Bit string representation of analysed data

The core of the bit string representation of analysed data is shown at Fig. 1.

Fig. 1: Cards of categories of attribute A_1 of data matrix M

Rows of M	Attributes of M				Cards of categories of attribute A_1			
	A_1	A_2	…	A_K	$A_1[1]$	$A_1[2]$	$A_1[3]$	$A_1[4]$
o_1	1	6	…	B	1	0	0	0
o_2	2	4	…	C	0	1	0	0
o_3	1	7	…	G	1	0	0	0
…	…	…	…	…	…	…	…	…
o_{n-1}	4	9	…	F	0	0	0	1
o_n	3	8	…	H	0	0	1	0

Here we have data matrix M with attributes $A_1, A_2, \ldots, A_K$ and with rows $o_1, \ldots,$ on. We suppose that the {1, 2, 3, 4} are categories (i.e. possible values) of the attribute A_1. Thus the attribute A_1 is represented by cards $A_1[1], A_1[2], A_1[3], A_1[4]$ of categories 1, 2, 3, 4 respectively.

The card $A_1[1]$ of the category *1* is the string of bits. Each row of data matrix M corresponds to one bit of the card $A_1[1]$. There is "1" in the bit corresponding to the row o_i if and only if there is the value *1* in the row o_i. The same is true for other categories and attributes.

The cards of categories and bit string operations are used to compute cards of literals and cards of derived Boolean.

The card $A[\alpha]$ of literal $A(\alpha)$ is a string of bits representing literal $A(\alpha)$. It is e.g.

$$A[1,2,4] = A[1] \vee A[2] \vee A[4]$$

where $A[1] \vee A[2] \vee A[4]$ is a bit-wise conjunction of cards of categories A[1], A[2] and A[4].

The card of Boolean attribute φ is a string of bits representing φ. E.g. the card of derived Boolean attribute $A_1(1,2) \wedge A_2(3,4)$ is a conjunction $A_1[1,2] \wedge A_2[3,4]$ of cards $A_1[1,2]$ and $A_2[3,4]$ of literals $A_1(1,2)$ and $A_2(3,4)$.

The bit-wise operations on cards are carried out by very fast computer instructions. The very fast computer instructions are used also in realisation of the function Count(ξ) that returns the number of "1" in the string of bits ξ. The function Count(ξ) is used in computation of necessary contingency tables.

Some further data structures and sophisticated algorithm based on cards of categories are used in analytical procedures of the system LISp-Miner. They approved to be fast enough for teaching and lot of research purposes, see e.g. něco o 4ft, Florida,

This bit string approach was used already in 1971 (see [Ra 71]) by J. Rauch in the frame of development of GUHA method, see also [Ra 78]. It was adapted for the LISp-Miner and implemented and further developed by M. Šimůnek, see e.g. [RS 01B] and [Si 03].

The current research concerns application of this approach in new analytical procedures of the LISp-Miner system. The other direction of research concerns the application of the bit string approach in multi-relational data mining as suggested in [Ra 86], see also [Ra 02B].