

# Exploration of Web Search Results Based on the Formal Concept Analysis

Peter Butka<sup>1</sup>(✉), Thomas Low<sup>2</sup>, Michael Kotzyba<sup>2</sup>, Stefan Haun<sup>2</sup>,  
and Andreas Nürnberger<sup>2</sup>

<sup>1</sup> Department of Cybernetics and Artificial Intelligence, Faculty of Electrical Engineering and Informatics, Technical University of Kosice, Kosice, Slovakia  
`peter.butka@tuke.sk`

<sup>2</sup> Data and Knowledge Engineering Group, Faculty of Computer Science,  
Otto von Guericke University Magdeburg, Magdeburg, Germany  
{thomas.low,michael.kotzyba,stefan.haun,andreas.nuernberger}@ovgu.de

**Abstract.** In this paper, we present an approach to support exploratory search by structuring search results based on concept lattices, which are created on the fly using advanced methods from the area of Formal Concept Analysis (FCA). The main aim of the approach is to organize query based search engine results (e.g. web documents) as a hierarchy of clusters that are composed of documents with similar attributes. The concept lattice provides a structured view on the query-related domains and hence can improve the understanding of document properties and shared features. Additionally, we applied a fuzzy extension of FCA in order to support the usage of different types of attributes within the analyzed query results set. The approach has been integrated into an interactive web search interface. It provides a smooth integration of keyword-based web search and interactive visualization of concept lattice and its concepts in order to support complex search tasks.

## 1 Introduction

One of the challenges in information retrieval (IR) is to provide a structured view on retrieved results, which allows to better navigate within document subsets and to more easily explore the analyzed domain. While linear visualizations of result sets, i.e. simple lists of results, are still commonly used, they do not convey information about the structure of result sets, e.g. clusters of similar results. In comparison, a well-organized representation can help users to better understand the current domain of their interest and discover relationships between the retrieved documents. For this goal, we decided to study possible applications of methods of exploratory data mining known as Formal Concept Analysis (FCA) [1]. Users will be supported by an interactive visualization of a hierarchical structure of document clusters - subsets of documents with shared attributes defined within the domain. This approach can be used to support a user during an exploratory search scenario [2,3], where the user needs to get an overview of some topic or domain. Such tool can be useful in a companion technology to support interactive information access and organization.

FCA provides methods that transform input data tables into concept hierarchies known as concept lattices. It has been successfully applied in several areas like conceptual data analysis, data and text mining, or information retrieval. In comparison to ontology it provides a dynamic view of actual input data table for user-specified query. In its basic version (the crisp case), FCA processes input data tables in the form of object-attribute models, also known as formal contexts. They are represented by binary relations, i.e., an object has or does not have an attribute value. Of course, there are many examples where it is natural to use formal contexts with multi-valued attributes. Hence, extensions of FCA were proposed in order to process object-attribute models, where the relationship between objects and their attributes is based on fuzzy relations, cf. [4–7].

For the purposes of practical data mining analysis, certain fuzzy FCA models, known as one-sided concept lattices, were introduced. In this case, object clusters are considered as ordinary subsets (as in the crisp case), while attributes are assumed to obtain fuzzy values [8]. In order to work with input data tables with different types of attributes an approach called Generalized One-Sided Concept Lattice (GOSCL) was introduced in [9]. This model produces a hierarchical structure of concepts from heterogeneous object-attribute models in the form of a concept lattice, which can be used in several applications for the analysis of input data. Although outputs of FCA-based methods in the form of concept lattices provide valuable information about object-attribute models, in many cases the resulting structure is too complex and hence a reduction is needed. Several approaches were already introduced, cf. [10–12]. Also for GOSCL, some methods were already tested for the reduction of concept lattices based on different ranking methods using support, stability, confidence, or other metrics, cf. [13–16].

The main goal of this paper is to apply methods like GOSCL to result sets of web search queries in order to get a structured view of the searched domain. When analyzing documents acquired from web engines using FCA-based methods, we need to prepare input tables in the form of the object-attribute model. In this case, objects are documents from the query result set and attributes are descriptive elements or features of these documents, e.g., the existence of a word within the document snippet, their importance measured by the *tf-idf* score, metadata of the document, etc. Then, a concept lattice can be created and visualized to the user. However, showing a concept lattice to the user is not straightforward due to the complex nature of its graph-like structure. Our goal is to support a user in interactively exploring a search domain. We hypothesize that visualizing search results as a concept lattice would support users during the process of search and exploration. In our first experiments, we integrated FCA-based methods into the graph visualization tool known as the creative exploration toolkit (CET) [17]. The main idea and some first work-in-progress details were introduced in [18]. In order to extend our system, we decided to re-use the back end of CET and its search functionality and add new interactive visualizations based on selected Javascript libraries (D3<sup>1</sup> and gojs<sup>2</sup>).

<sup>1</sup> D3: <https://d3js.org/>.

<sup>2</sup> gojs: <https://gojs.net/>.

In comparison to our previous work, we were able to extend the features of our visualizations and therefore gain a better understanding how users benefit from the visualized concept lattice of search results.

The remainder of this paper is organized as follows: Sect. 2 provides information on related work, introducing how FCA methods can be used in information retrieval tasks and extended to fuzzy FCA approaches. Section 3 describes necessary details regarding the theory of FCA, especially details regarding the GOSCL model and notes on the reduction of concept lattices. Section 4 is devoted to our proposed approach and designed visualizations. Section 5 describes our experiments with the system and discusses results. Finally, we draw conclusions and present ideas for future work.

## 2 Related Work and Motivation

FCA-based methods were already used in information retrieval systems and tasks. A survey on this topic can be found in [19]. Usually, approaches are divided into:

- *Direct usage for exploration:* A concept lattice is created and directly used for search and navigation in the space of hierarchically organized clusters of documents acquired from the query result set.
- *Usage for reduction:* The original concept lattice is first reduced (in order to simplify results for better and more understandable visualization) and then it is used for search and navigation.
- *Usage as a ranking function:* Methods from FCA are used as an analytical function in order to support the process of linear (re)ordering of the results according to the structure of the concept lattice and its features.

Examples are the CREDO system [20] and its extension CreChainDo [21], which visualize a hierarchy of document clusters or concepts. User feedback on the relevance of concepts is used to adapt the hierarchy, i.e., it is an interactive and adaptive exploration tool for documents from query result sets. Concepts are presented and organized in a tree structure, so it is a simplified and reduced version, i.e., a subpart of the concept lattice, which can be built on such data. In its extension, it is able to support a chain of the interactive steps and manage them. Similarly, in [22] an approach is described which uses an FCA lattice for navigation and attributes for query tuning. In [23] an approach for the enhancement of keyword-based search has been proposed, where local FCA models with agglomerative clustering were used in order to obtain a linear ranking of documents based on a combination of full-text and conceptual search. More work on this topic can be found in the aforementioned survey.

The main disadvantage of these approaches is that only the classic framework for FCA is used, i.e., every description of objects and their sets is based on binary attributes. Our vision is to apply a one-sided fuzzy approach, where objects are considered in a binary way. Defined clusters still are subsets, i.e., the object is or is not part of a concept or cluster. However, attributes can obtain fuzzy values. In this case particular documents are described by the set of

different types of attributes, i.e., not only binary attributes are used, but we can describe the documents within the query result set also by some nominal, ordinal, numeric, or general lattice attributes, in order to have a more natural description of them according to user without the possibility for specific preprocessing and transformation (called conceptual scaling) to a binary input data table (which can be less interpretable for users). Here fuzzy approaches, like our generalized one-sided concept lattice (an approach which is able to combine different types of attributes within one heterogeneous data table), can be very useful and their application to interactive exploration should be analyzed. As it was already mentioned, we provided some first work-in-progress version of the system in [18]. Here, we would like to propose an extension of our visualization with the help of specific javascript libraries and some additional experiments with the new system.

### 3 Generalized One-Sided Concept Lattices

Before discussing details on GOSCL [9], we would like to introduce the standard FCA approach. If we have some set of attributes  $A$  and set of objects  $B$ , the fact that some object from  $B$  has some attribute from  $A$  is written in binary incidence relation  $I$ . Then  $(B, A, I)$  is called formal context and it is object-attribute model of input data, where  $B, A \neq \emptyset$  and  $I \subseteq B \times A$ . Then, we are able to define a pair of mappings  $\uparrow : 2^B \rightarrow 2^A$  and  $\downarrow : 2^A \rightarrow 2^B$ , for every  $X \subseteq B$  and  $Y \subseteq A$ , as follows<sup>3</sup>:

$$X^\uparrow = \{y \in A : (x, y) \in I \text{ for all } x \in X\},$$

$$Y^\downarrow = \{x \in B : (x, y) \in I \text{ for all } y \in Y\}.$$

This pair of mappings forms Galois connection, which leads to concept lattice. Consequently there is given a set

$$\mathcal{B}(B, A, I) = \{(X, Y) : X \subseteq B, Y \subseteq A, X^\uparrow = Y, X = Y^\downarrow\}.$$

$\mathcal{B}(B, A, I)$  forms a complete lattice called concept lattice and pairs of subsets  $(X, Y)$  (fixed points of Galois connection) are called formal concepts.

This crisp model can be easily extended to one-sided fuzzy concept lattice, if one-side of the model will be defined as fuzzy set. In [8] author proposed model where the input consists of an  $L$ -context  $(B, A, R)$ , with  $L$  as a complete lattice and relation  $R : B \times A \rightarrow L$  as a binary  $L$ -fuzzy relation. It means that attributes now obtains values from  $L$ . Thanks to one-sided property, it is simple to define a pair of mappings  $\beta : 2^B \rightarrow L^A$  and  $\alpha : L^A \rightarrow 2^B$ , which forms Galois connection (and produces so-called one-sided fuzzy concept lattice) and is defined as follows:

$$\beta(X)(a) = \bigwedge_{b \in X} R(b, a),$$

<sup>3</sup> Note that  $2^S$  denotes the power set of the set  $S$ , i.e., the set of all subsets of  $S$ .

$$\alpha(g) = \{b \in B : \text{for each } a \in A, g(a) \leq R(b, a)\}.$$

A problem of the previous model is that one truth value structure  $L$  is applied for every attribute. Naturally, data tables are often heterogeneous with different types of attributes. Therefore, another generalization (Generalized One-Sided Concept Lattice - GOSCL) was proposed in [9], where every attribute is able to obtain values from its own truth value structure. Now we provide some basic details of GOSCL.

A 4-tuple  $(B, A, L, R)$  is *generalized one-sided formal context* if the following conditions are fulfilled:

- (1)  $B$  is a non-empty set of objects and  $A$  is a non-empty set of attributes.
- (2)  $L : A \rightarrow \text{CL}$  is a mapping from the set of attributes  $A$  to the class of all complete lattices. Hence, for an attribute  $a \in A$ ,  $L(a)$  denotes a complete lattice, which represents a structure of truth values associated to the attribute  $a$ .
- (3)  $R$  is generalized incidence relation (input data table), i.e.,  $R(b, a) \in L(a)$  for all  $b \in B$  and  $a \in A$ . Thus,  $R(b, a)$  represents a degree from the structure  $L(a)$  in which the object  $b$  has the attribute  $a$ .

Similarly to previous approaches, Galois connection will form concept lattice thanks to defined mappings, i.e., we have mappings between the classical subsets of the set of all objects  $\mathbf{P}(B)$  and the direct products of complete lattices  $\prod_{a \in A} L(a)$  presented in a generalized one-sided formal context.

Given a generalized one-sided formal context  $(B, A, L, R)$ , the concept forming operators (which again form Galois connection)  $\perp : \mathbf{P}(B) \rightarrow \prod_{a \in A} L(a)$  and  $\top : \prod_{a \in A} L(a) \rightarrow \mathbf{P}(B)$  are defined by

$$X^\perp(a) = \bigwedge_{b \in X} R(b, a),$$

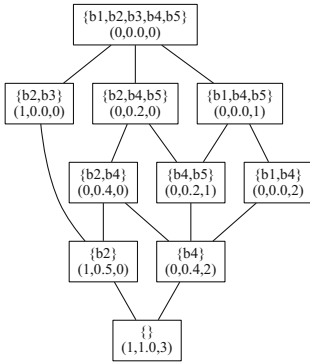
$$g^\top = \{b \in B : \forall a \in A, g(a) \leq R(b, a)\}.$$

Formal concepts are defined as the fixed points of the concept forming operators. Formally,  $\mathcal{C}(B, A, L, R)$  is defined as the set of all pairs  $(X, g)$ , where  $X \subseteq B$ ,  $g \in \prod_{a \in A} L(a)$ , satisfying  $X^\perp = g$  and  $g^\top = X$ . Then  $X$  is called the *extent* and  $g$  *intent* of a concept  $(X, g)$ . Moreover, we can define the partial order on  $\mathcal{C}(B, A, L, R)$  as:

$$(X_1, g_1) \leq (X_2, g_2) \quad \text{iff} \quad X_1 \subseteq X_2 \quad \text{iff} \quad g_1 \geq g_2.$$

Then, if  $(B, A, L, R)$  is a generalized one-sided formal context, the set of all concepts  $\mathcal{C}(B, A, L, R)$  with the partial order defined above forms a complete lattice, which is also called generalized one-sided concept lattice corresponding to the generalized one-sided formal context (object-attribute model)  $(B, A, L, R)$ .

We have also provided incremental algorithm for the creation of GOSCL model in several papers related to GOSCL (including [9]). We will skip it, because it is not necessary for understanding of GOSCL model. Now we add some small



**Table 1.** Input for GOSCL model.

R	$a_1$	$a_2$	$a_3$
$b_1$	0	0.0	2
$b_2$	1	0.5	0
$b_3$	1	0.0	0
$b_4$	0	0.4	2
$b_5$	0	0.2	1

**Fig. 1.** Output GOSCL model.

illustrative example, how input data table and its corresponding GOSCL model looks like.

Let  $B = \{b_1, b_2, b_3, b_4, b_5\}$  are objects and  $A = \{a_1, a_2, a_3\}$  are attributes. Here,  $a_1$  is binary attribute represented by two element chain with values 0 and 1, attribute  $a_2$  is numeric attribute with values from real interval  $\langle 0, 1 \rangle$  and attribute  $a_3$  is ordinal attribute represented by chain lattice with values from 0, 1, 2, 3. Example of data table is presented in Table 1 and corresponding generalized one-sided concept lattice constructed based on this data table is presented in Fig. 1.

One of the problems of FCA-based methods is that number of generated concepts can be large, therefore it is sometimes necessary to use some reduction technique in order to achieve a more comprehensive result or visualization. For the purpose of this work most important methods are ranking methods which reduce concepts using some metric. It means that some concepts are removed according to fact that their value in some criterion is lower than selected threshold. Especially useful for reduction of too specific concepts is the application of support metric. It means that all concepts with a number of objects lower than selected percentage of the number of all objects are removed.

## 4 Our Tool for FCA-Based Exploration of Search Results

The main idea for our approach is to provide a tool, where concept lattices of web search results are provided to the user. Hence, the user is able to explore the clusters of results in a structured way, where groups are composed of results which share some part of attributes similarly to any concept lattice. Additionally, the user is able to navigate through these clusters, select them and browse particular links to original web pages. While there were several approaches that tried FCA-based methods in information retrieval, our approach is unique due to the usage of the whole structure of concept lattice or its reductions, possibility to provide both general and focused view of output structure, as well as the

application of generalized one-sided approach for processing of different types of input attributes.

The basic idea for implementation of the approach was to integrate FCA approach into aforementioned CET Search tool, where the user can interact with the model of query result set and explore the searched domain. CET provides the possibility to an input query, call the search API (e.g., Bing API like in the first prototype, or Google API, etc.), analyze the results, generate the graph of nodes and edges, visualize them and then explore them in the user interface. After the testing of the first prototype, which was described in [18], we decided also analyze other possibilities for extension of the approach. Currently, many javascript libraries provide various interactive visualizations, which could be easily integrated, adapted and provided to users in usual web browser interface (while CET Search tool frontend is a desktop application). Therefore, for new experiments with the FCA-based exploration, we decided to use some of them. For more focused visualization we selected Double tree graph implemented in D3 javascript library, for a general graph with DAG (Directed Acyclic Graph) layout we tried GO.JS project (which is commercial, but it is possible to use it for research or education with some restrictions).

In more details, if we would like to integrate FCA-based methods (especially fuzzy extension) into graph-based visualization, we need to create object-attribute model from query results set (use weighting of terms in results to reach fuzzy extension), then create output model in the form of concept lattice with nodes (representing concepts = clusters of web engine results), connect them using edges (relation of generalization/specification between concepts) to create hierarchy of concepts, and provide interactivity to all visualization of output model. When some reduction is applied, it should be possible to select some threshold or setup.

Now we will describe some details about the current implementation. The process how the system works follow these steps:

1. User input query to system - user is able to select type of visualization graph (global or focused), number of results to process (10 is default), support value for reduction (0 for no reduction is default), check for use of vector (fuzzy) or binary (crisp) model,
2. System will use API (Bing, Google) to get search results (query results set),
3. System will preprocess model (metadata and snippets are extracted, snippets are used to create object-attribute model), create concept lattice (or its reduction) according to settings, prepare JSON file for selected visualization, push JSON to visualize part of application (provide output HTML code),
4. System will provide use fuzzy extension of FCA in order to analyze weighting of terms in results (vector model from extracted words) and/or metadata descriptions of different types,
5. provide the possibility to apply some reduction (like support-based reduction).

In order to illustrate these visualizations, we show two small examples for binary models. First, global visualization of concept lattice (see example in Fig. 2,

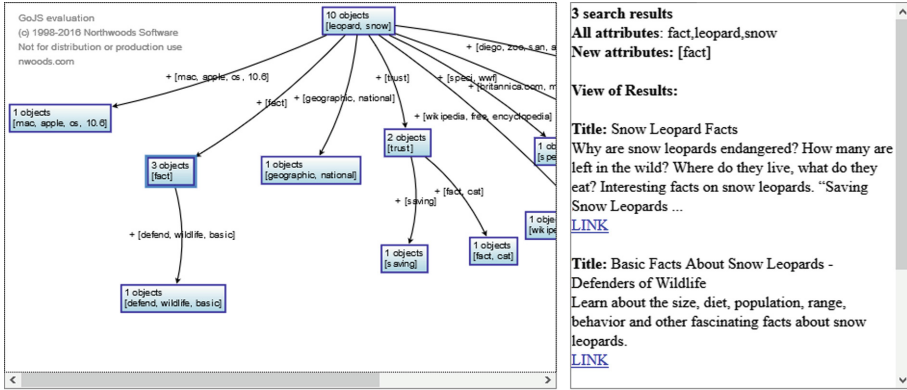


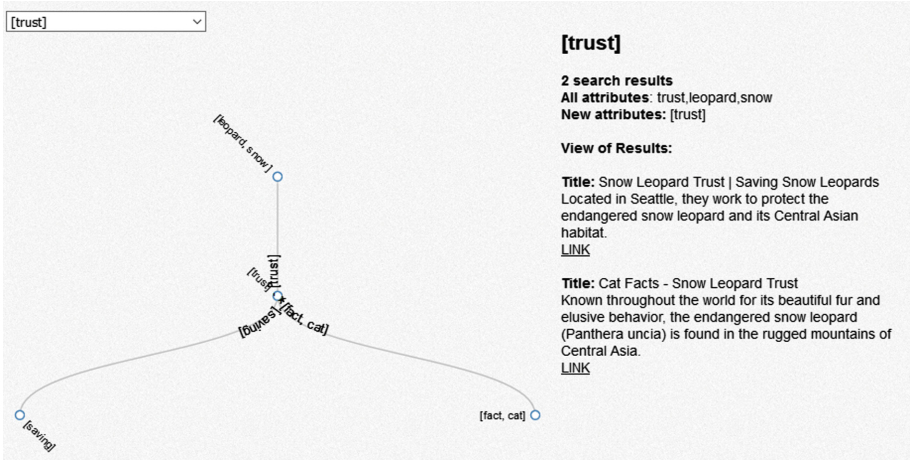
Fig. 2. Example of global visualization in DAG layout - query{snow, leopard}, 10 results, support 10%. Left part - reduced concept lattice, right part - details on selected concept.

10 web search results for specified query, reduced output) has two main parts. On the left side, concept lattice (or its reduction) is shown, where concepts are ordered in DAG layout supported by GO.JS implementation. It means that most general concepts (with all objects) are at the top, then other more specific concepts have an edge to parent concept. Also, the difference to parent concept is visualized on the edge, additional attributes (in this case terms, words in snippets of particular objects in concept) that specify more this concept are shown. The user is able to select one of the concepts and then on the right side, the details of selected concept (highlighted) are shown (at the start, details for top concept are shown). As we can see, the description shows a number of objects, shared attributes of this concept (in the fuzzy model also with their values), as well as information on new attributes according to the parent(s). Then particular results are shown for the current concept, including the link to the original source for every query result.

Another possibility for the user is to use interactive Double tree graph, which provides focused visualization (see Fig. 3) on one concept (in the middle of the left part) and its parents (upper to current node) and children nodes (lower to current node). Again, it is possible to see which words make difference between levels, also, nodes (concepts) are described by their most specific words. On the right side is detailed information on the current concept (same as in the previous type of visualization). The user is able to explore the concept lattice by click on some parent or child. Then clicked node will be focused in the middle (with all necessary changes to parents, nodes and detailed information). At the top of the double tree part, the user is also able to use droplist and quickly select any specific concept from the concept lattice defined by its specific words.

When setup for visualization is different, it will apply necessary changes. Thanks to GOSCL it is possible to use vector (fuzzy) model of snippets (from text of snippet, or original document, as well as add metadata attributes) and





**Fig. 3.** Example of focused visualization in Double tree graph - query {snow, leopard}, 10 results. Left part - focused view on selected concept and its neighbours, right part - details on selected concept.

create concept lattice. Then visualization is updated and together with specified name of attribute also change in value should be shown. In current version we only tested fuzzy model containing term frequency of snippet words, but the application is able to work with generally any type of attributes (and their combinations) which are supported by GOSCL model.

## 5 Experiments

In order to test our tool we did some preliminary experiments with 10 different queries. We would like to see how many concepts are produced, and how effective is the reduction. From this, we were able to estimate more suitable input parameters for visualization tool, i.e., which size of query results set and reduction give us still reasonable output of concept lattice. Then, it was interesting to see if the tool is able to be enough fast in the computation of models. For every experiment, where it was needed, we did more (usually 10) runs of the process and results are based on average numbers.

In the first experiment, we studied the relation between a number of concepts and number of searched results. We did two separate measurements based on the type of input context (binary, or fuzzy - with a minimum of term frequency 1) and no reduction of output. The result is available in Fig. 4, top-left graph, where we can see that difference in our experiments is relatively small. The main reason is the usage of snippets, where term frequencies are not very high and also input table is very sparse (as it is usual in any term-based model of documents). With a longer description of documents and additional metadata attributes, a number of fuzzy concepts will rise faster.

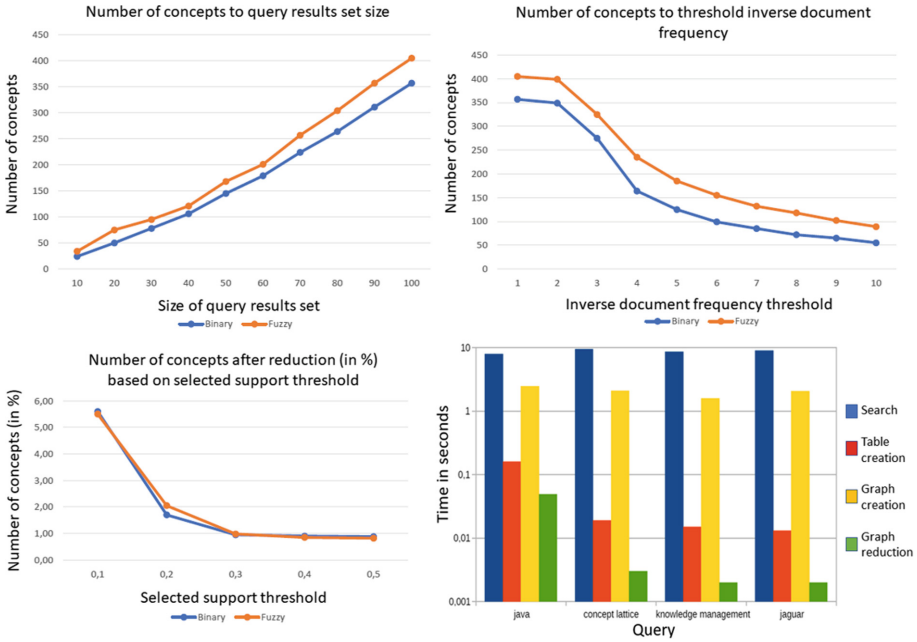


Fig. 4. The results of experiments with our tool (for more details see Sect. 5).

Another experiment tried to show a change in a number of concepts if we setup threshold for a minimum number of term existence in different documents within the query result set, i.e., terms that are in at least  $n$  documents are used for the creation of input context. We have used 100 results for the experiments, no reduction was used after creation of concept lattice. In the top-right graph of Fig. 4 we can see that number of concepts decrease fast until value 6, then decrease is linear and slower. Such reduction is good if we want to have more words which are not so specific (like only for one document in query result set), but still are able to discriminate clusters of web results.

The next experiment was related to seeing how support reduction reduce the output (see Fig. 4, bottom-left graph). As you can see, the value of support greatly reduces the number of concepts. Already at the lowest tested value of support (0.1), the number of concepts is reduced to 5%–6% in average. At 0.3, both for binary and fuzzy contexts, practically only 1% of concepts remains after the reduction phase.

The last graph available in Fig. 4 (bottom-right part) shows the computation aspects of the application within Google API. Here we found that for different queries, the most consuming part of the process is actually the step for the search of results through API, and the creation of graph. The processing times of particular operations for 100 objects in results set (on different queries) is shown in provided graph. For smaller query results sets (under 20–50 objects, depends on the usage of support reduction) with a reasonable amount of visualized concepts,

the processing time of application is usually under half of one second. Therefore, there is no real limitation for usage of the tool if we want to get some useful result.

In conclusion to our preliminary experiments, we also summarized feedback from several users. The readability and usability of resulted output structure are limited to the setup of query result set size to less than 15–20 objects when the reduction is not applied. Otherwise, the structure provided by the system is too large for the user in order to get a useful understanding of clusters and their connections. If we apply reduction techniques (like support function), the readability of output structure depends on its value. By our experiments, we found that even for a maximum of 100 results, we reached a readable and good structure for navigation for support threshold with a value of 0.2. Then understanding and discrimination (or disambiguation) of subgroups of results was relatively good. In the future, other reduction techniques and extensions can help to make even better structuring and visualization of query results set. In the future, we would like to run more experiments with various inputs and setups, as well as provide more reduction techniques and user test studies based on the usage of the system by more users.

## 6 Conclusions

The main goal of this paper was to provide information on a tool for exploration of search results (to selected query) from web engine using Formal Concept Analysis. While some approach in this area were proposed, we decided for interactive visualization of the concept lattice (and its reductions) based on the one-sided fuzzy FCA approach, which is able to involve different types of attributes about the results of the analysis. Our tool is able to support exploration of query results set using such approaches and help in understanding the structure of the domain. In next steps, we want to extend the functionality of the tool, as well as prepare user study and larger experiments with the system.

**Acknowledgments.** This work was done with the help of Short Term Scientific Mission visit supported by the COST action IC1302 KEYSTONE (semantic KEYword-based Search on sTructured data sOurcEs), and partially within the Transregional Collaborative Research Centre SFB/TRR 62 “A Companion-Technology for Cognitive Technical Systems” funded by the German Research Foundation (DFG) and Slovak VEGA research grant 1/0493/16.

## References

1. Ganter, B., Wille, R.: Formal Concept Analysis: Mathematical Foundations. Springer, Heidelberg (1999). <https://doi.org/10.1007/978-3-642-59830-2>
2. Marchionini, G.: Exploratory search: from finding to understanding. Commun. ACM **49**, 41–46 (2006)

3. Gossen, T., Nitsche, M., Haun, S., Nürnberger, A.: Data exploration for bisociative knowledge discovery: a brief overview of tools and evaluation methods. In: Berthold, M.R. (ed.) *Bisociative Knowledge Discovery*. LNCS (LNAI), vol. 7250, pp. 287–300. Springer, Heidelberg (2012). [https://doi.org/10.1007/978-3-642-31830-6\\_20](https://doi.org/10.1007/978-3-642-31830-6_20)
4. Medina, J., Ojeda-Aciego, M., Ruiz-Calviño, J.: Formal concept analysis via multi-adjoint concept lattices. *Fuzzy Set. Syst.* **160**, 130–144 (2009)
5. Pocs, J.: Note on generating fuzzy concept lattices via Galois connections. *Inf. Sci.* **185**(1), 128–136 (2012)
6. Antoni, L., Krajčí, S., Krídlo, O., Macek, B., Pisková, L.: On heterogeneous formal contexts. *Fuzzy Set. Syst.* **234**, 22–33 (2014)
7. Pócs, J., Pócsová, J.: Basic theorem as representation of heterogeneous concept lattices. *Front. Comput. Sci.* **9**(4), 636–642 (2015)
8. Krajčí, S.: Cluster based efficient generation of fuzzy concepts. *Neural Netw. World* **13**(5), 521–530 (2003)
9. Butka, P., Pocs, J.: Generalization of one-sided concept lattices. *Comput. Inform.* **32**(2), 355–370 (2013)
10. Kumar, C.A., Srinivas, S.: Concept lattice reduction using fuzzy K-Means clustering. *Expert Syst. Appl.* **37**(3), 2696–2704 (2010)
11. Antoni, L., Krajčí, S., Krídlo, O.: On stability of fuzzy formal concepts over randomized one-sided formal context. *Fuzzy Set. Syst.* **333**, 36–53 (2018)
12. Kardoš, F., Pócs, J., Pócsová, J.: On concept reduction based on some graph properties. *Knowl. Based Syst.* **93**, 67–74 (2016)
13. Butka, P., Pócs, J., Pócsová, J.: Reduction of concepts from generalized one-sided concept lattice based on subsets quality measure. In: Zgrzywa, A., Choroś, K., Siemiński, A. (eds.) *New Research in Multimedia and Internet Systems*. AISC, vol. 314, pp. 101–111. Springer, Cham (2015). [https://doi.org/10.1007/978-3-319-10383-9\\_10](https://doi.org/10.1007/978-3-319-10383-9_10)
14. Butka, P., Pócs, J., Pócsová, J.: On intent stability index for one-sided concept lattices. In: *Proceedings of 10th Jubilee IEEE International Symposium on Applied Computational Intelligence and Informatics (SACI 2015)*, pp. 79–84 (2015)
15. Butka, P., Pócs, J., Pócsová, J.: Two-step reduction of GOSCL based on subsets quality measure and stability index. In: Zgrzywa, A., Choroś, K., Siemiński, A. (eds.) *Multimedia and Network Information Systems*. AISC, vol. 506, pp. 419–429. Springer, Cham (2017). [https://doi.org/10.1007/978-3-319-43982-2\\_36](https://doi.org/10.1007/978-3-319-43982-2_36)
16. Smatana, M., Butka, P., Čöveková, L.: Tree based reduction of concept lattices based on conceptual indexes. In: Borzemski, L., Grzech, A., Świątek, J., Wilimowska, Z. (eds.) *Information Systems Architecture and Technology: Proceedings of 37th International Conference on Information Systems Architecture and Technology – ISAT 2016 – Part I*. AISC, vol. 521, pp. 211–220. Springer, Cham (2017). [https://doi.org/10.1007/978-3-319-46583-8\\_17](https://doi.org/10.1007/978-3-319-46583-8_17)
17. Haun, S., Nürnberger, A., Kötter, T., Thiel, K., Berthold, M.R.: CET: a tool for creative exploration of graphs. In: Balcázar, J.L., Bonchi, F., Gionis, A., Sebag, M. (eds.) *ECML PKDD 2010*. LNCS (LNAI), vol. 6323, pp. 587–590. Springer, Heidelberg (2010). [https://doi.org/10.1007/978-3-642-15939-8\\_39](https://doi.org/10.1007/978-3-642-15939-8_39)
18. Butka, P., Low, T., Kotzyba, M., Haun, S., Nurnberger, A.: FCA-supported exploratory web search. In: *1st International Symposium on Companion-Technology (ISCT 2015)*, pp. 131–136 (2015)

19. Poelmans, J., Ignatov, D.I., Viaene, S., Dedene, G., Kuznetsov, S.O.: Text mining scientific papers: a survey on FCA-based information retrieval research. In: Perner, P. (ed.) ICDM 2012. LNCS (LNAI), vol. 7377, pp. 273–287. Springer, Heidelberg (2012). [https://doi.org/10.1007/978-3-642-31488-9\\_22](https://doi.org/10.1007/978-3-642-31488-9_22)
20. Carpineto, C., Romano, G.: Exploiting the potential of concept lattices for information retrieval with CREDO. *J. Univ. Comput.* **10**(8), 985–1013 (2004)
21. Nauer, E., Toussaint, Y.: CreChainDo: an iterative and interactive Web information retrieval system based on lattices. *Int. J. Gen. Syst.* **38**(4), 363–378 (2009)
22. Spyratos, N., Meghini, C.: Preference-based query tuning through refinement/enlargement in a formal context. In: Dix, J., Hegner, S.J. (eds.) FoIKS 2006. LNCS, vol. 3861, pp. 278–293. Springer, Heidelberg (2006). [https://doi.org/10.1007/11663881\\_16](https://doi.org/10.1007/11663881_16)
23. Butka, P., Pócsová, J., Pócs, J.: A proposal of the information retrieval system based on the generalized one-sided concept lattices. In: Precup, R.E., Kovács, S., Preitl, S., Petriu, E. (eds.) Applied Computational Intelligence in Engineering and Information Technology. Topics in Intelligent Engineering and Informatics, vol. 1, pp. 59–70. Springer, Heidelberg (2012). [https://doi.org/10.1007/978-3-642-28305-5\\_5](https://doi.org/10.1007/978-3-642-28305-5_5)