RULE-BASED CLASSIFIERS FOR THE ACUTE ABDOMINAL PAIN DIAGNOSIS – COMPARATIVE STUDY

Michal Wozniak[1]

[1]Wroclaw University of Technology, Faculty of Electronics
Wybrzeze Wyspianskiego 27, 50-370 Wroclaw, Poland

Michal.Wozniak@pwr.wroc.pl (Michal Wozniak)

Abstract

The inductive learning algorithms are attractive methods generating hierarchical classifiers. They generate the hypothesis of the target concept on the basis of the set of labeled examples. These methods are typical for the medical decision support systems because for many cases the physicians can not formulate the rules, whose are used to make decision or the formulated set of rules is incomplete. Therefore if we can not obtain high quality original expert knowledge we can generate knowledge base on the basis of learning set.
This paper presents the comparative study of classification quality of heuristic classifier (given by experts) and popular inductive methods: C4.5, FID and AQ, and their boosted versions. Algorithms C4.5 and FID are the modifications of ID3 method generating decision tree. The AQ algorithm bases on the sequential covering strategy, which removes elements of learning set covered by any generated rule in each iteration. Metaclassifier like boosting is general method of improving quality of weak and unstable classifiers. The idea of boosting has its root in PAC theory. The underlying idea of boosting is to combine simple classifiers to form an ensemble which makes better decision than any simple classifier.
Evaluation of presented concepts were made on the basis of computer experiments. All tests were done for the acute abdominal pain decision problem. The superiority of the obtained results for the inductive learning classifiers over heuristic one demonstrates the effectiveness of the proposed concept in such computer-aided medical diagnosis problems. Advantages of the proposed methods make it attractive for a wide range of applications in medicine, which might significantly improve the quality of the care that the clinicians can give to their patients.

Keywords: boosting, machine learning, decision tree, medical decision support

Presenting Author's biography

Michal Wozniak is Assistant Professor of Computer Science in the Chair of Systems and Computer Networks, Faculty of Electronics, Wroclaw University of Technology, Poland. He received an M.S. degree in Biomedical Engineering from the Wroclaw University of Technology in 1992 and Ph.D. degrees in Computer Science, from the Wroclaw University of Technology in 1996. Dr. Wozniak has published over 80 papers and edited book "Computer Recognition Systems", Springer 2005. His research focuses on multiple classifier systems, machine learning, data and web mining, Bayes compound theory and teleinformatics.

1 Introduction

Machine learning [1] is the attractive approach for building decision support systems. For this type of software, the key-role plays the quality of the knowledge base. In many cases we can find following problems:

- the experts can not formulate the rules for decision problem, because they might not have the knowledge needed to develop effective algorithms (e.g. human face recognition from images),
- we want discover the rules in the large databases (data mining) e.g. to analyze outcomes of medical treatments from hospital information systems; this situation is typical for designing telemedical decision support system, which knowledge base is generated on the base on the large number hospital databases,
- decision support software has to dynamically adapt to changing condition.

These situations are typical for the medical knowledge. For many cases the physicians can not formulate the rules, whose are used to make decision or the formulated rule base is incomplete. Therefore if we can not obtain high quality original expert knowledge we can generate knowledge base on the basis of learning set using machine learning algorithms like decision tree induction algorithms [2-3] or algorithms based on sequential covering concept [5].

This paper presents the comparative study of classification quality of heuristic classifier (given by experts) and popular inductive methods: C4.5, FID and AQ, and their boosted versions. Algorithms C4.5 [3] and FID [4] are the modifications of ID3 [2] method generating decision tree. The AQ [5] algorithm bases on the sequential covering strategy, which removes elements of learning set covered by any generated rule in each iteration. There are several concepts how to stabilize and improve qualities of such classifiers. Metaclassifier like boosting is general method of improving quality of weak and unstable classifiers. The idea of boosting has its root in PAC theory. Generally idea of metaclassifier is based on repeated modifications of learning materials which are used for construction of new classifiers. Finally common decision is obtained via voting procedure.

The content of the work is as follows. Section 2 introduces idea of the inductive decision tree algorithms and learning sets of rules method. In Section 3 we describe mathematical model of the acute abdominal pain decision problem. Next section presents results of the experimental investigations of the algorithms. Section 4 concludes the paper.

2 Algorithms

We chose three of the inductive learning algorithm: (1) C4.5 algorithm given by R. J. Quinlan [2], (2) Fuzzy Decision Tree Algorithm FID 3.2 given by C. Janikow [3] and (3) rule generation algorithm - AQ given by R. Michalski [4].

2.1 Inductive decision tree

Algorithms C4.5 and FID are the modifications of ID3 method generating decision tree. Therefore let us present the main idea of the ID3 below

```
Create a Root node for tree
IF all examples are positive
      THEN return the single node
tree Root with label yes and return.
IF all examples are negative
      THEN return the single node
      tree Root with label no and
      return.
IF set of attributes is empty
      THEN return the single node
      tree Root with label = most
      common value of label in the
      set of examples and return
Choose "the best" attribute A from
the set of attributes.
FOR EACH possible value vi of
attribute
  1. Add new tree branch bellow Root,
     corresponding to the test A=vi.
  2. Let Evi be the subset of set of
     examples that has value vi for A.
  3. IF Evi is empty
      THEN bellow this new branches
      add a leaf node with label =
      most common value of label in
      the set of examples
      ELSE below this new branch add
      new subtree and do this
      function recursive.
END
RETURN Root
```

The central choice in the ID3 algorithm is selecting "the best" attribute (which attribute to test at each node in the tree). The proposed algorithm uses the information gain that measures how well the given attribute separates the training examples according to the target classification. This measure based on the Shanon's entropy of set *S:*

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

where p_i is the proportin of S belonging to klas i $(i \in \boldsymbol{M}, \boldsymbol{M} = \{1, 2, ..., M\})$.

The information gain of an attribute A relative to the collection of examples S, is defined as

$$Gain(S, A) = Entropy(S) - \sum_{c \in values(A)} \frac{|S_v|}{|S|} Entropy(S_v) \text{ ,(2)}$$

where $values(A)$ is the set of all possible values for attribute A and S_v is the subset of S for which $A = v$.

The C4.5 algorithm modifies ID3 that at the beginning the tree generation procedure does not use the whole set of examples. Fuzzy Decision Tree (FID) combines fuzzy representation, and its approximate reasoning, with symbolic decision trees. The FID algorithm assumes that attribute values are the fuzzy observations. FID has three major components: one for partitioning continuous attributes, one for building an explicit tree, and one for knowledge inference from the tree [4, 14].

One of the main advantage of this classifiers is that we can easy convert obtained tree into the set of rules.

2.2 Learning set of rules

The algorithms like CN2 [1] or AQ [5] based on the learning one rule (LOR) strategy, removing data it covers, then iterating the process. This sequential covering procedure is presented bellow.

```
Sequential_covering(examples)
R:= ∅.
P:= examples.
DO WHILE P<> ∅
r:=learn-one-rule (examples, P).
      R:=R∪r.
      remove from P all examples
      covered by r
END.
RETURN R.
```

The LOR method is similar to the ID3 algorithm presented above. The LOR algorithms follow only the most promising branch in the tree at the each step – returns only one rule, which covers at least some of the examples.

We have presented only idea of algorithm. Of course the method, we talk over, are more complicated. For example we do not present pruning methods whose protect us against situation, where obtained tree or rule set overfits the training set [1, 3, 8].

2.3 Boosting

Boosting is general method of producing an accurate classifier on basis of weak and unstable one[8, 9]. It is often called metaclassifier. The idea of boosting has its root in PAC (*Probably Approximately Correct*) theory. The underlying idea of boosting is to combine simple classifiers to form an ensemble such that the performance of the single member of ensemble is improved [7, 10]. As we see the main problem of the boosting is how to construct ensemble. The one of the most popular algorithm AdaBoost [11,12] produces at every stage, a classifier which is trained with the data set modified in each itteration. The output of the classifier is then added to the output of classifier ensemble, with the strength (proportional to how accurate obtained classifier is). Then, the data is reweighted: examples that the current learned function gets wrong are "boosted" in importance, so that the classifier obtained at the next stage will attempt to fix the errors. The main advantage of boosting is that it often does not suffer from overfitting. The pseudocode of AdaBoost.M1 procedure is presented bellow

Input:

1. sequence of m examples $S_1 = \{(x_1, y_1), (x_2, y_2), ..., (x_m, y_m)\}$ with labels $y_i \in Y = \{1, ..., k\}$

2. weak learning algorithm *WeakLearn*

3. integer T specifying number of iterations

for $i = 1, 2, ..., m$:

$D_1(i) = 1/m$

end.

for $t = 1, 2, ..., T$:

1. Call WeakLearn based on S_t

2. Get back a hypothesis $h_t : X \to Y$.

3. Calculate the error of h_t :
$$\varepsilon_t = \sum_{i:\, h_t(x_i) \neq y_i} D_t(i) \text{ .}$$

4. If $\varepsilon_t > 1/2$, then set $T := t - 1$ and abort loop.

5. Set $\beta_t = \varepsilon_t / (1 - \varepsilon_t)$.

6. Update distribution D_t :
$$D_{t+1} = \frac{D_t}{Z_t} \times \begin{cases} \beta_t & \text{if } h_t(x_i) = y_i \\ 1 & \text{otherwise} \end{cases}$$

where Z_t is a normalization constant (chosen so that D_{t+1} will be a distribution).

7. Generate S_{t+1} according the distribution D_{t+1} .

end.

ISBN 978-3-901608-32-2

`Output: the final hypothesis:`

$$h_{fin}(x) = \arg\max_{y \in Y} \sum_{t:\, h_x(x)=y} \log \frac{1}{\beta_t} .$$

3 Model of acute abdominal pain diagnosis

The mathematical model of the diagnosis of acute abdominal pain (AAP) was simplified. Hover the experts from the Clinic of Surgery, Wroclaw Medical Academy, regarded that stated problem of diagnosis as very useful.

It leads to the following classification of the AAP:

1. cholecystitis,
2. pancreatitis,
3. non-specyfic abdominal pain,
4. rare disorders of "acute abdominal",
5. appendicitis,
6. divercitulitis,
7. small-bowel obstruction,
8. perforated peptic ulcer.

Although the set of symptoms necessary to correctly assess the existing APP is pretty wide, in practice for the diagnosis, results of 36 (non-continuos) examinations are used, whose are presented in table 1.

3.1 Heuristic decision tree

The experts-physicians gave the decision tree [4] depicted in Fig.1. Numbers of leafs are the numbers of diagnosis presented above. The numbers in the nodes are corresponded with the following diagnosis:

9. acute enteropathy,
10. acute disorders of the digestive system,
11. others.

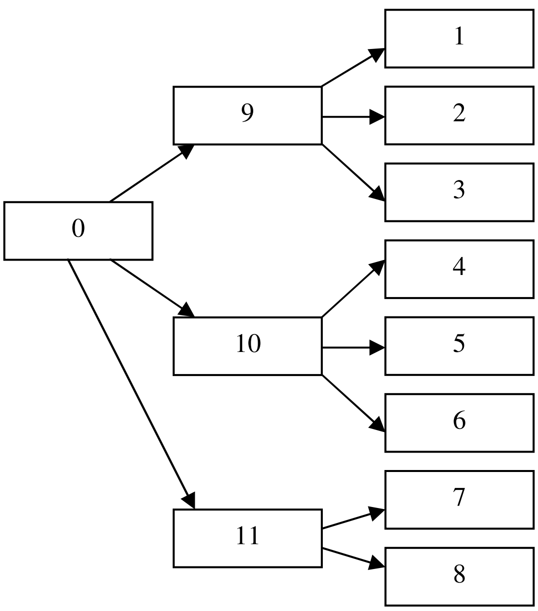

Fig. 1. Heuristic classifier for the APP diagnosis problem

Tab.1. Clinical feature considered

no	feature	no	feature
1	sex	19	previous surgery (abdominal)
2	age	20	drugs
3	site on onset	21	mood
4	Site on present	22	color
5	intensity	23	temperature
6	aggravating factors	24	pulse
7	relieving factors	25	systolic blood pressure
8	progress	26	diastolic blood pressure
9	duration	27	movement (of abdomen)
10	character on onset	28	distension
11	character on present	29	tenderness (site)
12	nausea and vomiting	30	Blumberg's sign
13	appetite	31	guarding
14	bowels	32	rigidity
15	micturition	33	swellings
16	previous indigestion	34	Murphy's sign
17	jaundice	35	abdominal auscultation (bowel sounds)
18	previous similar pain	36	rectal examinations

4 Experimental investigation

4.1 Experiment conditions

The presented algorithms C4.5, FID and AQ were used for creating rules for APP decision problem. Their frequencies of correct classification were compared with quality of heuristic classifier [9, 10] and qualities of the boosted versions of C4.5, FID, AQ. In our experiments number of the iterations of *boosting* algorithm was established on 20, because number increasing the iteration is not influencing (for the most cases) on qualities of the classifiers obtained via *boosting* method [15].

The set of data has been gathered in the Surgery Clinic. It contains 476 learning examples. For each learning method the following experiment was made:

- from the learning set 40 examples were chosen (according with frequency of the class appearance); this set was use for test,
- the rest of examples (436) were training ones.

This procedure was repeated 20 times for each of the algorithms. For this purpose we modified the C4.5 algorithm source code. The results of the experiments are presented in Table 2.

ISBN 978-3-901608-32-2

Tab. 2. Frequency of correct classifications

Class number	**Heuristic**	**AQ**	**Boosted AQ**	**C4.5**	**Boosed C4.5**	**FID**	**Boosted FID**
1	79,1	90,5	87,1	95,8	92,1	86,7	89,9
2	88,2	55,0	68,2	92,3	91,9	100,0	98
3	93,1	95,0	85,2	95,8	93,2	100,0	95,9
4	67,1	90,0	88,7	95,6	94,2	66,7	78,4
5	82,5	98,8	96,2	86,9	91,1	83,3	88,2
6	84,4	85,0	88,1	96,2	92,3	75,0	79,8
7	84,7	97,5	94,5	91.5	90,8	75,0	84,7
8	88,2	75,0	81,1	92,3	89,4	50,0	74,2
Average	**83,4**	**85,9**	**86,1**	**93,6**	**91,9**	**79,6**	**86,1**

4.2 Computer experiment evaluation

The results of test are clear. The classifier given by C4.5 algorithm is always better than heuristic one. The AQ and FID algorithm gives the better results for some of class, but for another the frequency of correct classification is very low. As we see boosting improved the quality of AQ and FID but their qualities is still worse than quality of C4.5. We can also observe that boosting did not improve quality of C4.5. Probably classifier obtained via C4.5 procedure was not weak and it could not be improved. The similar situation was described in [13].

Experts revised the structures of classifiers given by inductive learning algorithms. They confirmed that all of rules were correct and maybe the heuristic classifier was incomplete.

5 Conclusion

The methods of inductive learning and concept of metaclassifier were presented. The classifiers generated by these algorithms were applied to the medical decision problem (recognition of Acute Abdominal Pain). The results of test were compared with recognition quality of algorithm based on the heuristic decision tree.

It must be emphasised that we have not proposed a method of "computer diagnosis". What we have proposed are the algorithms whose can be used to help the clinicians to make their own diagnosis. The superiority of the presented empirical results for the inductive learning classifiers over heuristic one demonstrates the effectiveness of the proposed concept in such computer-aided medical diagnosis problems. Advantages of the proposed methods make it attractive for a wide range of applications in medicine, which might significantly improve the quality of the care that the clinicians can give to their patients.

6 Acknowledgments

This work is supported be The Polish State Committee for Scientific Research under the grant which is realizing in years 2006-2009.

References

[1] Mitchell T.M., Machine Learning, McGraw-Hill Comp., Inc, New York 1997

[2] Quinlan J.R., Induction on Decision Tree, *Machine Learning*, Vol.1, 1986, pp.81-106

[3] Quinlan J.R., *C4.5: Program for Machine Learning*, Morgan Kaufman, San Mateo, CA 1993

[4] Janikow C.Z., Fuzzy Decision Tree: Issues and Methods, *IEEE Trans on Man and Cybernetics*, vol 28, Issue 1, 1998

[5] Michalski R.S., A Theory and Methodology of Inductive Learning, *Artificial Intelligence*, no 20, 1983, pp.111-116

[6] Kurzynski M. Multistages *Pattern Recognition Algorithms and their Medical and Technical Implementations*, Wroclaw University of Technology Press, Wroclaw Poland 1987 (in polish).

[7] Schapire R. E., The boosting approach to machine learning: An overview. Proc. Of MSRI Workshop on Nonlinear Estimation and Classification, Berkeley, CA, 2001.

[8] Shapire R.E., The Strength of Weak Learnability, *Machine Learning*, No. 5, 1990, pp. 197-227.

[9] Schapire R.E., A Brief Introduction to Boosting, *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence*, 1999.

[10] Meir R. and Rätsch G. An introduction to boosting and leveraging. In S. Mendelson and A. Smola, editors, *Advanced Lectures on Machine Learning*, LNCS, pages 119-184. Springer, 2003.

[11] Freund Y., Schapire R.E., A decision-theoretic generalization of on-line learning and application

to boosting, *Journal of Computer and System Science*, 55(1),1997, pp. 119-139.

[12]Freund Y., Schapire R.E., Experiments with a New Boosting Algorithm, *Proceedings of the International Conference on Machine Learning*, 1996, pp. 148-156.

[13]Quinlan J.R., Bagging, Boosting, and C4.5, *Proceedings of the 13th National Conference on Artificial Intelligence and Eighth Innovative Applications of Artificial Intelligence Conference, AAAI 96, IAAI 96*, Volume 1, Portland, Oregon, August 4-8, 1996, pp. 725-230.

[14]Fajfer M., Janikow C.Z., Bottom-up partitioning in fuzzy decision trees, P*roceedings of the 19th International Conference of the North American Fuzzy Information Society*, Atlanta 2000, pp. 326-330.

[15]Quinlan J.R., Bagging, Boosting, and C4.5, *Proc.of the AAAI 96, IAAI 96*, Volume 1, Portland, Oregon, August 4-8, 1996, pp. 725-230.