Using Case-Based Reasoning in Interpreting Unsupervised Inductive Learning Results *

Tu Bao Ho
School of Information Science
Japan Advanced Institute
of Science and Technology
Tatsunokuchi, Ishikawa 923-12, Japan

Chi Mai Luong
Institute of Information Technology
National Center for
Natural Science and Technology
Nghiado, Tuliem, Hanoi, Vietnam

Abstract

The objective of this work is to interpret inductive results obtained by the unsupervised learning method OSHAM. We briefly introduce the learning process of OSHAM, that extracts concept hierarchies from unlabelled data, based on a representation combining the classical, prototype and exemplar views on concepts. The interpretive process is considered as an intrinsic part in OSHAM and is carried out by a combination of case-based reasoning with matching approaches in inductive learning. An experimental comparative study of some learning methods in terms of knowledge description and prediction is given.

1 Introduction

Though the interpretation of induction results has a significant role in machine learning applications, there have been little work in inductive learning, particularly in unsupervised learning, associated with interpretation procedures, e.g., [Bergadano *et a*/., 1992], [Wu, 1996].

There are three broad classes of logical, threshold, and competitive interpreters for intensional concept descriptions [Langley, 1996]. Naturally, three types of outcomes occur when matching logically an unknown instance with learned concepts: no-match, single-match, and multiple-match. Most work dealing with the cases of no-match and multiple-match employ a probabilistic estimation, e.g., [Michalski *et a*/., 1986]. However, it is not always possible to obtain such an estimation in unsupervised learning when it requires using the class information. Moreover, the logical match of the concept intent does not always provide a prediction with enough satisfaction, particularly in boundary regions of concepts.

One of two styles of case-based reasoning (CBR) is interpretive by which new situations are evaluated in the context of old situations [Kolodner, 1992]. Rather than classifying new cases using the intensional concept descriptions, CBR typically does classification by using the nearest neighbor methods which have been demonstrated to be able to work often as well as other inductive learning techniques [Aha *et* al., 1991]. However, one limitation of the CBR is that it does not provide the concept description which is the main advantage of inductive learning about the knowledge understandability.

This paper highlights the intrinsic role of the interpretive process in unsupervised inductive learning and proposes a procedure that combines CBR with matching approaches in inductive learning to interpret the concepts learned by method OSHAM [Ho, 1996], [Ho, 1997]. The reason for this combination lies in the fact that the use of results in unsupervised learning, obtained by a non-exhaustive searching for regularities, can be suported by the nearest neighbor rule. The paper is organized as follows. Section 2 briefly resumes the learning phase in OSHAM consisting of an extended representation of concepts in the Galois lattice, and the essential ideas of the learning algorithm for extracting concept hierarchies from unsupervised data. Section 3 presents the interpretation phase of OSHAM to classify unknown cases using learned knowledge. Section 4 presents an experimental comparative study of four learning methods and the discussion. Section 5 is a short conclusion.

2 Learning Concept Hierarchies

2.1 Concept representation and extraction

Among *views on concepts* in cognitive science and machine learning, the classical, prototype and exemplar ones are widely known and used. Main strengths and limitations of these views on concepts have been widely recognized, e.g., [Van Mechelen *et* al., 1993], [Wrobel, 1994J. Moreover, without the class information, unsupervised learning systems often compose solutions by employing one or more of three main *categorization con-*

*This work is supported by Kokusai Electric Co., Ltd (Japan) to JAIST (Japan Advanced Institute of Science and Technology), and by the National Research Programme on Information Technology KC01-RD08 (Vietnam).

straints based on similarity, feature correlation, and syntactical structure of conceptual knowledge.

Recently, several concept learning systems have been developed using the classical view on concepts in the Galois lattice structure [Wille, 1982]. In [Godin and Missaoui, 1994], [Carpineto and Romano, 1996], the authors generate incrementally all possible concepts in the Galois lattice. In [Ho, 1995], an alternative approach to hierarchical conceptual clustering was proposed that extracts a part of the Galois lattice in the form of a concept hierarchy. Although the Galois lattice provides a powerful structure for learning concepts, the classical view on concepts in this framework has some considerable limitations, such as it does not capture typicality effects and vagueness. Otherwise, to find all possible concepts is not always tractable as in the worse case the number of concepts can be exponential in the size of datasets, e.g., even for the small well-known dataset of Congressional voting (435 instances x 17 attributes), the Galois lattice has about 150,000 nodes [Carpineto and Romano, 1996],

As analysed in [Van Mechelen *et al.*, 1993], each system relies on a single view on concepts can be limited in capturing the rich variety of conceptual knowledge. Therefore, hybrid systems attempt to improve the concept learning process by combining fairly different theoretical views on concept and categorization constraints. In [Ho, 1996], the learning method OSHAM was improved by an extension of the classical view of concepts in the Galois lattice. Instead of characterizing a concept only by its intent and extent, OSHAM represents each concept C_k in a concept hierarchy $\mathcal{H}$ by a 10-tuple

$$< l(C_k), f(C_k), s(C_k), i(C_k), e(C_k), \\ d(C_k), p(C_k), d(C_k^r), p(C_k^r \mid C_k), q(C_k) > \qquad (1)$$

where

- $l(C_k)$ is the level of C_k in $\mathcal{H}$;
- $f(C_k)$ is the list of direct superconcepts of C_k;
- $s(C_k)$ is the list of direct subconcepts of C_k;
- $i(C_k)$ is the intent of C_k which is the set of all common properties of instances of C_k;
- $e(C_k)$ is the extent of C_k which is the set of all instances satisfying properties of $i(C_k)$;
- $d(C_k)$ is the dispersion between instances of C_k;
- $p(C_k)$ is the occurrence probability of C_k;
- $d(C_k^r)$ is the dispersion of local instances of C_k which are not classified into subconcepts of C_k;
- $p(C_k^r \mid C_k)$ is the conditional probability of these unclassified instances of C_k;
- $q(C_k)$ is the quality estimation of splitting C_k into subconcepts C_{k_i}.

The induction and interpretation of components in (1) employ the following distance metrics. Denote by $\lambda(X)$ for any instance set X the largest set of properties common to all elements of X, and by $\rho(S)$ for any property set S the set of all instances satisfying S. The *distance* $\delta(o_p, o_q)$ between two instances o_p and o_q is defined as an extension of Jaccard distance

$$\delta(o_p, o_q) = 1 - \frac{\sum_{a \in \lambda(\{o_p, o_q\})} \gamma(a)}{\sum_{a \in \lambda(\{o_p\}) \cup \lambda(\{o_q\})} \gamma(a)} \qquad (2)$$

where $\gamma(a) \in \mathbb{Z}^+$ are positive integer weights of attributes a (with value 1 by default). The *dispersion* $d(C_k)$ between instances in $e(C_k)$, considered as the inverse of the homogeneity of $e(C_k)$, is defined as the average distance between all pairs of instances in $e(C_k)$

$$d(C_k) = \frac{2 \times \sum_{o_p, o_q \in e(C_k)} \delta(o_p, o_q)}{\mid e(C_k) \mid \times (\mid e(C_k) \mid -1)} \qquad (3)$$

If C_k is a non-leaf concept, its local instances in C_k^r can be considered to be more typical and representative than its instances classified into subconcepts C_{k_i}. As an instance o is member of different concepts along a branch in the concept hierarchy, the concepts C_k that $o \in C_k^r$ is of particular interest. If C_k is a leaf concept, we have $e(C_k) = C_k^r$ and all of its instances are considered with the same representative role.

The extent of all direct subconcepts $C_{k_1}, C_{k_2}, ..., C_{k_n}$ of C_k and the set of local instances C_k^r form a partition P of $e(C_k)$. Denote by $W(C_k)$ the average of all $d(C_{k_i})$ and $d(C_k^r)$. The dissimilarity between subconcepts of C_k, denoted by $B(C_k)$, is defined as the average of distances $\Delta(c(C_{k_i}), c(C_{k_j}))$ between all pairs C_{k_i}, C_{k_j} in P, where the distance $\Delta(c(C_{k_i}), c(C_{k_j}))$ is defined as the smallest distance among distances of all pairs $o_p \in C_{k_i}$, $o_q \in C_{k_j}$

Table 1: Brief description of learning process

1. While C_k is still splittable, find a new subconcept of it that corresponds to the hypothesis minimizing the quality function $q(C_k)$ among η hypotheses generated by the following steps
   (a) Find a "good" attribute-value pair concerning the best cover of C_k.
   (b) Find a closed attribute-value subset S containing this attribute-value pair.
   (c) Form a subconcept C_{k_i} with the intent is S.
   (d) Evaluate the quality function with the new hypothesized subconcept.

   Form intersecting concepts corresponding to intersections of the extent of the new concept with the extent of existing concepts excluding its superconcepts.
2. If one of the following conditions holds then C_k is considered as unsplittable
   (a) There exist not any closed proper feature subset.
   (b) The local instances set C_k^r is too small.
   (c) The local instances set C_k^r is homogeneous enough.
3. Apply recursively the procedure to concepts generated in step 1.

$$\Delta(c(C_{k_i}), c(C_{k_j})) = Min_{o_p \in O_{k_i}, o_q \in O_{k_j}} \delta(o_p, o_q) \qquad (4)$$

The *quality* of splitting a concept C_k into subconcepts in the next level, denoted by $q(C_k)$, is measured by

$$q(C_k) = W(C_k)/B(C_k) \qquad (5)$$

The basis of learning in OSHAM is a generate-and-test procedure to split a concept *C* into subconcepts at a higher level of $\mathcal{H}$. Starting from the root concept of the Galois lattice with the whole set of training instances, it extracts the concept hierarchy $\mathcal{H}$ recursively in a top-down direction. This algorithm is originally designed for discrete attributes with unordered nominal values. In the current version, continuous attributes are discretized before learning process by k-means clustering [Hartigan, 1975]. In fact, for each continuous attribute the k-means algorithm is applied to cluster its values into *k* groups (*k* = 1,2,...,K). A criterion similar to (5) with the Euclidean distance is used to choose a value of *k* that corresponds to the best partition according to this criterion. The basic idea of learning algorithm in OSHAM, described fully in [Ho, 1996], is resumed in Table 1.

2.2 About learned concept hierarchies

By using different constraints for I.(a) in the learning algorithm, OSHAM is able to extract both overlapping or disjoint concepts depending the user's interest [Ho, 1997]. OSHAM has been implemented in the X Window on a Sparcstation with the direct manipulation style of interaction which allows the user to participate actively in the learning process. The user can initialize parameters to cluster data, visualize the concept hierarchy gradually, observe the results and the quality estimation, manually modify the parameters when necessary before the system continues to go further to cluster subsequent data or backtrack to regrow branches of the concept hierarchy with respect to the categorization scheme. Figure 1 shows a main screen of the interactive OSHAM with a hierarchy of overlapping concepts learned from the Wisconsin breast cancer dataset. A full description of concept 43 in this figure is given below

CONCEPT 43
Level = 5
Super-Concepts = {29}, Sub-Concepts = {52, 53}
Features = (Uniformity of Cell Size, 1) $\wedge$ (Bare Nuclei, 1) $\wedge$ (Bland Chromatin, 1) $\wedge$ (Uniformity of Cell Shape, 2)
LocaUnstances/Coverec_instances = 6/25
LocaLinstances = {8, 127, 221, 236, 415, 661}
Concept_probabilrty = 0.041666
LocaUnatance.conditionaLprobability = 0.240000
Concept_dispersion = 0.258848
LocaLinstancejdispersion = 0.055556
Subconcept-part'rtion-quality = 0.519719

There is a considerable distinction in the concept description of OSHAM in contrast to those of other methods such as the supervised learning system C4.5 [Quintan, 1993], the unsupervised learning systems COBWEB [Fisher, 1987] and AUTOCLASS [Cheeseman and Stutz, 1996]. C4.5 induces decision trees in which concepts are represented by their intent associated with a predicted error rate, and it has not to maintain intermediate concepts. COBWEB represents each concept C_k as a set of attributes a, associated with a set of their possible values V_i the occurrence probability, and the conditional probability $P(a_i = v_{ij} \mid C_k)$ associated with each value v_{ij} .A *classification* in AUTOCLASS is defined as a set of classes, the probability of each class, and two additional probabilities for each hypothesized model: the model probability *P(H)* and the conditional parameter probability distribution *P(p* | H).

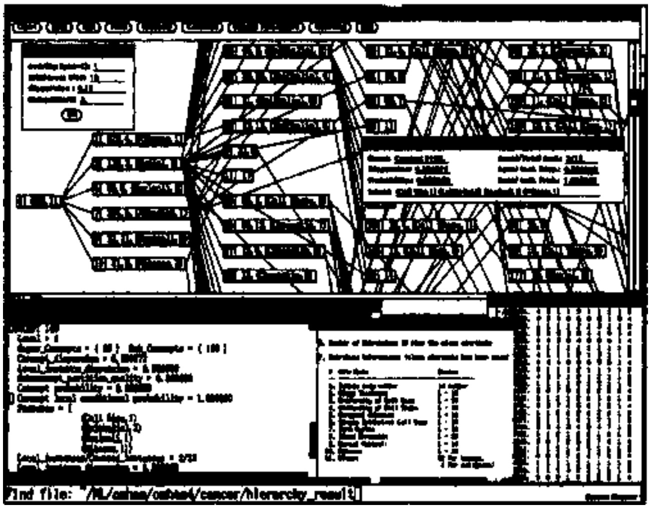


Figure 1: A screen of the interactive OSHAM

We share the opinion in [Langley, 1996] that the interpretive process is a central issue in learning. An intensional representation has no meaning (e.g., no extension) without some associated interpreters and different interpreters can yield different meaning for the same representation. Next section describes the second phase in OSHAM - its interpretive process.

3 Interpreting concept hierarchies

3.1 Interpretive CBR

Interpretive case-based learning is a process of evaluating situations in the context of previous experience. One way a case-based classifier works is to ask whether the unknown case is enough like another one known. It does classification by trying to find the closest matching case in its case base to the new case rather than using intensional concept descriptions. Many studies have pointed out the strong points of CBR (e.g., simplicity, relatively robust, often excellent performance, etc.) and its weak points, e.g., {Aha *et al.*, 1991], [Kolodner, 1992].

In inductive learning, the *logical* interpretation approach carries out an "all or none" matching process depending on whether the unknown instance satisfies the concept intent. The *threshold* approach carries out a partial matching process and employs some threshold to determine an acceptable degree of match. The *competitive* approach also carries out a partial matching process and selects the best competitor based on estimated degrees of match [Langley, 1996]. The interpretation of inductive learning results is commonly understood as the process of comparing an unknown case to the learned concepts. In OSHAM, as the generality decreases along branches of the concept hierarchy, we say that a concept C_k *matches* the unknown instance e if C_k is the most specific concept in a branch that matches e intensionally (though all superconcepts of C_k match e). Naturally, there are three types of outcomes when matching logically an unknown instance e with the learned concepts: only one concept that matches *e (single-match),* many concepts that match *e (multiple-match),* and no concept that matches *e (no-match).* We believe in the alternative roles of CBR and generalization in inductive learning. As noted in [Kolodner, 1992], rules could be used when they matched cases exactly, while cases would be used when rules were not immediately applicable. In fact, the nearest neighbor of *e* in the training set and the learned concept to which it belongs, denoted by *NN(e)* and c[NN(e)], provide useful information which could be used in all cases of single-match, multiple-match and no-match.

3.2 Interpretation of induction results

We develop an interpretation procedure for concept hierarchies that uses the concept intent, the hierarchical structure information, the probabilistic estimations and the nearest neighbors of unknown instances. This interpretation procedure consists of two stages: (1) find all concepts on the concept hierarchy that match *e* intensionally, and (2) decide among these concepts which matches *e* best. This procedure shares the same scheme of the system POSEIDON [Bergadano *et* a*l*., 1992], but functions differently. In the second stage, it determines the best matched concept of *e* with some satisficing degree of prediction.

Consider the multiple-match case when we have to decide among the competitors the best matched concept. To do it we need to determine and compare the degree of match of competitors. From various experiment case-studies we note that a logically matched concept C_k will match *e* well (with low error rate) if it satisfies a majority of the following conditions: $l(C_k)$ is high, C_k is a leaf concept, $p(C_k) \times p(C_k^r)$ is high, $d(C_k)$ is low, $d(C_k^r)$ is low, and generally none of these factors has a clearly higher priority than the others. Formally, the following functions $\tau_N, \tau_L, \tau_P, \tau_D, \tau_R$ can be used to compare these factors, respectively, between two concepts C_k and C_h which matched e intensionally

$$\tau_N(C_k, C_h) = \begin{cases} 1, & \text{if } l(C_k) > l(C_h) \\ 0, & \text{if } l(C_k) = l(C_h) \\ -1, & \text{if } l(C_k) < l(C_h) \end{cases} \qquad (6)$$

$$\tau_L(C_k, C_h) = \begin{cases} 1, & \text{if } C_k = \text{leaf} \wedge C_h \neq \text{leaf} \\ 0, & \text{if } C_k = \text{leaf} \wedge C_h = \text{leaf} \\ & \quad \vee C_k \neq \text{leaf} \wedge C_h \neq \text{leaf} \\ -1, & \text{if } C_k \neq \text{leaf} \wedge C_h = \text{leaf} \end{cases} \qquad (7)$$

$$\tau_P(C_k, C_h) = \begin{cases} 1, & \text{if } p(C_k) \times p(C_k^r \mid C_k) \\ & \quad > p(C_h) \times p(C_h^r \mid C_h) \\ 0, & \text{if } p(C_k) \times p(C_k^r \mid C_k) \\ & \quad = p(C_h) \times p(C_h^r \mid C_h) \\ -1, & \text{if } p(C_k) \times p(C_k^r \mid C_k) \\ & \quad < p(C_h) \times p(C_h^r \mid C_h) \end{cases} \qquad (8)$$

$$\tau_D(C_k, C_h) = \begin{cases} 1, & \text{if } d(C_k) < d(C_h) \\ 0, & \text{if } d(C_k) = d(C_h) \\ -1, & \text{if } d(C_k) > d(C_h) \end{cases} \qquad (9)$$

$$\tau_R(C_k, C_h) = \begin{cases} 1, & \text{if } d(C_k^r) < d(C_h^r) \\ 0, & \text{if } d(C_k^r) = d(C_h^r) \\ -1, & \text{if } d(C_k^r) > d(C_h^r) \end{cases} \qquad (10)$$

The following heuristic function τ compares the degree of match of C_k and C_h. We say that C_k has a *higher satisficing degree than* C_h in matching e if

$$\tau(C_k, C_h) = \theta_N \times \tau_N(C_k, C_h) + \theta_L \times \tau_L(C_k, C_h) + \theta_P \times \tau_P(C_k, C_h) + \theta_D \times \tau_D(C_k, C_h) + \theta_R \times \tau_R(C_k, C_h) > 0 \qquad (11)$$

Table 2: Interpretation procedure

```
Input      concept hierarchy $\mathcal{H}$, unknown instance $e$.
Result     best matched concept c[e], satisficing degree $\phi$.
Variables  $\sigma$ is a given threshold.

Procedure Interpretation($\mathcal{H}$, e, c[e], $\phi$)

If there is only one concept $C_k \in \mathcal{H}$ that matches $e$ intensionally then
    if $c[NN(e)] = C_k$ then $c[e] \leftarrow C_k, \phi \leftarrow S_1$
    else if $\tau(C_k, c[NN(e)]) \geq 0)$ then $c[e] \leftarrow C_k, \phi \leftarrow S_2$
    else $c[e] \leftarrow c[NN(e)], \phi \leftarrow S_3$.

If there are $m$ concepts $C_{i_1}, ..., C_{i_m} \in \mathcal{H}$ that match $e$ intensionally then
    Choose $i_K \in \{i_1, ..., i_m\}$ satisfying $\tau(C_{i_K}, C_{i_k}) \geq 0$
    when comparing $C_{i_K}$ with $C_{i_k}$ for all $i_k \in \{i_1, ..., i_m\}$.
    If $C_{i_K} = c[NN(e)]$ then $c[e] \leftarrow C_{i_K}, \phi \leftarrow M_1$
    else if $\tau(C_{i_K}, c[NN(e)]) \geq 0)$ then $c[e] \leftarrow C_{i_K}, \phi \leftarrow M_2$
    else $c[e] \leftarrow c[NN(e)], \phi \leftarrow M_3$.

If there is not any concept that match $e$ intensionally then
    if $\delta(NN(e), e) \leq \sigma$ then $c[e] \leftarrow c[NN(e)], \phi \leftarrow N_1$
    else $c[e] = \varnothing, \phi \leftarrow N_2$.
```

where $\theta_N, \theta_L, \theta_P, \theta_D, \theta_R$ are positive weights for the importance of the level, leaf concept, local instance conditional probability, concept dispersion, and local instance dispersion (all with value 1 by default).

Denote by *c[e]* the best matched concept that e is finally decided to belong to, and by ϕ the *satisficing degree* of prediction. Table 2 presents the interpretation procedure in OSHAM based on the function τ and the nearest-neighbor principle. Essentially, this procedure makes final decision by comparing the best concept matching c with the concept *c[NN(e)]* containing the nearest neighbor *NN(e)* of e regarding the function τ. In this interpretation procedure, different symbolic values are assigned to the satisficing degree of prediction ϕ.

Values of ϕ indicate the decreasing rank of prediction satisfaction. For example, we may consider S_1 as "best prediction", M_1 as "strong prediction" while N_1 as "weakly accepted prediction" and N_2 as "no prediction". The interpretation for different values of ϕ depends on the judgment of the user or domain experts.

4 Evaluation

4.1 Experimental Results

A way to evaluate unsupervised learning system is to employ supervised data but hide the class information in the whole learning and interpreting phases and use the class information only to estimate the predictive accuracy. We employ this way to evaluate unsupervised learning systems where the predicted name of each learned concept C_k is determined by the most frequently occurring name of instances in $e(C_k)$. With this predicted name of learned concepts, the error rate of an unsupervised learning system can be estimated as the ratio of the number of testing instances correctly predicted regarding the predicted name over the total number of testing instances. It is worth mentioning that multiple train-and-test experiments are much more computationally expensive but give more reliable evaluation than a single train-and-test experiment.

Experiments are carried out on ten datasets from the UCI repository of machine learning databases, including the Wisconsin breast cancer (breast-w), Congressional voting (vote), Mushroom (mushroom), Tic-tactoe (tictactoe), Glass identification (glass), Ionosphere (ionosphere), Waveform (waveform), Pima diabetes (diabetes), Thyroid (new) disease (thyroid), and Heart disease Cleveland (heart-c). The numbers of attributes (discrete and continuous), instances and "natural" classes of these datasets are given in columns 2-5 of Table 3. All experiments on these datasets are carried out with 10-fold cross validation by four programs C4.5 [Quinlan, 1993], CART-like [Breiman et al., 1984], AUTOCLASS and OSHAM in the same condition, i.e., the same randomly divided datasets into subsets. For AUTOCLASS, we use the public version AUTOCLASS-C implemented in C and run three steps of *search, report* and *predict* with the default parameters. The predicted name and predictive accuracy of AUTOCLASS and OSHAM are obtained as mentioned above. Columns 6-8 report the predictive accuracies of C4.5, CART-like and AUTOCLASS, respectively.

For OSHAM we carried out experiments for two interpretation procedures: OSHAM-NN fusing only the nearest neighbors) and **OSHAM-NN+τ** (using the nearest neighbors and matching regarding the function τ by the procedure described in Table 2). Experimental results are reported in columns 9-10, respectively. In order to avoid a biased evaluation of OSHAM, although with each dataset parameters can be adjusted to obtain the most suitable concept hierarchy, we fixed values $\alpha = 1\%$ of the size of the training set, $\beta = 15\%$ **and** $\sigma = 10\%$ of the number of attributes, and the beam size $\eta = 3$, commonly to all datasets. Two last columns in Table 3 give the average size of concept hierarchies (number of concepts) and CPU time (in second) of OSHAM learned from these datasets.

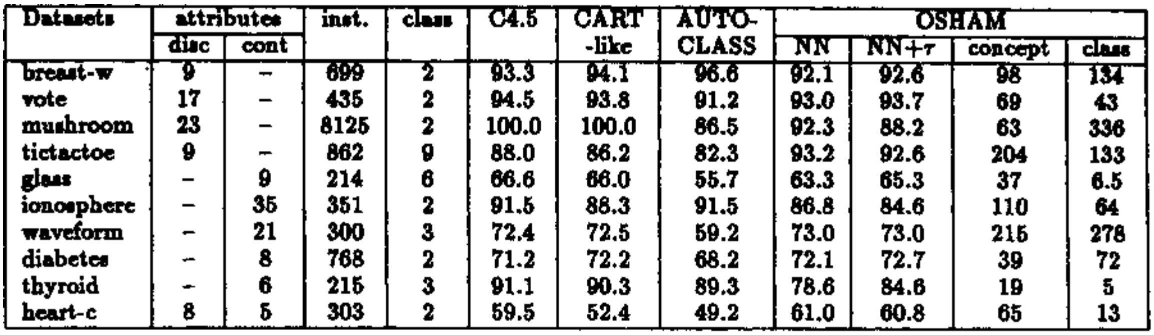
4.2 Discussion

The predicted name obtained in OSHAM and AUTOCLASS by the majority of occurring name of instances in concepts is different from the concept name obtained in supervised learning (e.g., C4.5) using the pruning threshold based on the class information. An unsupervised concept in the worse case may contain nearly equal numbers of instances belonging to different natural classes, and an unsupervised classification may be failed in distinguishing very similar instances. It explains that while the predictive accuracies between these supervised and unsupervised methods look not so different, they are slightly different in nature. Note that the recent release 8 of C4.5 [Quinlan, 1996] treats the continuous attribute better than the release we used in this work.

The predictive accuracies of OSHAM and AUTOCLASS in these experiments are only slightly different. In these first trials, each system is better in several datasets and these two systems can be considered having comparable performance. One advantage of OSHAM is its concept hierarchies can be easily understood by its extended classical view on concepts and the graphical support.

Empirical results with OSHAM-NN and OSHAM-**NN+τ** illustrate that these strategies are both good for interpreting unsupervised induction results. We believe that in general CBR can also be used to interpret results of unsupervised learning and this topic is worth for a further investigation.

Table 3: Datasets, Predictive accuracies of methods, Average size and CPU time of OSHAM



Datasets	attributes		inst.	class	C4.5	CART	AUTO-	OSHAM			
	disc	cont				-like	CLASS	NN	NN+τ	concept	class
breast-w	9	–	699	2	93.3	94.1	96.6	92.1	92.6	98	134
vote	17	–	435	2	94.5	93.8	91.2	93.0	93.7	69	43
mushroom	23	–	8125	2	100.0	100.0	86.5	92.3	88.2	63	336
tictactoe	9	–	862	9	88.0	86.2	82.3	93.2	92.6	204	133
glass	–	9	214	6	66.6	66.0	55.7	63.3	65.3	37	6.5
ionosphere	–	35	351	2	91.5	88.3	91.5	86.8	84.6	110	64
waveform	–	21	300	3	72.4	72.5	59.2	73.0	73.0	215	278
diabetes	–	8	768	2	71.2	72.2	68.2	72.1	72.7	39	72
thyroid	–	6	215	3	91.1	90.3	89.3	78.6	84.6	19	5
heart-c	8	5	303	2	59.5	52.4	49.2	61.0	60.8	65	13

5 Conclusion

In this paper we first briefly resumed the main ideas of the unsupervised learning method OSHAM in terms of description and extraction of concepts. We then described how the nearest neighbor rule is combined with domain knowledge to interpret the induction results. Careful experiments with different datasets have demonstrated that this combination provides a good solution to the use of unsupervised learned knowledge in prediction. The main conclusions can be drawn from this research are (1) the interpretive process needs to be a part of a unsupervised learning method, and (2) the CBR can be used to interpret results obtained from the non-exhaustive search in unsupervised inductive learning. Our near future research concerns further investigations on the effects of k-nearest neighbors and the discretization of continuous attributes to the interpretation of unsupervised induction results, based on experiments with a larger number of datasets.

References

[Aha *et al.*, 1991] D.W. Aha, D. Kibler, M.K. Albert. Instance-based learning algorithms. *Machine Learning,* Vol. 6 (1991), 37-66.

[Bergadano *et al.*, 1992] F. Bergadano, S. Matwin, R.S. Michalski, J. Zhang. Learning two-tiered descriptions of flexible concepts: the POSEIDON system. *Machine Learning,* Vol 8 (1992), 5-43.

[Breiman et al., 1984] L. Breiman, J. Friedman, R. Olshen, C. Stone. *Classification and Regression Trees.* Belmont, CA: Wadsworth, 1984.

[Carpineto and Romano, 1996] C. Carpineto and G. Romano. A lattice conceptual clustering system and its application to browsing retrieval. *Machine Learning,* Vol. 10 (1996), 95-122.

[Cheeseman and Stutz, 1996] P. Cheeseman and J. Stutz. Bayesian classification (AutoClass): Theory and results. In *Advances in Knowledge Discovery and Data Mining,* U.M. Fayyad *et al* (Eds.). AAAI Press/MIT Press, 1996, 153-180.

[Fisher, 1987] D. Fisher. Knowledge acquisition via incremental conceptual clustering. *Machine Learning,* Vol.2 (1987), 139-172.

[Godin and Missaoui, 1994] R. Godin and R. Missaoui. An incremental concept formation approach for learning from databases. *Theoretical Computer Science,* 133 (1994), 387-419.

[Hartigan, 1975] J.A. Hartigan. *Clustering Algorithms.* Wiley, New York, 1975.

[Ho, 1995] T.B. Ho. An approach to concept formation based on formal concept analysis. *IEICE Trans. Information and Systems.* Vol. E78-D (1995), No.5, 553-559.

[Ho, 1996] T.B. Ho. A hybrid model for concept formation. In *Information Modelling and Knowledge Bases* VII, Y. Tanaka *et al.* (Eds.). IOS Publisher, 1996, 22-35.

[Ho, 1997] T.B. Ho. Discovering and Using Knowledge From Unsupervised Data. In *Decision Support Systems,* June 1997 (in press).

[Kolodner, 1992] J. Kolodner. An introduction to Case-Based Reasoning. *Artificial Intelligence Review,* Vol. 6 (1992), 3-34.

[Langley, 1996] P. Langley. *Elements of Machine Learning.* Morgan Kaufmann, 1996.

[Michalski *et al,* 1986] R.S. Michalski, I. Mozetic, J. Hong, N. Lavrac. The multi-purpose incremental learning systems AQ15 and its testing application to three medical domains. In *Proceedings of AAAI 1986,* 1041-1045.

[Quinlan, 1993] J.R. Quinlan. (C_4.5: *Programs for Machine Learning.* Morgan Kaufmann, 1993.

[Quinlan, 1996] J.R. Quinlan. Improved use of continuous attributes in C4.5. *Journal of Artificial Intelligence Research,* Vol. 4 (1996), 77-90.

[Van Mechelen *et al,* 1993] I. Van Mechelen, J. Hampton, R.S. Michalski, P. Theuns (Eds.)., *Categories and Concepts. Theoretical Views and Inductive Data Analysis.* Academic Press, 1993.

[Wille, 1982] R. Wille. Restructuring lattice theory: an approach based on hierarchies of concepts. In *Ordered Sets,* I. Rival (Ed.). Reidel, 1982, 445-470.

[Wrobel, 1994] S. Wrobel. *Concept Formation and Knowledge Revision.* Kluwer Academic Publishers, 1994.

[Wu, 1996] X. Wu. Hybrid interpretation of induction results. In *Advanced IT Tools,* N. Terashimaand E. Altman (Eds.). Chapman & Hall, 1996, 497-506.