

On concept lattices and implication bases from reduced contexts

Vaclav Snasel, Martin Polovincak, Hussam M. Dahwa, and Zdenek Horak

VSB Technical University Ostrava, Czech Republic
{Vaclav.Snasel, Martin.Polovincak.fei, Hussam.Dahwa,
Zdenek.Horak.st4}@vsb.cz

Abstract. Our paper introduces well-known methods for compressing formal context and focuses on concept lattices and attribute implication base changes of compressed formal contexts. In this paper Singular Value Decomposition and Non-negative Matrix Factorisation methods for compressing formal context are discussed. Computing concept lattices from reduced formal contexts results in a smaller number of concepts (with respect to the original lattice). Similarly, we present results of experiments in which we show a way to control smoothly the size of generated Guigues-Duquenne bases and provide some noise resistance for the basis construction process.

1 Introduction

In this paper we are dealing with approaches to obtain concept lattices and attribute implication bases from binary data tables using methods of matrix decomposition. Matrix decomposition methods are well-known in the area of information retrieval under the name Latent Semantic Indexing (LSI) or Latent Semantic Analysis (LSA) ([1]). LSI and LSA have been used for discovery of latent dependencies between terms (or documents). We would like to apply this approach in the area of formal concept analysis (FCA). The goal is to minimise input data before construction of the concept lattices and implication bases, which will result in reduced computational time.

Bases of attribute implications are an interesting form of knowledge extraction, because they are human-readable, convey all information from the data source, and still are as small as possible. Since they are in the basic form very exact, they are also vulnerable to noise in the data and we have almost no control over the resulting number of implications in the bases. Reducing the data to a lower dimension and reconstructing them could help us solve both previous problems. The scalability and computational tractability of FCA are a frequent problem; see, for example, [9] for references. Relevant experiments can be found also in [10].

2 Basic notions

2.1 Formal concept analysis

Formal Concept Analysis (FCA) was first introduced by Rudolf Wille in 1980. FCA is based on the philosophical understanding of the world in terms of objects

and attributes. It is assumed that a relation exists to connect objects to the attributes they possess. Formal context and formal concept are the fundamental notions of FCA [2], [3].

A formal context $C = (G, M, I)$ consists of two sets, G and M , with I in relation to G and M . The elements of G are defined as objects and the elements of M are defined as attributes of the context. In order to express that an object $g \in G$ is related to I with the attribute $m \in M$, we record it as gIm or $(g, m) \in I$ and read that object g has the attribute m . I is also defined as the context incidence relation. For a set $A \subseteq G$ of objects we define $A' = \{m \in M \mid gIm \text{ for all } g \in A\}$ (the set of attributes common to the objects in A). Correspondingly, for a set $B \subseteq M$ of attributes, we define $B' = \{g \in G \mid gIm \text{ for all } m \in B\}$ (the set of objects which have all attributes in B).

A formal concept of the context (G, M, I) is a pair (A, B) with $A \subseteq G$, $B \subseteq M$, $A' = B$ and $B' = A$. We call A the extent and B the intent of the concept (A, B) . $\mathcal{B}(G, M, I)$ denotes the set of all concepts of context (G, M, I) and forms a complete lattice. For more details, see [2].

2.2 Attribute implication

Attribute implication (over set of attributes M) is an expression $A \Rightarrow B$, where $A, B \subseteq M$ (A and B are sets of attributes). The implication can be read as: *if an object has all attributes from A , then it also has all attributes from B and holds in the context (G, M, I) if $A' \subseteq B'$.*

Pseudo-intent of formal context (G, M, I) is a set A of attributes which holds that $A \neq A''$ and $B' \subseteq A$ for each pseudo-intent $B \subset A$. We call a set T of attribute implications **non-redundant**, if no implication from T follows (see [4] for details) from the rest of the set. Set T of attribute implications is **true** in formal context (G, M, I) if all implications from T hold in (G, M, I) . Set T of implications is called **sound and complete** with respect to formal context (G, M, I) if T is true in (G, M, I) and each implication true in (G, M, I) follows from T . As **base** w.r.t. (G, M, I) we call the set of attribute implications, which is sound and complete (w.r.t. (G, M, I)) and non-redundant. The set $T = \{A \Rightarrow A'' \mid A \text{ is a pseudo-intent of } (G, M, I)\}$ is a complete, minimal and non-redundant set of implications and is called the Guigues-Duquenne basis (referred to below as GD).

Bases of implications are interesting to us, since they convey all the information contained in the data table in human-understandable form and they are as small as possible. More on GD bases can be found in [4].

Illustrative example As objects we can consider numbers from 1 to 10 and some basic properties of these numbers form attributes of these objects. Table containing this information can be used as formal context. The computed Guigues-Duquenne basis is presented below.

$$\begin{aligned}
\{\text{composite, odd}\} &\Rightarrow \{\text{composite, odd, square}\} \\
\{\text{even, square}\} &\Rightarrow \{\text{composite, even, square}\} \\
\{\text{even, odd}\} &\Rightarrow \{\text{composite, even, odd, prime, square}\} \\
\{\text{composite, prime}\} &\Rightarrow \{\text{composite, even, odd, prime, square}\} \\
\{\text{odd, square}\} &\Rightarrow \{\text{composite, even, odd, prime, square}\}
\end{aligned}$$

2.3 Singular Value Decomposition

Singular value decomposition (SVD) is well-known because of its application in information retrieval as LSI. SVD is especially suitable in its variant for sparse matrices [5].

Theorem 1: Let A be an $m \times n$ rank- r matrix, $\sigma_1 \geq \dots \geq \sigma_r$ be the eigenvalues of a matrix $\sqrt{AA^T}$. Then there are orthogonal matrices $U = (u_1, \dots, u_r)$ and $V = (v_1, \dots, v_r)$, whose column vectors are orthonormal, and a diagonal matrix $\Sigma = \text{diag}(\sigma_1, \dots, \sigma_r)$. The decomposition $A = U\Sigma V^T$ is called *singular value decomposition* of matrix A and numbers $\sigma_1, \dots, \sigma_r$ are *singular values* of the matrix A . Columns of U (or V) are called *left (or right) singular vectors* of matrix A .

Because the singular values usually fall quickly, we can take only k greatest singular values and corresponding singular vector co-ordinates and create a *k-reduced singular decomposition* of A . Let us have $k, 0 < k < r$ and singular value decomposition of A

$$A = U\Sigma V^T = (U_k U_0) \begin{pmatrix} \Sigma_k & 0 \\ 0 & \Sigma_0 \end{pmatrix} \begin{pmatrix} V_k^T \\ V_0^T \end{pmatrix}$$

We call $A_k = U_k \Sigma_k V_k^T$ a *k-reduced singular value decomposition (rank-k SVD)*.

Theorem 2: (Eckart-Young) among all $m \times n$ matrices C of rank at most k A_k is the one, that minimises $\|A_k - A\|_F^2 = \sum_{i,j} (A_{i,j} - C_{w,j})^2$.

2.4 Non-negative Matrix Decomposition

Non-negative matrix factorisation differs from other rank reduction methods for vector space models in text mining by the use of constraints that produce non-negative basis vectors, which make possible the concept of a parts-based representation. [6] first introduced the notion of parts-based representations for problems in image analysis or text mining that occupy non-negative subspaces in a vector-space model. Basis vectors contain no negative entries. This allows only additive combinations of the vectors to reproduce the original. NMF can be used to organise text collections into partitioned structures or clusters directly derived from the non-negative factors (see [8]).

Common approaches to NMF obtain an approximation of V by computing a (W, H) pair to minimise the Frobenius norm of the difference $V - WH$. Let $V \in R^{m \times n}$ be a non-negative matrix and $W \in R^{m \times k}$ and $H \in R^{k \times n}$ for $0 < k \ll \min(m, n)$. Then, the objective function or minimisation problem can be stated as $\min \|V - WH\|_F^2$ with $W_{ij} > 0$ and $H_{ij} > 0$ for each i and j .

There are several methods for computing NMF. We have used the multiplicative method algorithm proposed by Lee and Seung [6], [7].

3 Experiments

In our experiments we have focused on generating concept lattices and bases from original and reduced context and analysing the differences with respect to the results obtained using original context. Since used reduction methods generate non-binary data, simple rounding was used to obtain boolean matrices.

3.1 Concept lattices

Concept lattice experiments were based on the formal context in Table 1 (see fig. 1 for corresponding lattice).

	A0	A1	A2	A3	A4	A5	A6	A7	A8	A9
O0	0	0	0	0	0	0	0	0	0	1
O1	1	0	0	0	0	0	1	0	0	1
O2	0	0	0	0	0	0	1	0	0	1
O3	0	0	0	0	0	0	1	0	0	0
O4	1	0	0	0	0	0	1	0	0	0
O5	1	0	0	0	0	0	0	0	0	0
O6	0	0	1	1	0	0	1	1	0	0
O7	0	0	0	0	0	0	1	1	0	0
O8	0	0	1	1	0	0	0	0	0	0
O9	0	0	1	0	0	0	1	1	0	0
O10	0	0	0	1	0	0	1	1	0	0
O11	0	0	1	1	0	0	0	1	0	0
O12	0	0	1	1	0	0	1	0	0	0
O13	0	0	1	0	0	0	1	0	0	0
O14	0	0	1	0	0	0	0	1	0	0
O15	0	0	0	1	0	0	1	0	0	0
O16	0	0	0	1	0	0	0	1	0	0
O17	1	0	0	0	0	0	1	1	0	0
O18	0	0	1	1	0	0	0	0	0	1
O19	1	0	1	1	0	0	0	0	0	1

Table 1. Formal context

	A0	A1	A2	A3	A4	A5	A6	A7	A8	A9
O0	0	0	0	0	0	0	0	0	0	0
O1	1	0	0	0	0	0	1	0	0	0
O2	1	0	0	0	0	0	0	0	0	1
O3	1	0	0	0	0	0	0	0	0	1
O4	1	0	0	0	0	0	0	1	0	0
O5	0	0	0	0	0	0	0	1	0	0
O6	0	0	1	1	0	0	1	1	0	0
O7	1	0	0	0	0	0	0	1	0	0
O8	0	0	1	1	0	0	1	0	0	0
O9	0	0	1	1	0	0	1	1	0	0
O10	0	0	1	1	0	0	1	1	0	0
O11	0	0	1	1	0	0	1	0	0	0
O12	0	0	1	1	0	0	1	0	0	1
O13	1	0	1	1	0	0	0	0	0	1
O14	0	0	0	0	0	0	1	0	0	0
O15	1	0	1	1	0	0	1	0	0	1
O16	0	0	0	0	0	0	1	0	0	0
O17	0	0	0	0	0	0	0	1	0	0
O18	0	0	1	1	0	0	1	0	0	0
O19	0	0	1	1	0	0	1	1	0	0

Table 2. Context after SVD reduction

In the following figures we can see that the node 3 from the original concept lattice was deleted, because the attributes composition (A6 and A9) in the objects (O1, O2) is not available, as after using SVD the attribute A6 was deleted from the object O2. The node 20 was deleted, too, because the attributes composition (A0 and A6) in the objects (O1, O4, O17) is not available, as after using SVD, the attribute A6 was deleted from the original objects (O4, O17) and the attribute A0 was deleted from object O17.

We can see also, that after use of SVD, some attributes are removed and added, and more objects have the same compositions of attributes. The node 11 has a composition of attributes (A2 and A6) in the objects (O6, O9, O12, O13); this composition of attributes (A2, A6) existed in the objects (O8, O10, O11, O18, O19), too. The node 6 has composition of attributes (A3 and A6) in the objects (O6, O10, O11, O16); this composition of attributes (A3, A6) existed in the objects (O8, O9, O11, O18, O19) too. From that, the nodes (11 and 6) are incorporated in new node (5), because all the attributes in the two nodes are in all of the objects in the two nodes. That means that the new node 5 has composition of attributes (A2, A3, A6) in the objects (O6, O8, O9, O10, O11,

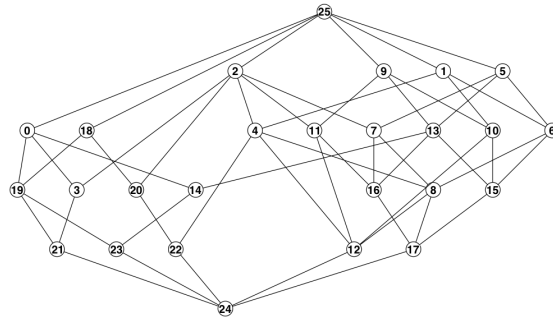


Fig. 1. Concept lattice computed from formal context (Table 1)

Objects	Attrs.
0	0, 1, 2, 18, 19
1	6, 7, 9, 10, 11, 14, 16, 17
2	1-4, 6, 7, 9, 10, 12, 13, 15, 17
3	1, 2
4	6, 7, 9, 10, 17
5	6, 8, 10-12, 15, 16, 18, 19
6	6, 10, 11, 16
7	6, 10, 12, 15
8	6, 10
9	6, 8-14, 18, 19
10	6, 9, 11, 14
11	6, 9, 12, 13
12	6, 9
13	6, 8, 11, 12, 18, 19
14	18, 19
15	6, 11
16	6, 12
17	6
18	1, 4, 5, 17, 19
19	1, 19
20	1, 4, 17
21	1
22	17
23	19
24	
25	0-19

Table 3. Formal concepts

Objects	Attrs.
0	2, 3, 12, 13, 15
1	1, 4-7, 9, 10, 17, 19
2	6, 8-12, 14, 16, 18, 19
3	6, 8-13, 15, 18, 19
4	12, 13, 15
5	6, 8-12, 18, 19
6	12
7	6, 9, 10, 19
8	1, 2, 3, 4, 7, 13, 15
9	2, 3, 13, 15
10	1, 4, 7
11	13, 15
12	
13	0-19

Table 4. Formal concepts after SVD reduction

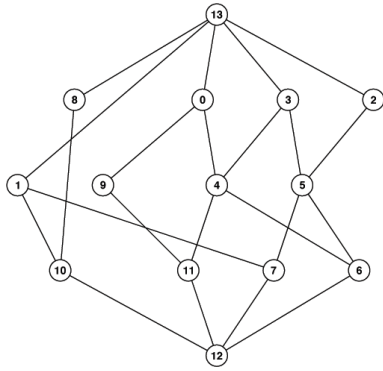


Fig. 2. Concept lattice computed from SVD reduced formal context (Table 2)

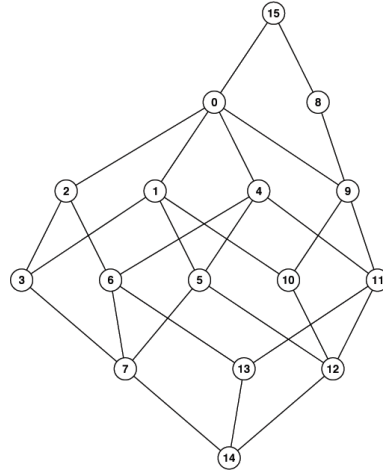


Fig. 3. Concept lattice computed from NMF reduced formal context

O12, O18, O19). Similar results can be obtained using NMF method. Consequent lattice is shown in fig. 3.

3.2 Implication bases

Controlling size of the basis Controlling size of the basis In the first experiment, we generated random contexts (binary data table with sixty objects, fifteen attributes and several densities - 25%, 50%, 75%). Then the matrix was reduced to a lower dimension by use of one of the methods mentioned. The Guigues-Duquenne basis was later computed and results compared against the basis computed from the original data. Size of input data has been selected after computation of several different samples with respect to computational tractability.

The following charts (Fig. 4) illustrate the results of this experiment: the first row corresponds to reduction with the SVD method, the second one to the NMF method. In the figures on the left we present the decreasing number of implications in bases constructed from reduced contexts. Each curve corresponds to one of the aforementioned densities. While lowering the dimension of the data, we are surely losing a certain amount of information. The ratio of objects from the original context, which do not hold in the new basis, is shown in the figures on the right. The results were averaged among hundreds of samples.

Noise resistance Even one small change in source data can cause quite large changes to the GD basis. That can be a huge problem in noisy environments, so we have studied whether reduction into a lower dimension could be helpful. In the following, we suppose that the data contain redundancy and we know the number of rules contained in the data in advance. This situation is not uncommon in applications. Since this is so, we can lower the dimension of the formal context to the number of rules.

More precisely, we have taken several randomly-generated rules (using ten attributes) and combined them into tens of rows to create formal context. Then we put an amount of noise into the data. Later we reduced these datasets to a lower dimension (with the original number of rules used as rank), using SVD and NMF methods. In the last step we compared the GD bases computed from

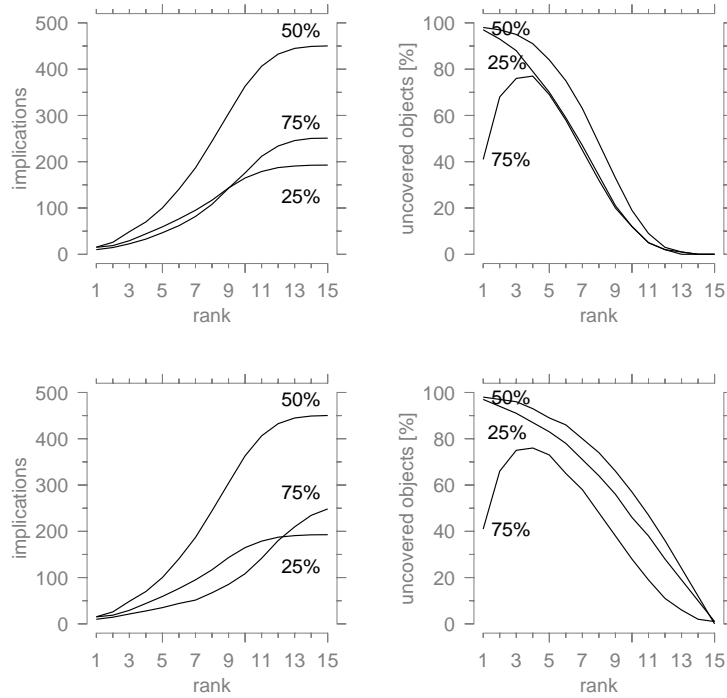


Fig. 4. Average base size and average ratio of uncovered objects for contexts with various densities. Reduced using SVD (first row), NMF (second row).

the original data with the bases from reduced contexts. The results were again averaged among hundreds of samples and fig. 5 comprises an illustrative chart of the results of this experiment.

4 Conclusion and further work

Concept lattices We can see that singular value decomposition used as the first, and non-negative matrix factorisation used as the second, practical approach, were successful and reduced original concept lattices. The number of concepts in reduced concept lattices is lower than in the case of original concept lattices. It implies that computation time of reduced lattices will be lower, and that is why reducing lattices can be useful.

Implication bases We have seen that the size of the resulting implication basis can be smoothly controlled by reduction of the formal context. Our hypothesis is as follows: reduction of formal context to lower dimension with SVD or NMF can lead to faster computation of GD basis, while retaining the most important parts (most objects are still covered by the new basis). Noise resistance in basis construction can also be obtained by use of this method under usual conditions (redundancy, etc.).

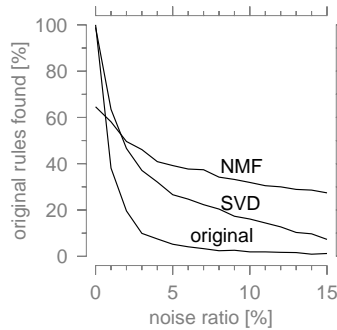


Fig. 5. Ratio of original implications found in bases with added noise (original - without preprocessing, using SVD preprocessing, using NMF preprocessing)

References

1. S. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landauer, R. Harshman: Indexing by latent semantic analysis, *Journal of the American Society for Information Science*, vol 41, 391–407 (1990)
2. B. Ganter, R. Wille: *Formal Concept Analysis: Mathematical Foundations*, Springer-Verlag, New York (1997)
3. R. Wille: Lattices and data analysis: How to draw them using a computer, In I. Rival, Ed., *Algorithms and Order*, Kluwer, Boston, 33–58 (1989)
4. J.L.Guigues, V. Duquenne: Familles minimales d'implications informatives resultant dun tableau de donnees binaires *Math. Sci. Humaines* 95, 5-18 (1986)
5. T. Letsche, M. Berry, S. Dumais: Computation methods for intelligent information access, *Proceedings of the 1995 ACM/IEEE conference on Supercomputing*, ACM Press New York, NY (1995)
6. D. Lee, H. Seung: Learning the parts of objects by non-negative matrix factorization, *Nature*, vol 401, 788–791 (1999)
7. D. Lee, H. Seung: Algorithms for Non-Negative Matrix Factorization, *Advances in Neural Information Processing Systems*, vol 13, 556–562 (2001)
8. F. Shahnaz, M.W. Berry, V.P. Pauca, R.J. Plemmons: Document clustering using nonnegative matrix factorization, *Information Processing and Management*, vol 42, 373–386 (2006)
9. R.J. Cole, P.W. Eklund: Scalability in Formal Concept Analysis, *Computational Intelligence*, vol 15, 11–27 (1999)
10. K. S. Cheung, D. Vogel: Complexity Reduction in Lattice-Based Information Retrieval, *Information Retrieval*, vol 8, 285–299 (2005)