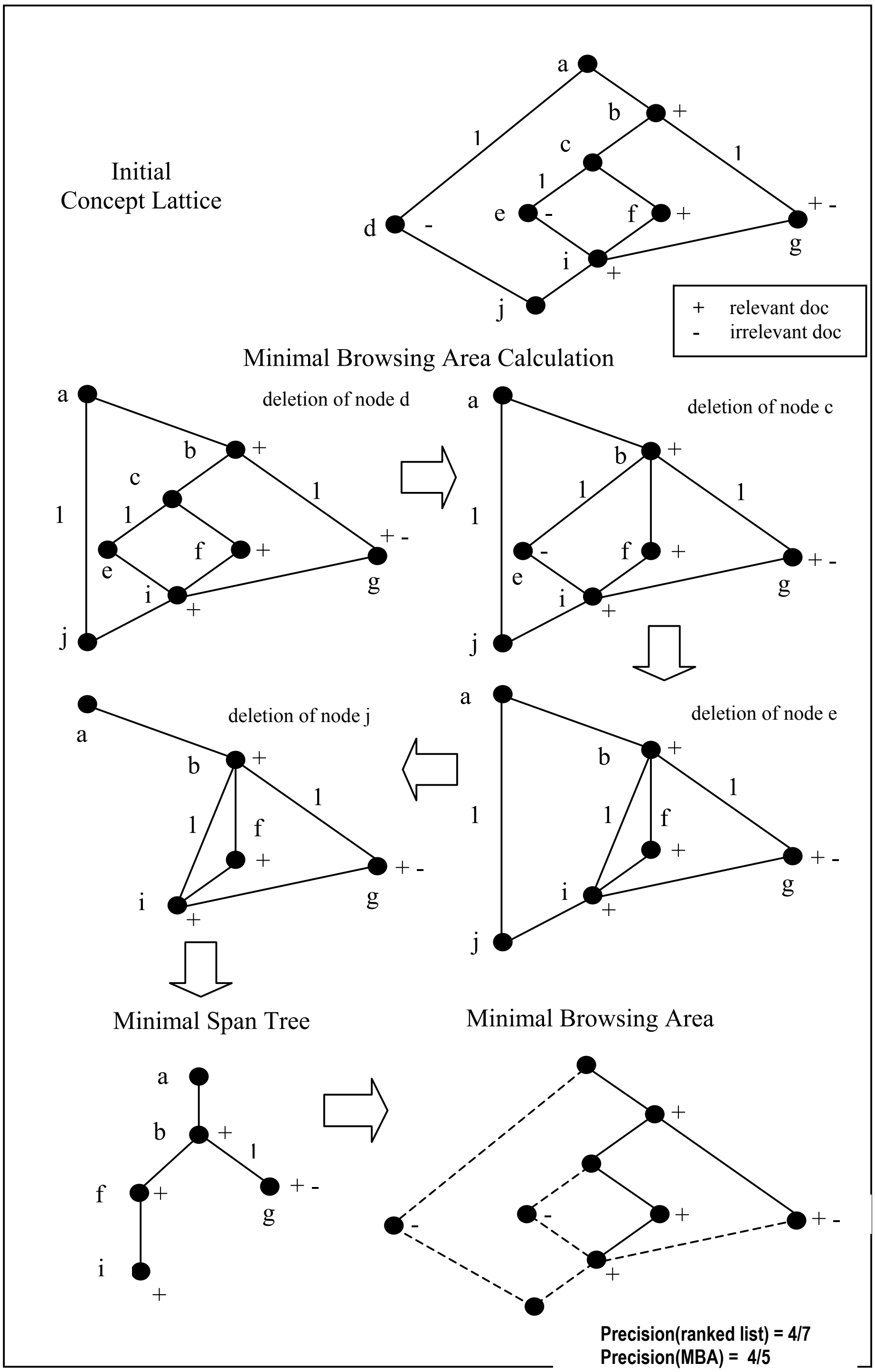


Fig. 2. This lattice has a high distillation factor (LDF = 100%), but the clustering is poor.

We need to introduce, then, another measure estimating the percentage of nodes that must be considered (rather than visited) in a lattice in order to reach all relevant information. We propose a measure of the *lattice browsing complexity* (*LBC*) as the proportion of nodes in the lattice that the user sees when traversing the minimal browsing area. The idea is that, when a node is explored, all its lower neighbors have to be considered, while only some of them will be in turn explored.

Being C the set of nodes in the concept lattice, the set of viewed nodes C_{VIEW} is formed by the lower neighbors of each node belonging to the minimal browsing area. The lattice browsing complexity is the percentage of lattice nodes that belong to C_{VIEW}: $LBC(C) = | C_{VIEW} |/|C|*100.$ Figure 4 shows an example of how the lattice browsing complexity is calculated.



Precision(ranked list) = 4/7
Precision(MBA) = 4/5
LDF = (4/5-4/7)/(4/7)*100=40%

Fig. 3. Calculation of the Lattice Distillation Factor.

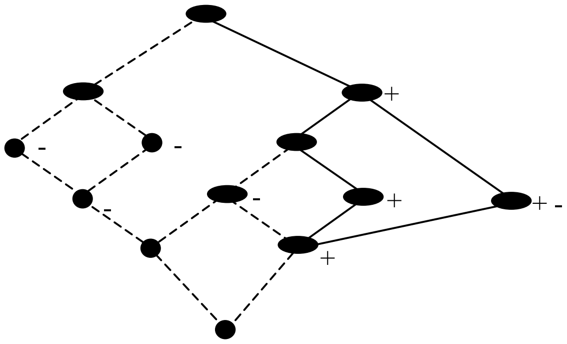


Fig. 4. Concept Lattice with LBC = 61%. *MBA* links are represented as continuous lines on the concept lattice. *LBC* nodes are drawn as oval nodes. Circular nodes represent nodes which are not seen when traversing the *MBA*.

3.3 Experiments

We have used CLEF topics 41-87, corresponding to the CLEF 2001 campaign. For each experiment, all topics (title+description) are searched. For every search, a formal context *K=(G,M,I)* is build, where *G* is the set of the first 100 documents returned by the search engine in response to a query, *M* is the set of attributes (variable between experiments), and *d I t* iff the attribute *t* is a term occurring in document *d*.

The two weighting measures used to generate the formal contexts are the Okapi weighting scheme and the terminological formula proposed in [12].

3.3.1 Okapi

Term weights in an IR system measure the importance of a term as a discriminative descriptor for a given document. We have selected the Okapi BM25 weight, which has given the best results for the CLEF collection used in our experiments [15]:

$$w_i = \frac{(k_1+1)\cdot tf_i}{K+tf_i}\cdot\frac{\ln\left(\frac{N+0.5}{f_i}\right)}{\ln(N+1)} \tag{2}$$

$$K = k_1\cdot\left[(1-b)+b\cdot\frac{l_{ret}}{avdl}\right] \tag{3}$$

where the parameters k_1, b and *avdl* are adjusted for the Spanish CLEF test collection with the values $b = 0.5$, $k_1 = 1.2$, and $avdl = 300$ taken from [15]. tf_i represents the frequency of the term i in the document (in our case, in the set of retrieved documents), and f_i the document frequency of term i in the whole collection. Finally, l_{ret} represents the total length (in terms) of the retrieved document set.

3.3.2 Terminological Weight Formula

A terminological weight is designed to find, in a collection which is representative from some specific domain, which terms are more suitable as descriptors for a thesaurus of the domain. We use a variation of a formula introduced in [12] which compares the domain-specific collection with a collection from a different domain, and assigns a higher weight to terms that are more frequent in the domain-specific collection than in the contrastive collection. In our case, the domain-specific collection can be the set of retrieved documents, and the contrastive collection is the whole document collection minus the retrieved set:

$$w_i = 1 - \frac{1}{\log_2\left(2 + \frac{tf_{i,ret} \cdot f_{i,ret} - 1}{tf_{i,col} + 1}\right)} \qquad \textbf{(4)}$$

where w_i is the terminological weight of term i, $tf_{i,ret}$ is the relative frequency of term i in the retrieved document set, $f_{i,ret}$ is the retrieved set document frequency of term i, and $tf_{i,col}$ is the relative frequency of term i in the whole collection minus the retrieved set.

3.3.3 Number of attributes

Finally, for the formulas above we have studied how the number of attributes affects the size and quality of the conceptual lattice. After some initial testing, we have kept the number of attributes between 10 and 20: with less attributes, the clustering capacity is too low, and with more than 20 attributes, the number of nodes becomes exceedingly high for browsing purposes, and the computational cost makes online calculation too slow.

3.4 Results

Table 1 shows the basic outcome of our experiments. In all cases, the Distillation Factor is high, ranging from 346% to 594%. Note that this measure is an upper bound on the behavior of real users: only an optimal traversing of the lattice will give such relative precision gains. Note also that the microaveraged LDF is much higher than would result from the average precisions of the ranked list and minimal browsing area. This is because the LDF expresses the relative precision gain rather than the absolute precision gain.

For 10 attribute terms, the Okapi formula gives a higher Distillation Factor (580% versus 346%) but at the cost of a much larger lattice (70 nodes in average versus 35 nodes with the terminological formula). Both differences are statistically significant according to a paired t-test ($p<0.05$). In practice, the Okapi formula generates too large lattices for browsing a hundred documents, hence the terminological formula should give better results with experiments involving users.

The LDF seems to grow linearly with the number of attributes, and the complexity factor seems to decay linearly the number of attributes. The number of nodes, however, grows almost exponentially. For 15 terms, the number of nodes generated by the terminological formula is already too large (94 nodes) for practical purposes.

Overall, it seems clear that conceptual lattices can be very effective to group relevant information, and the grouping effect is higher for larger attribute spaces. But the number of nodes quickly becomes impractical. From this point of view, understading attributes as potential terminological units seems to give more compact lattices than seeing attributes as IR indexing terms.

Table 1. Experimental results. The average precision of the original ranked lists was 0.17.

	# Terms	Prec. MBA	LDF	# Nodes	# Nodes Viewed MBA	LBC
Terminological Formula	10	0.35	346 %	35	19	54 %
	15	0.43	493 %	94	50	43 %
	20	0.52	594 %	184	65	36 %
Okapi	10	0.43	580 %	70	32	44 %

4 Related work

The application of FCAs to Information Retrieval is an increasingly successful field, which has already produced some commercial applications, although all research known to us concentrates on manually (or semi automatically) indexed or classified according to some domain specific thesaurus or classification scheme.

Two early applications for which empirical tests with users were conducted are [11] and [1]. In [11], navigation in a Galois lattice is compared to boolean retrieval and hierarchical navigation in an Information Retrieval task involving users. In the experiment, recall obtained using lattices and boolean retrieval is superior to navigation in a hierarchy. The document collection consisted on 113 short animation film descriptions, and every document was manually indexed by an average of 6.53 classification terms. In [1], a lattice conceptual clustering system is proposed that incorporates background knowledge from the indexing thesaurus (i.e. the broader/narrower term relationships in the thesaurus) into the process of building the conceptual clustering lattice. Browsing with and without the background knowledge were compared in the context of users searchers against a collection of 1555 documents about Artificial Intelligence extracted from a computer engineering collection (INSPEC). Browsing with background knowledge led a 30% relative improvement in recall, showing that the incorporation of specificity relations between indexing terms is a significant improvement over building the lattice without considering the relations between the keywords manually assigned to documents.

One of the application domains that has received more attention is medical documentation. In [3], a set of 9000 patient medical discharge summaries are indexed using SNOMED (Systematized nomenclature of medicine), showing the viability of the approach. The approach has a continuation in [4,2] and [5], where documents are automatically indexed using UMLS (Unified Medical Language System) metathesaurus terms, and the notions of *conceptual scales* and *purified contexts* are introduced for improved, scalable knowledge visualization. Unfortunately, no empirical, quantitative evaluations or user studies have been conducted in this domain, to our knowledge.

FCA has also been applied to document retrieval in conjunction with *faceted thesauruses,* a notion which is related to conceptual scales, in which different aspects of an article description (for instance, the topic of an article and the level of difficulty) have descriptors in different facets of the thesaurus. The IR system *FaIR* [14] is an example of such a system, which is applied to an on-line collection of about 5000 FAQ documents of computing questions. Another application in the computer domain is *Aran* [9], an Information Help System that applies FCA to Unix man pages. A characteristic feature of this system is that it does not employ any prior thesaurus; indexes are obtained from free text in the short (one or two lines) command descriptions that summarize every unix command. As in the medical domain, none of these systems have been quantitatively evaluated.

An attractive example of the possibilities of FCA for knowledge management is the HIERMAIL system [8,6], which provides a structured ontology and IR system for Email search and discovery, in which the principles of FCA are supported by an inverted file index that provides efficient client iteration. Although there is no empirical evaluation of the utility of the system (perhaps because it is not trivial to design an evaluation for knowledge management tasks), an indirect evidence of its value is that the idea of applying FCA to e-mail management has already reached the market with the Mail-Sleuth application (http://mail-sleuth.com).

More recently, [7] combine Information Extraction on web documents with FCAs in an information access application on the domain of classified advertisements for Real Estate properties. Rather than simply extracting keywords from documents, the Information Extraction process extracts template-based data to describe advertisements, improving the input to the FCA process.

Finally, a work which is similar in spirit to the JBraindead approach is described in [10]. The authors cluster a news collection (Reuters-21578) combining a standard clustering technique, which is applied to the whole collection, with FCA, which is applied individually to every cluster produced in the first process. One of the salient features of the system is that they use a general purpose lexical knowledge base (WordNet) rather than a domain specific thesaurus as background knowledge, both for the initial clustering process and for the subsequent Conceptual Clustering step. In practice, that means that the input for FCA is closest to free indexing terms than in any of the applications mentioned above. JBraindead uses a similar approach, but in an IR application: Hotho and Stumme apply FCA to smaller subsets of the collection by applying a standard clustering technique, and then performing FCA on every cluster returned; JBraindead applies FCA to smaller subsets of the collection by applying standard IR, and then performing FCA online on the results of the search. It is worth mentioning that in Hotho and Stumme's work, the indexes for the FCA process are the terms with higher values in the centroid vector representing the cluster. The combination of WordNet and centroid vectors is an interesting alternative to the methods evaluated in this paper, which we seek to adapt and compare with our current keyword extraction procedures.

5 Conclusions and Future Work

We have described the JBraindead Information Retrieval system, which combines standard IR techniques with online conceptual clustering applied on the results of the initial user query. The system is domain independent and operates without resorting to thesauruses or other predefined sets of indexing terms. Hence, the contributions of JBraindead to the application of FCA in Information Retrieval lies in the approaches to extract indexing terms for the FCA process and to build natural descriptions of the nodes in the resulting lattice.

In this paper we have focused on the process of attributes selection, comparing two weighting schemas of different nature: the Okapi probabilistic weights, related to the discriminative power of a term for IR purposes, and a terminological weight related to the adequacy of a term as topic-specific descriptor. We have also measured the influence of the number of attributes in the quality of the outcoming lattice for searching purposes. We have made a special emphasis in the definition of metrics to compare different conceptual structures for the task of browsing free-text results, introducing: a) a *lattice distillation factor*, related to how well the conceptual structure prevents the user from reading irrelevant documents, and b) a *lattice browsing complexity*, related to the proportion of nodes in the structure that have to be considered to reach all relevant information. An optimal lattice will have a high distillation factor, a low browsing complexity and a low number of nodes.

The results show that the terminological weighting is better than the IR Okapi weight, and that an increasing number of attributes improves the distillation factor at the cost of a higher browsing complexity. Most differences between runs are statistically significant, showing that the quality of the conceptual structures is highly sensitive to parameter settings.

The JBraindead system illustrates the scalability of FCA to unrestricted Information Retrieval settings, if it is applied to organize search results, rather than trying to structure the whole document collection with conceptual analysis. To our knowledge, this is the first IR system based on FCA that operates on a collection of more than 500 Mb comprising more than 200,000 documents.

JBraindead provides, as well, a test bed to study optimal querying, indexing, visualization and refinement strategies for free-text retrieval based on conceptual clustering. The experiments reported here are just a first step towards optimal, interactive content retrieval and browsing. We are currently experimenting with shallow Information Extraction techniques (named entity recognition, noun phrase indexing) to reach a selection of terms that can be used both to produce better lattice structures and as natural descriptors of nodes.

6 References

1. Carpineto, C. and Romano, G. A lattice Conceptual Clustering System and Its Application to Browsing Retrieval. Machine Learning (1996) 24, 95-122.
2. Cole, R. J. The management and visualization of document collections using Formal Concept Analysis (2000). Ph. D. Thesis, Griffith University.

3. Cole, R. J. and Eklund, P. W. Application of Formal Concept Analysis to Information Retrieval using a Hierarchically structured thesaurus.
4. 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).
5. Cole, R. J. and Eklund, P. W. Scalability in Formal Concept Analysis. Computational Intelligence (1999), 15 (1), pp. 11-27
6. Cole, R. J., Eklund, P. and Stumme, G. Document Retrieval for Email Search and Discovery using Formal Concept Analysis. Applied Artificial Intelligence (2003), 17 (3)
7. Cole, R., Eklund, P. and Amardeilh, F. Browsing Semi-structured Texts on the web using Formal Concept Analysis. Web Intelligence (2003).
8. Eklund, P. and Cole, R. Structured Ontology and IR for Email Search and Discovery. In Proceedings of the Sixth Australasian Document Computing Symposium (2001), Coffs Harbour, Australia.
9. Fernández-Manjón, B., Cigarrán, J., Navarro, A. and Fernández-Valmayor, A. Applying Formal Concept Analysis to Domain Modeling in an Intelligent Help System. In Proceedings of Information Technology and Knowledge Systems (1998), 5th IFIP World Computer Congress, Vienna-Budapest.
10. Hotho, A. and Stumme, G. Conceptual Clustering of Text Clusters. In Proceedings of the FGML Workshop (2002), Hannover.
11. Godin, R., Missaoui, R. and April, A. Experimental Comparison of navigation in a Galois lattice with conventional Information Retrieval methods. Int. J. Man-Machine Studies (1993) 38, 747-767.
12. Peñas, A., Verdejo, F. and Gonzalo, J. Corpus-Based Terminology Extraction applied to Information Access. In Proceedings of Corpus Linguistics 2001 (2001), Lancaster University.
13. Peters, C., Braschler, M., Gonzalo, J. and Kluck, M (eds.) Evaluation of Cross-Language Information Retrieval Systems (2002). Springer-Verlag LNCS 2406, Berlin.
14. Priss, U. Lattice-based Information Retrieval. Knowledge Organization (2000), 27 (3), p. 132-142.
15. Savoy, J. Report on CLEF 2002 experiments: Combining multiple sources of evidence. In Peters, C., Braschler, M., Gonzalo, J. and Kluck, M. (eds): Advances in Cross-Language Evaluation Retrieval (2003). Springer-Verlag LNCS 2875, Berlin.