The Third International Conference "Problems of Cybernetics and Informatics", September 6-8, 2010, Baku, Azerbaijan. Section #7 "Numerical Methods and Computational Technology"
www.pci2010.science.az/7/11.pdf

BOOLEAN REASONING APPROACH TO FEATURE SELECTION IN INFORMATION SYSTEMS

Shirzat Kahramanli[1] and Murat Selek[2]

Selcuk University, Konya, Turkey
[1]*sirzat@selcuk.edu.tr*, [2]*mselek@selcuk.edu.tr*

1. INTRODUCTION

An information system (IS) can be represented as $S = \{U, F, D\}$ where, $U=\{U_1,U_2,\ldots,U_M\}$ is a finite set of *M objects (instances)*, $C=\{C_1, C_2,\ldots,C_N\}$ is a finite set of *N condition features*, and *D* is a *decision feature*. There $\forall(j=1,2, \ldots,N)$ $V_j=\{V_{1j}, V_{2j}, \ldots V_{ij}, \ldots, V_{Mj}\}$, and $D=\{d_1,d_2,\ldots d_i, \ldots, d_M\}$, where V_{ij} and d_i are the values that take the condition feature C_j and the decision feature *D* for the object U_i, respectively [1-4]. Such an IS usually is considered as a *data table (data set)* in which the ith row represents a piece of information about the object U_i, and the jth column represents the values of the feature C_j by such a order in which the value $V_{ij} \in V_j$ belongs to the object U_i. Generally speaking, the more columns (features) the data table has the more amount of memory for its storage and the more time for its processing are needed [5-6].

To reduce the computational difficulties related with processing of large data sets the *feature selection* (*attribute reduction*) methods are used. These methods allow to achieve a number of important effects such as: simplification of data set description, reduction of the task of data set collection, minimization of the needed amount of data set storage, speed up a data set mining algorithm, and improvisation of classification accuracy and result comprehensibility [1-4,5-7].

The goal .of feature selection is to find the smallest (optimal) subset *R* of condition features set *C* such that *R* has the same classification power as C with respect to the given decision function [1-3,5,7]. But usually one data set can have several minimal subsets of features (MSAs) and those that of least cardinality are called *reducts* [5,6]. To obtain reducts for a data set with *N* features by the simple sequential search approach, it should be generated and verifyed $\sum_{i=1}^{N}\binom{N}{i} = 2^N\text{-}1$ subsets of the set *F*. That is, feature selection is a problem of exponential space and time complexities in *N* and therefore, it is known as to be *NP-hard* [8,9]. Therefore, the designers of huge data sets are obliging to evaluate a smaller number of subsets [5-9] by some *heuristics* that allow to narrow the search space of the problem as much as possible [2,6,10]. By using some heuristic we can obtain some subsets of features within an acceptable time, but unfortunately they are usually not optimal [5,6,9]. Moreover, since there alternative subsets are not generated, it is impossible to estimate, how much the subset generated as a reduct is near to optimal one.

As it is well known, every object is arranged from the features values associated with this object. This is to say that by processing the objects we can simultaneously process also the features, but indirectly and implicitly. This is achieved by viewing a data table with the binary-valued features as a logic truth table and processing it as one to be minimized logically. But since feature selection differs from the logic minimization in some aspects, we adjust the logic minimization approach according to specific properties of feature selection. As result of this approach, only the set of all possible MSAs is generated. The proposed approach uses only the set union and the logic bitwise operations performed on bit-strings each of which represents one certain object. All steps of this approach are realized by procedures each of which is of linear complexity in N.

2. BOOLEAN MINIMIZATION APPROACH TO THE FEATURE SELECTION PROBLEM

2.1. The Interpretation of a Decision Information System as a Boolean Function

In order to reduce the feature selection problem to a Boolean minimization one, we look at an information decision system with condition features $C_1, C_2, \dots ,C_N$ and decision feature D as a function *D* of variables $C_1, C_2, \dots , C_N$. Namely, $D=f(C_1, C_2, \dots , C_N)$. If there all features are *binary-valued* (Yes-No, Accept-Reject, Open-Close and so on), then the formula $D=f(C_1, C_2, \dots , C_N)$ becomes identical to a Boolean function of *N* variables. Such an IS we will call a *binary-valued* IS. But there are possible also the ISs containing at least one feature taking more than two different values. We will call such a feature as *multiple–valued feature* and an IS containing one or more multiple-valued-feature as a *multiple-valued* IS. In order to look at a multiple-valued feature as a subject of the Boolean functions theory, we encode its values as follows.

1. A set *V* containing all of different values of the given feature are formed.
2. The number of bits of needed codes is obtained as $n=\rceil log_2V\lceil$. But it would be better if to obtain the number of bits for a decision feature as $n=\rceil log_2(V+1)\lceil$ the code $\{0\}^n$ (the bit-string consisting of n zeros) assign to the Don't care objects (the objects that do not present in the IS but would be asked to be classified).
3. A set E_V containing *V* deferent *n*-bit codes is formed. Recall that the set S_C formed for the decision feature should not contain the code $\{0\}^n$.

2.2. The Truth Table-image of an Information System

In order to transform an IS into the corresponding truth table, it is necessary to perform the following operations.

1. Binary-encode the values of the conditional and decision features,
2. Transform of the objects into the minterms,
3. Consider the result of the step 2 as truth table for the decision feature.

Example 1. Consider the binary-valued IS that borrowed from [20] and shown in the Table 1.

Table 1. An example of IS with only binary-valued features

Objects U	**Condition Features**				**Decision Feature**
	Weight C_1	**Door** C_2	**Size** C_3	**Cylinder** C_4	**Mileage** D
U_1	Low	2	Com	4	High
U_2	Low	4	Sub	6	Low
U_3	Low	4	Com	4	High
U_4	High	2	Com	6	Low
U_5	High	4	Com	4	Low
U_6	High	4	Sub	6	Low
U_7	Low	2	Sub	6	Low

After binary-encoding the values of features as: *E:(Low, High)→(0,1); E:(4,2)→(0,1); E: (Com, Sub) →(0,1); E:(4,6)→(0,1)*, where *E* is the function that map the ordered sets of values of the features to the ordered set of bit-strings with the identification space $\{0,1\}^4$ (Table 2).

Table 2. The truth table-image of the Table 1

Minterms		Function
Label	$C_1 C_2 C_3 C_4$	D
T_1	0 0 0 0	1
T_2	0 0 1 1	0
T_3	0 1 0 0	1
T_4	0 1 1 1	0
T_5	1 0 0 0	0
T_6	1 0 1 1	0
T_7	1 1 0 1	0

From this table the following sets of On-minterms and Off-minterms are formed.

$S_{ON}(D) = \{T_1, T_3\}=\{0100, 0000\}$
$S_{OFF}(D) = \{T_2, T_4, T_5, T_6, T_7\}=$
$=\{0011, 1101, 1000, 1011, 0111\}$

3. FEATURE SELECTION BY USING OFF-SET BASED BOOLEAN MINIMIZATION METHODS.

3.1. The Reduced Off-set Based Minimization Concept

According to Off-set reducing concept, if $S_{ON}(f) =\{P_i\}_{i=1,2, \dots , N1}$ and $S_{OFF}(f) =\{Z_j\}_{j=1,2, \dots , N2}$ then the reduction of an Off-cube $Z_j=z_{jn-1}z_{jn-2}\dots z_{j0}$ on a certain On-cube $P_j = p_{n-1}p_{n-2}\dots p_0$ is performed as follows.

$$\text{If } p_i = \bar{z}_{ji} \text{ then } c_{ji} = z_{ji} \text{ else } c_{ji} = *, \quad \forall\, i \in \{0,1,\dots,n-1\} \tag{1}$$

It is obviously that applying the formula to the Off-cube $Z_j=z_{n-1}z_{n-2}\dots z_0$ we get the reduced Off-cube

$$Z_j^r = c_{jn-1}c_{jn-2}\dots c_{j0}.$$

If the value of some feature is * in the given reduced Off-cube then we should consider this feature as irrelevant, otherwise as relevant. In the other words, for relevancy of some feature it is important that this feature has a value differing from *. That is, to transform any Boolean clause into the appropriate feature clause, we should replace all values equivalent to 0(1) by 1(0)s in all of reduced Off-cubes. For example, consider the set $S_{RM}(101)= \{**0,\ \ 11*\}$ obtained for the On-cube $P=001$in the Example 1. It contains the reduced Off-cubes ***0* and *11** . We view these cubes as ***1* and *11** from feature relevancy point of view. Moreover, since there only the values a * and a *1* are used, we may replace the * by the 0. Such a look at the reduced Off-cubes allows to reduce the variable specification space from {0, 1,*} into {0, 1} that causes to reduce the formula (1) into the following one.

$$c_{ji}=z_{ji}\oplus p_i,\ \forall i\in\{0,1,\ldots,n-1\} \qquad (2)$$

In result of processing a Off-cube Z_j by the formula (5) the following cube will be formed.

$$Z_j^{sr} = c_{jn-1}c_{jn-2}\ldots c_{j0}$$

A cube generated by this way we will call as a *SR-cube* (strongly reduced cube). It is obvious that the formula (2) will be realized significantly faster than the formula (1). By applying this formula to all of $Z_j \in S_{OFF}(f)$ and removing non- prime SR-cubes from the result we get the minimal set of SR-cubes as follows.

$$S_{FS}(p)=\{ Z_j^{sr} \}_{j=1}^{Q} \qquad (3)$$

But such a representation of SR-cubes does not allow us to perform any bitwise operation on them. Therefore, we should represent the components of each appropriate SR-cube by unit bit-strings that can be derived by projecting the source SR-cube on the own non-zero bit-positions.

$$E(z_k^{sr})=\{\Pr(z_{ki}^{sr}) \mid d_i =1\} \qquad (4)$$

Where, d_i is the bit of the position i. For example, the SR-cubes z_1^{sr} = 001 and z_2^{sr} =110, obtained above for the On-cube $P=001$ and $n=3$ are to be expanded into the appropriate clauses as follows.

$$E(Z_1^{sr})=\Pr (Z_{13}^{sr})=\Pr (001) = \{001\}$$
$$E(Z_2^{sr})=\Pr (Z_{21}^{sr})\cup\Pr (Z_{22}^{sr}) =\{100, 010\}$$

The subsets of the attributes is generated by simply expanding the $E(Z_1^{sr})$s as follows.

$$R(U) = \bigcup_{q=1}^{Q} E(Z_q^{sr}) \qquad (5)$$

This formula may be computed iteratively as follows.

$$R(U)_{q+1} = R(U)_q \cup E(Z_q^{sr}) \ , \qquad \textit{for all } q= 1,2,\ldots,N \qquad (6)$$

In order to reduce the computational complexity of the formula (8), the dynamical reduction consisting of detection and elimination of the implicants that will produce the redundant terms is used. This is done as follows.

1. $R_1(U)_{q+1} = R(U)_q \setminus \forall a:\ a \cup E(Z_{q+1}^{sr}) = a$
2. $R_2(U)_{q+1} = R(U)_q \setminus R_1(U)_{q+1}$
3. $R(U)_{q+1} = R_1(U)_{q1+1} \vee R_2(U)_{q+1} \cup E(Z_q^{sr})$

This allows us to reduce the computational complexity of the formula (5) at least $\sqrt{S}$ times, where S is the computational complexity of this formula without dynamical reduction of the temporary results.

References

1. R. W. Swiniıairski, "Rough Sets Methods in Feature Reductıon and Classıfıcatıon". Int. J. Appl. Math. Comput. Sci., Vol. 11, No. 3, pp. 565-582, 2001.
2. Z. Pawlak, "Rough Sets. Theoretical Aspects of Reasoning About Data". Boston: Kluwer Academic Publishers,1991.
3. A. Skowron, "The rough sets theory and evidence theory". Fundamenta Informaticae, Vol.13, pp.245-262,1990.
4. J. Komorowski, L. Polkowski, A. Skowron". Rough Set: A Tutorial". http:// folli.loria.fr/cds/ 1999/library/pdf/skowron.pdf.
5. H. Liu and L. Yu, " Toward Integrating Feature Selection Algorithms for Classification and Clustering", IEEE Transactions on Knowledge and data set Engineering". Vol. 17, No.4, pp. 491-502. 2008.
6. W. C. Chen, S.S. Tseng, T.P. Hong, " An efficient bit-based feature selection method". Expert Systems with Applications 34, pp. 2858-2869, 2008.
7. C. Degang, W. Changzhong, H. Qqingha," A New Approach to Attribute Reduction of Consistent and inconsistent covering decision systems with Covering Rough Sets", Information Sciences 177, pp. 3500-3518, 2007.
8. A.L. Blum and P. Langley, "Selection of Relevant Features and Examples in machine learning", Artificial Intelligence, Vol. 97, pp.245-271,1997.
9. P. Mitra, C.A. Murthy, and S.k. Pal, "Universal Feature Selection Using Feature Similarity", IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 24, no. 3, pp. 3001-312, March,2002.
10. A. Blum and P. Langley, "Selection of relevant features and examples in machine learning," Artificial Intelligence, Vol. 97, no. 1-2, pp. 245-271, 1997.
11. T. Ma, M. Tang, " A New Reduction Implementation Based on Concept", Journal of Information and Computation Science, Vol. 1, No. 5, pp. 290-294, 2006.