Comparison of the classification rules generated by See 5.0 and SSCO Systems

Višnja Ognjenović, Vladimir Brtka, Ivana Berković, Eleonora Brtka
University of Novi Sad
Technical faculty "Mihajlo Pupin"
Đure Đakovića bb, 23000 Zrenjanin, Serbia
{visnjao, vbrtka, berkovic}@tfzr.uns.ac.rs

Abstract. *This paper presents the comparison of the classification rules generated by See5.0/C5.0 and SSCO systems. See5.0/C5.0 system is based on C4.5 algorithm, while SSCO system is based on an algorithm, theoretically correlated to Rough Set Theory. Both systems generate classification rules in the IF THEN form.*

The goal of comparison of the classification rules, generated by those two systems is detection and extraction of important rules in the terms of classification power. Some experimental comparison of two systems has been done using the Wisconsin Breast Cancer Database (January 8, 1991), obtained from UCI Machine Learning Repository.

Keywords. classification rules, rule merging, classification power, See5.0/C5.0, SSCO

1 Introduction

Classification is useful in many decision problems, where for a given dataset a decision is to be made. Classification of data can be based on a training set, which is a part of database. The result of this training can be expressed in the form of classification rules. These rules can be written in the IF THEN form and represent the model for classifying new data.

The classification rules induced by machine learning systems are estimated by two criteria: their classification accuracy on an independent test set, and their complexity. "There are in the literature some indications that very simple rules may achieve surprisingly high accuracy on many datasets" [9]. For example, Rendell and Seshu [15] remark that many real world data sets have 'few peaks (often just one)' and are therefore 'easy to learn'.

This paper will compare classification rules (in the IF THEN form) obtained by the systems based on different algorithms. The first one is See5.0/C5.0 system, which is one of the most popular inductive learning tools, originally proposed by J. R. Quinlan as C4.5 algorithm [14]. The second one is SSCO system, based on algorithm theoretically correlated to Rough Set Theory (RST) [2]. Their comparison has been conducted by experiments with data sets. The goal of comparison of the classification rules generated by those two systems is detection and extraction of important rules, as well as detection of the rules that can be merged together.

Given that this is the initial of research, a small database with good classification features was used.

The paper is organized as follows: Preliminaries are in section two, section three describes See5.0/C5.0 systems. Section 4 is the presentation of the SSCO system. Experimental results are in Section 5 while comparison and conclusions are in Section 6.

2 Preliminaries

Let be given set of n independent variables $\{x_1,..,x_n\}$ such that each x_1 takes on values from a domain C_{Xk} Domain of decision variable x_d called the class variable, is $D=\{d_1,...d_m\}$, with m being the number of classes. The task of a classification is to:

(i) Determine a training dataset consisting of a set of (n+1)-tuples: $(t_1,..,t_n,d)$, where $t_k \in C_{Xk},(k=1,...,n)$ and $d \in D$;

(ii) Construct a mapping $f:(C_{X1},..,C_{Xn}) \to D$.

Mapping f can be used to predict the class of new tuples [7].

Mapping f can be written in the IF THEN form:

IF ($x_1 = t_1$ AND … AND $x_n = t_n$) THEN $x_d = t_d$

Here, rule's antecedent denotes a conjunction (AND logical operator).

Classificatory power of the rules can be measured by the number of tuples from dataset. Some of them are generated by only one tuple of dataset while some

of them are generated by more than hundred of tuples. It is known that decision trees generate small number of rules compared to other systems. A reason for this is following: the values of rules variables belong to the interval, so they can satisfy a large number of tuples.

3 Overview of See5.0 system

See5.0 system is based on algorithm C5.0. Algorithm C5.0 is an extension of ID3 attribute-based machine learning system. The ID3 algorithm (Iterative Dichotomiser 3) is an algorithm used to generate a decision tree invented by Ross Quinlan [14].

3.1 ID3 algorithm

ID3 algorithm builds decision trees from a set of training data, using the concept of information entropy. For n independent variables $\{x_1,..,x_n\}$ such that each x_k takes on values from a dataset S, algorithm determines the one that has the biggest information gain.

Definition 1. Entropy. Entropy is a measure of an uncertainty. If the probability of an instance's belonging to the class 'i' is marked with p_i, then the entropy is:

$$Entropy(S) = -\sum_{i=1}^{c} p_i \log_2(p_i) \qquad (1)$$

where 'c' is the number of classes. It can be interpreted as a minimal expected number of bits necessary for coding the classification of arbitrary instance from S.

Definition 2. Gain. In general terms, the information gain is the remainder in information entropy.

$$Gain(S, x_1) = Entropy(S) - Entropy(S \mid x_1) \qquad (2)$$

where $Entropy(S \mid x_1)$ denote conditional entropy. Formally, information gain is calculated by the formula:

$$Gain(S, x_1) = Entropy(S) - \sum_{v \in V(x_1)} \frac{|S_v|}{|S|} Entropy(S_v) \qquad (2')$$

where $V(x_1)$ is the set of all possible values of attribute x_1, and $S_v = \{s \in S | x_1(s) = v\}$.

An example: A simple dataset is given in Table 1. Here, x_1, x_2, x_3, x_4 are called condition attributes while x_d is called decision attribute. Each line in Table 1 is called a tuple.

Table 1. Dataset

No.	x_1	x_2	x_3	x_4	x_d
1	4	2	8	0	yes
2	4	2	3	2	yes
3	4	2	8	3	yes
4	7	2	8	0	yes
5	7	5	8	2	no
6	4	5	8	0	yes
7	4	2	3	0	no
8	7	5	1	2	yes
9	4	2	8	2	yes
10	4	1	3	2	yes
11	7	1	8	2	no
12	4	1	8	3	yes

$$Entropy(S) = -\frac{\text{the number of tuples with class label yes}}{\text{the number of all tuples}} * \log_2\left(\frac{\text{the number of tuples with class label yes}}{\text{the number of all tuples}}\right) - \frac{\text{the number of tuples with class label no}}{\text{the number of all tuples}} * \log_2\left(\frac{\text{the number of tuples with class label no}}{\text{the number of all tuples}}\right) = -\frac{9}{12} * \log_2\left(\frac{9}{12}\right) - \frac{3}{12}\log_2\frac{3}{12} = 0{,}811 bits$$

$$Entropy(S \mid x_1) = \frac{8}{12}\left(-\frac{7}{8} * \log_2\frac{7}{8} - \frac{1}{8} * \log_2\frac{1}{8}\right) + \frac{4}{12}\left(-\frac{2}{4} * \log_2\frac{2}{4} - \frac{2}{4} * \log_2\frac{2}{4}\right) = 0{,}696 bits$$

$$Gain(S, x_1) = 0{,}811 bits - 0{,}696 bits = 0{,}115 bits$$

Similarly:

$Gain(S, x_2) = 0{,}027 bits$

$Gain(S, x_3) = 0{,}04 bits$

$Gain(S, x_4) = 0{,}082 bits$

Algorithm takes attribute x_1 as the first node in decision tree because its gain is the biggest. According to [14], decision tree is formed in the following way:

- Take all unused attributes and calculate their entropy. Each node in the decision tree corresponds to a non-categorical attribute and each arc corresponds to a possible value of that attribute. A leaf of the tree specifies the expected value of the categorical attribute for the records described by the path from the root to that leaf.
- Choose attribute for which entropy is minimal. The non-categorical attribute, as the most informative among the attributes still unconsidered in their path from the root, should be associated at each node in the decision tree.

– Form the node containing that attribute. Measure how informative the node is using entropy.

3.2 Data Mining Tools See5\C5.0

See5 (Windows 2000/XP/Vista/Windows 7) and C5.0 (Unix) are sophisticated data mining tools for discovering patterns that delineate categories, assembling them into classifiers, and using them to make predictions [6].

Some important features are:

– See5/C5.0 has been designed to analyze substantial databases containing thousands to hundreds of thousands of records and tens to hundreds of numeric, time, date, or nominal fields. See5/C5.0 also takes advantage of computers with up to eight cores in one or more CPUs (including Intel Hyper-Threading) to speed up the analysis.

– In order to maximize interpretability, See5/C5.0 classifiers are expressed as forms that are generally easier to understand than neural networks. These forms are decision trees or sets of if-then rules.

4 Overview of SSCO system

Syntactic Systematic Classification of Objects (SSCO) is the rule generating technique that was developed in period 2006 to 2008 at Technical Faculty "Mihajlo Pupin" in Zrenjanin, Serbia. It is theoretically connected with RST.

In 1982, Pawlak introduced theory of rough sets [12]. He derived rough dependency of attributes in information systems. Basic concepts of RS are:

Let U be a universe (finite set of objects), $Q=\{q_1,q_2,...q_m\}$ is a finite set of attributes, V_q is the domain of attribute q and $V=\bigcup_{q\in Q}V_q$ [13].

Definition 3. Information System. An information system is the quadruple $S=\langle U,Q,V,f\rangle$ where $f=U\times Q\to V$ is a total function such that $f(x,q)\in V_q$ for each $q\in Q,x\in U$, called information function.

If some attributes are interpreted as outcome of classification, the information system $S=\langle U,Q,V,f\rangle$ can be defined as a decision system by $DS=\langle U,C,D,V,f\rangle$, where $C\cup D=Q$, $C\cap D=\varnothing$. C is called the set of condition attributes and set D is called the set of decision attributes [13]. Usually, there is one binary decision attribute.

Definition 4. Indiscernibility Relation. To every non–empty subset of attributes P is associated an indiscernibility relation on U, denoted by I_P:

$$I_P=\{(x,y)\in U\times U: f(x,q)=f(y,q),\forall q\in P\} \quad (3)$$

The relation (3) is an equivalence relation – reflexive, symmetric and transitive. The family of all the equivalence classes of the I_P is denoted by $U|I_P$ and class containing an element x by $I_P(x)$ [10].

According to indiscernibility relation, the main task in RST is to find the smallest subset of features without losing any information. These minimal subsets of features are called reducts. The reducts in RST are sets that contain the same quality of data information as the original set.

It is possible to generate classifying rules from the reducts. The rules are logical statements of the type "IF conjunction of condition features THEN disjunction of decision features" which are induced from the reduced set [8].

4.1 SSCO algorithm

Similarly as RST, SSCO system uses indiscernibility relation to classify data. In [11] graph representation of SSCO algorithm is examined. This algorithm enables the partitioning of the universe of objects represented by their attributes. As proposed in [2, 3, 5] the automated rule extraction technique without previous reduct computation based on the classification of the objects is possible to implement by state–space search algorithm in the manner of depth first search. The root of the state–space graph is set $X\subseteq U$; nodes are the sets of objects, while attribute-value pairs define arcs. An iterating algorithm exploits functional dependences between condition and decision attributes. Under the assumption that decision attribute is the last one, every path of state–space graph which ends with non–empty leaf will produce one classification rule.

An example: For dataset from Table 1, rule-generating process is represented by graph on Figure 1.

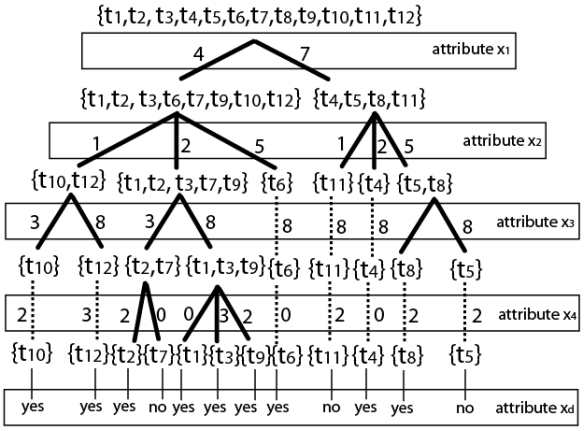

Figure 1. Graph representation rules synthesis

4.2 SSCO software system

One particular advantage offered by the SSCO system is synthesis of smaller number of IF THEN rules compared to number of rules generated by RST software such as Rosetta. This was proved significantly important in analysis of medical data [2, 4, 5].

Every rule generated by SSCO system is accompanied by rule support and rule probability. For example:

```
[4,0.75] IF (a1,1), (a2,6), (a5,4), (a7,1) THEN (a8,1)
```

Here, 4 (from [4, 0.75]) refers to a number of supporting objects while probability is 0.75. The IF part contains the condition attributes (a1, a2, a5, a7) and their values (1, 6, 4, 1). The THEN part contains decision attribute (a8) with its value (1).

5 Experimental results based on Wisconsin Breast Cancer Database

The Wisconsin Breast Cancer Database (January 8, 1991) had been taken from the UCI Machine Repository [1]. This database was obtained from the University of Wisconsin Hospitals, Madison from Dr. William H. Wolberg:

Attributes 2 through 10, see Table2, have been used to represent instances (objects or cases). Each instance has one of two possible classes: benign or malignant.

Class attribute (decision attribute) has been moved to last column. Attribute Sample code number is irrelevant and therefore was excluded from further experiments.

Table 2. Attribute description

No.	Attribute	Domain
1.	Sample code number	id number
2.	Clump Thickness	1 - 10
3.	Uniformity of Cell Size	1 - 10
4.	Uniformity of Cell Shape	1 - 10
5.	Marginal Adhesion	1 - 10
6.	Single Epithelial Cell Size	1 - 10
7.	Bare Nuclei	1 - 10
8.	Bland Chromatin	1 - 10
9.	Normal Nucleoli	1 - 10
10.	Mitoses	1 - 10
11.	Class	(2 for benign, 4 for malignant)

5.1 Methodology

To evaluate obtained rules the confusion matrix was used. A confusion matrix is a table layout that allows evaluation of the performance of a classifier. It contains information about actual and predicted classifications done by a classification system: $|V_d| \times |V_d|$ matrix, where V_d is the set of possible values of decision attribute. This matrix with integer entries summarizes the performance of rule set while classifying the set of objects. Entry:

$$C_{i,j} = \left|\{x \in U : d(x) = i, \bar{d}(x) = j\}\right|,$$

where $d(x)$ is the actual decision and $\bar{d}(x)$ is the predicted decision, which counts the number of objects that really belong to class *i*, but were classified to class *j*.

Rules were also compared by number of support tuples and length of the rules.

5.2 Experiment

5.2.1. Experimental results from See5.0

The 400 objects were taken for training, while 199 objects were used for test. The following results were obtained:

```
Rules:

Rule 1: (224, lift 1.6)
        Uniformity of Cell Size <= 3
        Bare Nuclei <= 2
        ->  class 2  [0.996]

Rule 2: (238/1, lift 1.5)
        Clump Thickness <= 6
        Uniformity of Cell Size <= 3
        Bland Chromatin <= 3
        ->  class 2  [0.992]

Rule 3: (207/1, lift 1.5)
        Clump Thickness <= 6
        Marginal Adhesion <= 1
        Bland Chromatin <= 4
        Mitoses <= 2
        ->  class 2  [0.990]

Rule 4: (172/1, lift 1.5)
        Clump Thickness <= 6
        Bland Chromatin <= 2
        ->  class 2  [0.989]

Rule 5: (112/2, lift 2.7)
        Bare Nuclei > 2
        Bland Chromatin > 3
        ->  class 4  [0.974]

Rule 6: (115/4, lift 2.7)
        Uniformity of Cell Size > 3
        Marginal Adhesion > 1
        Bland Chromatin > 2
        ->  class 4  [0.957]

Rule 7: (91/4, lift 2.6)
        Clump Thickness > 6
        ->  class 4  [0.946]

Rule 8: (130/10, lift 2.5)
        Bland Chromatin > 3
        ->  class 4  [0.917]
```

```
Default class: 2


Evaluation on training data (400 cases):

          Rules
        ----------------
          No      Errors

           8    5( 1.3%)   <<


         (a)   (b)    <-classified as
        ----  ----
        252     4    (a): class 2
          1   143    (b): class 4
```

By evaluation on test set (199 objects), we have:

```
Evaluation on test data (199 cases):

          Rules
        ----------------
          No      Errors

           8   10( 5.0%)   <<


         (a)   (b)    <-classified as
        ----  ----
        125     5    (a): class 2
          5    64    (b): class 4
```

Here, five objects that really belong to class a are classified to class b and five objects that really belong to class b were classified to class a.

Eight rules were generated in total. The classification test set had the success rate of 95%.

5.2.2 Experimental results obtained by SSCO

Again, the 400 objects were taken for training, while 199 objects were used for test, 65 rules were synthesized; some of them are listed below:

```
IF THEN Form
[28,1] IF  (a2,10) THEN (a11,2)
[4,1]  IF  (a2,9) THEN (a11,2)
[7,1]  IF  (a2,8), (a3,10) THEN (a11,2)
[1,1]  IF  (a2,8), (a3,8) THEN (a11,2)
[4,1]  IF  (a2,8), (a3,7) THEN (a11,2)
...
[22,1] IF  (a2,2), (a3,1) THEN (a11,1)
[81,1] IF  (a2,1) THEN (a11,1)

Evaluation on test data (199 objects):

Confusion matrix:
        1       2

1       102     4

2       11      72
```

The test set had the success rate of 94.7%.

Six rules with significant support were extracted:

```
1. IF  (a2,10) THEN (a11,2)
2. IF  (a2,5), (a3,1) THEN (a11,1)
3. IF  (a2,4), (a3,1) THEN (a11,1)
4. IF  (a2,3), (a3,1) THEN (a11,1)
5. IF  (a2,2), (a3,1) THEN (a11,1)
6. IF  (a2,1) THEN (a11,1)
```

In a post-process, first rule can be rewritten in the form:

```
IF  Clump Thickness =10 THEN Class=4
```

By merging rules 2 to 6, the following rule is obtained:

```
IF  Clump Thickness <=5 AND Uniformity
of Cell Size=1 THEN Class=2
```

Now, we have just two rules instead of six.

Furthermore, classification power of six rules was tested on the same 199 objects. Following confusion matrix was obtained:

```
Evaluation on test data (199 objects):

Confusion matrix:
        1       2

1       95      0

2       4       30
```

The test set had the success rate of 64.8 %.

The classification power of two derived rules is same as the classification power of six rules generated by SSCO.

Further comparison was made between two rules that were derived from six rules with significant support factor, generated by SSCO, and eight rules generated by See5.0.

The first rule:

```
IF  Clump Thickness =10 THEN Class=4
```

can be compared with rule number 7 of See5.0 system:

```
Clump Thickness > 6
        ->  class 4
```

We can see that See5.0 rule is more general.

The second rule:

```
IF   Clump Thickness <=5 AND Uniformity
of Cell Size=1 THEN Class=2
```

can be compared with rule number 2, of See5.0 system:

```
Clump Thickness <= 6
        Uniformity of Cell Size <= 3
        Bland Chromatin <= 3
        ->  class 2
```

In this case, rule obtained by merging multiple rules is shorter (have less condition attributes in the IF part) and consequently more general than rule generated by See5.0.

6 Conclusion

Classification rules in the IF THEN form were generated by well-known See5.0 system which is based on decision trees and state-space search based SSCO system. See5.0 generated eight rules compared to 65 rules generated by SSCO system. The See5.0 system is in small advantage in terms of classification power (95% to 94.7%).

From 65 generated rules of SSCO system, six rules are statistically important (have the important number of supporting objects). The classification power of six rules is significantly lower (64.8%).

However, it is shown how to extract most important rules as well as how to implement the merging procedure so that multiple rules can be merged together. In this case, six rules were merged together so that they form two rules, maintaining the same classification power. Furthermore, rules generated by different systems were compared with the aim of extracting most important rules.

This analysis presents a stimulus for post processing of a set of classification rules. It is shown that extraction of important rules in terms of classification power is possible. Further experiments will be done on a bigger database.

References

[1] Blake, C.L., Merz, C.J.: UCI Machine Learning Repository, http://archive.ics.uci.edu/ml/

[2] Brtka V. Berkovic I., Stokic E., Srdic B., A comparison of rule sets generated from Databases by indiscernibility relation - A rough sets approach, ICCP 2007: IEEE *3rd International Conference on Intelligent Computer Communication and Processing*, Proceedings, (2007), vol, pp. 279-282

[3] Brtka V., Berkovic I., Brtka E., Jevtic V., A Comparison of Rule Sets Induced by Techniques Based on Rough Set Theory, SISY 2008, *6th International Symposium on Intelligent Systems and Informatics*, September 26-27, 2008 Subotica, Serbia, IEEE Catalog Number: CFP0884C-CDR, ISBN: 978-1-4244-2407-8, Library of Congress: 2008903275.

[4] Brtka V., Berkovic I., Stokic E., Srdic B., Automated extraction of decision rules from medical databases - A rough sets approach (Proceedings Paper), 2007 *5TH INTERNATIONAL SYMPOSIUM ON INTELLIGENT SYSTEMS & INFORMATICS*, (2007), vol. br. , str. 27-31

[5] Brtka V., Stokic E., Srdic B., "Automated extraction of decision rules for leptin dynamics—A rough sets approach", *Journal of Biomedical Informatics* 41, pp. 667 – 674, 2008.

[6] Data Mining Tools See5 and C5.0, www.rulequest.com/see5-info.htlm

[7] Giuffrida G., Chu W. W., Hanssens M. D., Mining Classification Rules from Datasets with Large Number of Many-Valued Attributes, http://www.anderson.ucla.edu/faculty/dominique. hanssens/content/mining.pdf (2012)

[8] Hafizah S. J., Marriyam S. S., Bariah Y., Munira I., (2009), A Predictive Model Construction Applying Rough Set Methodology for Malaysian Stock Market Returns, International Research, *Journal of Finance and Economics*, ISSN 1450-2887, Issue (30), pp. 211-218.

[9] Holte R. C.; Very Simple Classification Rules Perform Well on Most Commonly Used. Datasets; *Machine Learning*, Vol. 11, pp 63-90., (1993)

[10] Komorowski J., Pawlak Z., Polkowski L., Skowron A., "Rough Sets: A Tutorial", http://citeseer.ist.psu.edu/komorowski98rough.html, 1998.

[11] Ognjenovic V., Brtka V., Jovanovic M., Brtka E., Berkovic E., "The Representation of Indiscernibility Relation by Graph", Proceedings of *9th International Symposium on Intelligent Systems and Informatics*, pp. 91-94, (2011).

[12] Pawlak, Z.: Rough sets. International *Journal of Computer and Information Sciences* 11, 341–356 (1982)

[13] Pawlak, Z.: Rough sets: Theoretical Aspects of Reasoning about Data. Kluwer *Academic Publishers*, Dordrecht (1991)

[14] Quinlan, J.R. C4.5: Programs for Machine Learning. Morgan Kaufmann. San Mateo, CA, (1993)

[15] Rendell, L., Seshu, R. , Learning Hard Concepts Through Constructive Induction. *Computational Intelligence*, 6, 247−270., (1990)