An Enhancement of Mutual Information Based Algorithm for Reduction of Knowledge

Ahmed I. Sharaf

I. ABSTRACT

II. INTRODUCTION

Feature selection is one of the most significant research topic in pattern recognition, machine learning, decision-making systems, data-mining and many other applications. In such applications, dataset built with varied significance of each attribute. Therefore it is possible to find a redundant or irrelevant attributes in dataset, which not only occupy extensive computing resources, but also seriously influence the decision making. Therefore, eliminating the irrelevant attributes are necessary to increase the efficiency of decision making and reduce the computation time. Wide-ranging researchers proposed several methods for feature selection (Wang, Yang et al. 2007, Liu, Sun et al. 2009, Bae, Yeh et al. 2010, Sun, Xu et al. 2012). An effective and powerful mathematical tool used is Rough Set theory. Rough Set theory proposed by Pawlak (Pawlak 1982) is used to analyze imprecise, uncertainty and vague information. Rough Set theoretical model, introduced the concept of equivalence class which replaces the original class of attributes by a minified set of non-repeated attributes called reduct. Finding all possible reducts is a classical NP-hard problem. Therefore, suboptimal reduction algorithms are used with heuristic approaches to find semi-optimal reduct. Various methods are developed to compute the reduct using evolutionary algorithms (Ke, Feng et al. 2008, Yu, Wang et al. 2008), discernibility matrix (Kryszkiewicz 1998, Yao and Zhao 2009) and heuristic reduction (Dash and Liu 2003, Hu, Zhao et al. 2007, Błaszczyński, Greco et al. 2009). Furthermore, information entropy is used within heuristic reduction algorithms as an effective measure of uncertainty (Cuiru and Fangfang 2008, Cuiru and Fangfang 2008, Qian, Liang et al. 2010). The acquired number of bits is expected by means of information entropy measure, also any change of data is represented by the change of entropy. This paper presents a novel algorithm of feature selection using rough set theory. From the point of view of information theory, an enhanced measure of entropy is used.

III. PRELIMINARY KNOWLEDGE

In this section, some basics definitions and essential concepts of Rough Set theory based on information entropy will be reviewed.

Definition 1. *A Decision table in Rough Set theory is represented as a four-tuple $T = (U, A, V, f)$ where U denotes a non-empty finite set of objects which is called the universe, A denotes the set of attributes that contains the condition set C and decision set D, V denotes the non-empty set of attribute values, and f is a Cartesian product of into Vsuch that $V \leftarrow U \times A$.*

Definition 2. *For any given decision table $T = (U, C \cup D, V, f)$, an indiscernibility relation regarding to a non-empty subset B such that $B \subseteq C$ can be expressed as: $IND(B) = (x, y)|f(x, a) = f(a, y), \forall a \in B$ The relation $IND(B)$ splits the universe U into a finite number of classes which are called equivalence classes, thus constructing a partition denoted by $U/IND(B)$. The subset B equivalence class of $x \in U$ is formed as $[x]_B = y \in U|(x, y) \in IND(B)$. Similarly, a partition of $U/(IND(D) = Y_1, Y_2, \ldots, Y_N$ regarding to the decision attributes D.*

Definition 3. *For any given decision table $T = (U, C \cup D, V, f)$, $B \subseteq C$, the lower approximation $\underline{B}(D_i)$ and upper approximation $\bar{B}(D_i)$ regarding to B are expressed as follows:*
$\underline{B}(D_i) = \{x \in U|[x]_B \subseteq D_i\}$ *where* $1 \leq i \leq n$

$\bar{B}(D_i) = \{x \in U|[x]_B \cap D_i \neq \phi\}$

Definition 4. *Let U be a universe, and subsets P, Q are a family of equivalence relations on the universe. Then P, Q are considered to be two random variables that composed of the subsets of the universe U. Let X, Y are two sets of the universe induced by P, Q respectively such that:*
$X = U/(IND(P) = X_1, X_2, \ldots X_n)$
$Y = U/(IND(Q) = Y_1, Y_2, \ldots Y_m)$
Then the probability distribution of X, Y are expressed respectively as follows:

$$[X : p] = \begin{bmatrix} X_1 & X_2 & ... & X_m \\ p(X_1) & p(X_2) & ... & p(X_m) \end{bmatrix}$$

$$[Y : q] = \begin{bmatrix} Y_1 & Y_2 & ... & Y_m \\ p(Y_1) & p(Y_2) & ... & p(Y_m) \end{bmatrix}$$

Where $p(X_i) = Card(X_i)/(Card(U))$, $i = 1, 2, 3, \ldots, n$, $p(Y_1) = card(Y_j)/(card(U))$,$j = 1, 2, 3, \ldots, m$ *and* $card(.)$ *denotes the cardinality of the set.*

Definition 5. *For any given decision table* $T = (U, C \cup D, V, f)$, $B \subseteq C$. *Then, one can obtain conditional partitions* $U/B = X_1, X_2, \ldots X_m$ *and* $U/D = Y_1, Y_2, \ldots, Y_n$. *Based on these partitions, a conditional entropy of set* B *relative to set* D *is expressed as follows:*
6895ad $H(D|B) = \sum_{i=1}^{m} \frac{|X_i|}{|U|} \sum_{j=1}^{n} \frac{|X_i \cap Y_j|}{|X_i|} . \log_2 \frac{|X_i|}{|X_i \cap Y_j|}$

Definition 6. *Mutual information can be defined using entropy and conditional entropy as follows:* $I(P, Q) = H(Q) - H(Q|P)$

Definition 7. *For any given decision table* $T = (U, C \cup D, V, f), B \subseteq C, \forall a \in B$ *,the inner significance measure of attribute* a *is defined as:*
$Sig^{inner}(a, B, D) = H(D|B - a) - H(D|B)$

Definition 8. *For any given decision table* $T = (U, C \cup D, V, f), B \subseteq C, \forall a \in C - B$ *, the outer significance measure of attribute* a *is defined as:*
$Sig^{outer}(a, B, D) = H(D|B) - H(D|B \cup a)$

IV. Attribute reduction Algorithm based on information entropy

Rough Set attribute reduction has been employed to remove redundant attributes from discrete values datasets, while retaining their information content. An effective algorithm using mutual information entropy is MIBARK which was presented by (Miao and Hu 1999). Many researches have been proposed based on the MIBARK algorithm similar to (Xu, Miao et al. 2009), (Sun, Xu et al. 2012), (Dai, Wang et al. 2013), (Zheng, Hu et al. 2014) that prove its efficiency. The MIBARK attempts to determine minimum reduct without exhaustively generating all possible subsets. The MIBARK starts with a relative core and perform incremental attributes addition according to their corresponding significance values. Although the MIBARK is sufficient, it has been proved that this method does not always generate a minimal reduct (Feifei, Duoqian et al. 2007). Another problem arises when more than one attribute has the same value of significance. The significance of different attributes are given as identically largest values in some rounds especially in small sized datasets (Zheng, Hu et al. 2014). The pseudo-code of the MIBARK is shown as follows:

Algorithm 1 Mutual Information Based Algorithm for Reduction of Knowledge (MIBARK)

1: **Input**: A decision table $T = (U, C \cup D)$
2: **Ouput**: Reduct B
3: Compute the mutual information entropy $I(C, D)$ for decision table T.
4: Compute the relative core $C_0 = CORE_D(C)$ when $C_0 = \phi$ then $I(C_0, D) = 0$.
5: Let $B = C_0$
6: **repeat**
7: **for all** attribute $a \in C - B$ **do**
8: Compute the significance of a
9: **end for**
10: Select the attributes which brings the maximum (a, B, D) , if more than one attribute achieving the maximum significance, choose one whose combination with B reaches least as a. then $B \leftarrow B \cup \{a\}$.
11: **until** $I(B, D) = I(C, D)$

V. Enhancement of mutual information entropy attribute reduction

In this section, an enhancement of the mutual information entropy algorithm is presented. The proposed algorithm aims to solve some of the MIBARK limitations discussed in the previous section. The proposed algorithm starts with an empty set R that represents the relative core, it seeks to establish the reduct from bottom to up approach. Consequently, the significance of attributes are computed, if one attribute has the largest value of significance its added to the reduct set R. Otherwise, when more than one attributes have the largest values of significance, then a combination of the attributes are built. Afterwards, one attribute which has the largest value of significance is combined with anther unselected attribute to establish a combination. The suggested subroutine will continue processing until all the combinations are built. Although, this approach could produce duplicated attribute combinations therefore it is followed by removing duplicated attributes. Afterwards, the significance of each combination is computed due to select the largest value of significance combination. When two or more combination have the same largest significance, then return to the regeneration step to produce different combination of attributes. Finally, a different step of reduction is performed by eliminating the unnecessary attributes according to outer significance. The flow chart and pseudo-code of the proposed algorithm are shown below to demonstrate the workflow of the algorithm.

Algorithm 2 Enhancement algorithm of mutual information entropy attribute reduction

1: **Input**: A decision table $T = (U, C \cup D)$
2: **Ouput**: Reduct red
3: Compute the mutual information $I(C, D)$
4: **for all** $a \in C - B$ **do**
5: compute the significance of a.
6: **end for**
7: Sort the attributes according to their corresponding significance values.
8: **if** one attribute satisfy the maximum significance **then**
9: $B \leftarrow B \cup \{a_0\}$ where a_0 denotes the maximum significance attribute.
10: **else**
11: REDUCE(a)
12: **end if**
13: **function** REDUCE($S = (U, C \cup D, V, f)$)
14: Let $A'' = \{A'_1, A'_2, ..., A'_m\}$ where A'' denotes the set of attribute combinations
15: Let $R' \leftarrow R' \cup \{A'_i\}$ with random selection of i where $1 \leq i \leq m$.
16: **if** R' has one attribute **then**
17: **return** $R = \{\{R \cup A'_1\}, \{R \cup A'_2\}, ...\{R \cup A'_m\}\}$
18: **else**
19: continue;
20: **end if**
21: **for all** attribute A'_i and $b \in C - R - A'_i$ **do**
22: $A \leftarrow \{A'_i \cup b\}$
23: compute the $Sig^{inner}(A, R, D)$
24: $A'' \leftarrow$ the attribute combination that maximize the $Sig^{inner}(A, R, D)$
25: **end for**
26: **if** A'' has one attribute **then**
27: return A'';
28: **else**
29: goto line 14;
30: **end if**
31: **return** $R = \{\{R \cup A'_1\}, \{R \cup A'_2\}, ...\{R, A'_m\}\}$
32: **end function**