PAPER *Special Issue on Algorithmic Learning Theory*

An Approach to Concept Formation Based on Formal Concept Analysis

Tu Bao HO†, *Nonmember*

SUMMARY Computational approaches to concept formation often share a top-down, incremental, hill-climbing classification, and differ from each other in the concept representation and quality criteria. Each of them captures part of the rich variety of conceptual knowledge and many are well suited only when the object-attribute distribution is not sparse. Formal concept analysis is a set-theoretic model that mathematically formulates the human understanding of concepts, and investigates the algebraic structure, Galois lattice, of possible concepts in a given domain. Adopting the idea of representing concepts by mutual closed sets of objects and attributes as well as the Galois lattice structure for concepts from formal concept analysis, we propose an approach to concept formation and develop OSHAM, a method that forms concept hierarchies with high utility score, clear semantics and effective even with sparse object-attribute distributions. In this paper we describe OSHAM, and in an attempt to show its performance we present experimental studies on a number of data sets from the machine learning literature.

key words: machine learning, concept formation, formal concept analysis, concept lattice, concept hierarchy.

1. Introduction

While supervised learning (learning from examples) is relatively well developed and understood, unsupervised learning (concept formation and discovery) is certainly still a great challenge in machine learning. In recent years there has been a growing interest in concept formation and several computational models have been developed for its two simultaneous tasks:

- *Given* a set of object descriptions;
- *Find* a hierarchical clustering that determines useful subsets of objects (*clustering*);
- *Find* intensional definitions for these subsets of objects (*characterization*).

Concept formation differs from the traditional numerical unsupervised clustering in finding not only hierarchical clusters of data but also their 'good' conceptual descriptions. It differs from inductive learning from examples in dealing with objects which are not assigned to classes *a priori*. Although concept formation methods often share a top-down, incremental, hill climbing classification with the assumption that each object is described as a point in a discrete p-dimensional (attribute) space, they differ mainly from each other in the concept representation and criteria of goodness (biases). Most concept formation methods employ one of the three well known views on concepts (classical, probabilistic and exemplar) and their criteria used to modify and evaluate the concept hierarchy are mostly local and/or heuristic [4], [5].

CLUSTER/2 [9] is one influential conceptual clustering method that forms categories with 'good' conjunctions of common features to all category members. Its bias is to prefer short and specific conjunctive descriptions. AUTOCLASS [1] employs a probabilistic representation for each cluster and a Bayesian method to form clusters. Its bias is defined by a collection of prior probability distribution over the space of clusters including priors on the number of true clusters and priors on their attributes. UNIMEM [7] organizes concepts in the hierarchy with an concept description as conjunction of attribute-value pairs. The search in UNIMEM is guided by the idea of predictiveness and predictability though these notions are not formally defined with clear semantics. COBWEB [3] is often referred to as having many positive characteristics. It employs a probabilistic representation for concepts and a heuristic measure called category utility. The search of COBWEB is guided by its bias of finding a set of clusters that maximizes category utility. Like COBWEB, ARACHNE [8] represents knowledge as a hierarchy of probabilistic concepts, and focuses on the structural quality of the hierarchies while maintaining high predictive accuracy.

Concept formation systems are often based on a measure of similarity between objects and, depending on the used measure, many of them are well suited only when objects are described in terms of a fixed number of attributes. However, in many real-world situations objects are not described by the same attributes. Often, only a few attributes are related to a given object and the description of this object does not concern other attributes in the whole set of attributes. For example, each disease is described by a small number of symptoms from the set of all symptoms for certain diseases. In these cases the object-attribute distribution is always sparse.

Formal concept analysis, which has been developed during the last ten years by Wille and his colleagues,

Manuscript received September 14, 1994
Manuscript revised January 18, 1995

†The author is with the School of Information Science, Japan Advanced Institute of Science and Technology – Hokuriku, Tatsunokuchi, Ishikawa, 923-12 Japan

for example, [11], [12], [2], is a set-theoretic model that mathematically formulates the human understanding of concepts, and studies the algebraic structure of possible concepts in a given domain. Adopting the idea of representing concepts by mutual closed sets of objects and attributes as well as the Galois lattice structure for concepts in formal concept analysis, we propose in this paper OSHAM (standing for Making Automatically a Hierarchical Structure of Objects), a novel concept formation method which is highly comprehensible and effective even with sparse object-attribute distributions. In section 2, we briefly present formal concept analysis, the method OSHAM, and prove its correctness. Section 3 provides empirical results of OSHAM, particularly with data sets extracted from the UC Irvine Repository of Machine Learning Databases. Section 4 concludes with a summary, related and future works.

2. Description of OSHAM

2.1 Formal Concept Analysis

Formal concept analysis aims to formulate the philosophical understanding of a concept as a unit of two parts: its *extent* (all objects belong to the concept) and its *intent* (collection of all attributes shared by all those objects). Formal concept analysis may be considered to play a similar role in unsupervised learning as version spaces [10] do in supervised learning. As it is often difficult to list all objects belonging to a concept and usually impossible to list all its attributes, it is natural to work within a specific *context* in which the objects and attributes are observed.

Definition 1: A *context* is a triple (O, A, R) where O be a set of objects, A be a set of attributes and R be a binary relation between O and A, i.e., $R \subseteq O \times A$ and $(o, a) \in R$ is understood as the fact that object o has attribute a.

The current version of OSHAM limits itself to symbolic attributes, each attribute a has a finite set of values, $dom(a) = \{v_{a_1}, v_{a_2}, ..., v_{a_m}\}$. In that context, saying "object o has attribute a" means that object o has attribute a with some value v_{a_i}. For the simplicity of representation, we describe OSHAM with binary attributes. OSHAM has been implemented for multivalued attributes and may be extended to more sophisticated context such as mixed numeric and symbolic domains, predicate logic, etc.

Definition 2: Two *derivation operators* $\rho : \wp(A) \rightarrow \wp(O)$ and $\lambda : \wp(O) \rightarrow \wp(A)$, where $\wp(O)$ and $\wp(A)$ are the power sets of O and A, are determined as follows:

$$S \subseteq A, \quad \rho(S) = \{o \in O \mid \forall a \in S, (o, a) \in R\}$$
$$X \subseteq O, \quad \lambda(X) = \{a \in A \mid \forall o \in X, (o, a) \in R\}$$

Definition 3: An attribute subset S of A is called *closed* if $\lambda\rho(S) = S$ (X of O is *closed* if $\rho\lambda(X) = X$). For all $S \subseteq A$ the subset $\rho(S)$ is closed and is called to be *generated by* S (for all $X \subseteq O$ the subset $\lambda(X)$ is closed and called to be *generated by* X).

We are particularly interested in the subsets of objects and attributes which are closed under the operators ρ and λ. The main reason is that, restricted to these closed sets, ρ and λ are two order-reversing one-to-one operators. The following theorem indicates a natural "duality" between objects and attributes.

Theorem 1: The mappings λ and ρ define a Galois connection between $\wp(O)$ and $\wp(A)$.

As a consequence, the following properties hold:

$$\text{if} \quad S_1 \subseteq S_2 \quad \text{then} \quad \rho(S_1) \supseteq \rho(S_2) \quad \text{and} \quad \lambda\rho(S_1) \subseteq \lambda\rho(S_2)$$
$$\text{if} \quad X_1 \subseteq X_2 \quad \text{then} \quad \lambda(X_1) \supseteq \lambda(X_2) \quad \text{and} \quad \rho\lambda(X_1) \subseteq \rho\lambda(X_2)$$
$$S \subseteq \lambda\rho(S), \quad X \subseteq \rho\lambda(X)$$
$$\rho\lambda\rho = \rho, \quad \lambda\rho\lambda = \lambda, \quad \lambda\rho(\lambda\rho(S)) = \lambda\rho(S)$$
$$\rho(\bigcup_j S_j) = \bigcap_j \rho(S_j), \quad \lambda(\bigcup_j X_j) = \bigcap_j \lambda(X_j)$$

Definition 4: A *concept* C is a pair (X, S) with $X \subseteq O$, $S \subseteq A$ satisfying $\rho(S) = X$ and $\lambda(X) = S$ (i.e., X and S are closed). X and S are called the *extension* and the *intension* of C, respectively.

The set of all concepts (X, S) of the context (O, A, R) is denoted by $\mathcal{B}(\mathcal{O}, \mathcal{A}, \mathcal{R})$. Consider one more order relation "subconcept-superconcept" (denoted by $\leq$) that constitutes the most important structure on $\mathcal{B}(\mathcal{O}, \mathcal{A}, \mathcal{R})$.

Definition 5: Concept (X_1, S_1) is a *superconcept* of concept (X_2, S_2) if $X_1 \supseteq X_2$ which is equivalent to $S_1 \subseteq S_2$. (X_2, S_2) is then *subconcept* of (X_1, S_1).

Basic theorem: Let (O, A, R) be a context. Then $\underline{\mathcal{B}}(O, A, R) := (\mathcal{B}(\mathcal{O}, \mathcal{A}, \mathcal{R}), \leq)$ is a complete lattice[†] in which infimum and supremum can be described as follows:

$$\bigwedge_{t \in T} (X_t, S_t) = (\bigcap_{t \in T} X_t, \lambda\rho(\bigcup_{t \in T} S_t))$$
$$\bigvee_{t \in T} (X_t, S_t) = (\rho\lambda(\bigcup_{t \in T} X_t), \bigcap_{t \in T} S_t)$$

In general, a complete lattice L is isomorphic to $\underline{\mathcal{B}}(O, A, R)$ if and only if there are mappings $\gamma : O \longrightarrow L$ and $\mu : A \longrightarrow L$ such that γO is supremum-dense in L (i.e. $L = \{\bigvee X | X \subseteq \gamma O\}$), μA is infimum-dense in L (i.e. $L = \{\bigwedge X | X \subseteq \mu A\}$), and $(o, a) \in R \iff \gamma(o) \leq \mu(a)$ for all $o \in O$ and $a \in A$. In particular, $L \cong \underline{\mathcal{B}}(\text{L,L,}\leq)$.

The basic theorem shows the algebraic structure of the concept space. Let us illustrate the concept lattice with an example given by the cross-table in Table 1. This table can be understood as a description of a formal context: its objects are the eight students whose names are heading the rows (Anna, Boris, Carol, Dana,

[†]A lattice L is complete when each of its subset X has a least upper bound and a greatest lower bound in L.

Emily, Frank, Garis, Henri, denoted by A, B, C, D, E, F, G, H) and its attributes are the six hobbies which are represented by the columns (Sport, Music, Movie, Reading, Cooking, Sleeping, denoted by Sp, Mu, Mo, Re, Co, Sl); furthermore, the symbols × indicate when an object has an attribute, i.e., which student has enjoyed in which hobby. The corresponding concept lattice is represented in Figure 1.

	Sp	Mu	Mo	Re	Co	Sl
A	×	×		×		×
B		×	×	×		×
C		×	×	×		
D	×	×			×	×
E	×		×			×
F	×	×	×	×		
G		×		×	×	
H				×	×	×

Table 1. Context about hobbies of students

Formal concept analysis provides a mathematical model of conceptual knowledge which enables us to fulfill specified aims. Some efforts have been pursued to find and draw the concept lattice with a computer or to apply it in data analysis and knowledge acquisition, see [12], [13]. The number of all concepts in a real-world context, in the worst case, may be an exponential function of the number of objects or attributes. Just as a human considers only some special useful concepts, an efficient learning method in a real-world context should not carry out an exhaustive search of the whole concept lattice.

The basic idea of OSHAM is to generate a part of the concept lattice corresponding to a concept hierarchy with a high utility score. As ρ and λ are two order-reversing one-to-one operators, OSHAM tends to a tradeoff between the coverage and length of concept's intensions in order to guarantee forming sufficiently general and informative concepts, where the *coverage* $\varphi(S)$ of an attribute subset S is defined by $\varphi(S) = card(\rho(S))/card(O)$. Starting from a set of objects, OSHAM detects and organizes recursively concepts at different levels generality in the concept hierarchy. Each level of the hierarchy corresponds to a partition of the whole object set. Each concept is then clustered recursively into subconcepts with more special properties.

OSHAM is described in a main algorithm and auxiliary procedures. Starting from O and A, a hierarchy H which is empty at the beginning will be formed gradually. For simplicity, as each node in the hierarchy will be found corresponding to a closed object set generated by a closed attribute set and OSHAM is applied recursively, we identify in what follows a concept $(\rho(S), S)$ on the hierarchy with its extension $\rho(S)$ or its intension S. Each concept $C = (\rho(S), S)$ is determined by a 6-tuple *description* <name, level, {superconcepts}, {subconcepts}, S, $\rho(S)$ >.

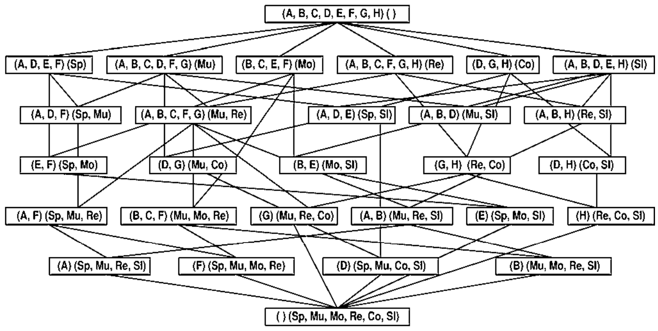

Fig. 1 Concept lattice of the formal context in Table 1

2.2 Algorithm OSHAM (O, A, H)

Input	Object set O, attribute set A, concept hierarchy H
Result	H formed recursively
Top-level	Call OSHAM (O, A, H) with all initial elements of O and A; H is empty
Variables	ω is the set of classified objects, δ is a given threshold.

1. Initially $\omega = \emptyset$.
2. Verify the splitting conditions of (O, A). **If** one of the following conditions hold:
   a. ClosedProperAttSet (O, A) = *failure*;
   b. $card(O) \leq \delta$;

   then consider (O, A) as an unsplitable concept and return.
3. Split (O, A) into subconcepts corresponding to a partition of O. To form subconcepts **repeat** the steps a–d **until** $O \backslash \omega = \emptyset$:
   a. Find a promising attribute a^* by procedure MaxCoverage (O, A, ω, a^*).
   b. Find a maximum closed attribute subset S containing a^* by procedure MaxClosedAttSet (O, A, a^*, S).
   c. Form $(\rho(S), S)$ as a subconcept of (O, A). Let $\omega = \omega \cup \rho(S)$.
   d. Form intersecting subconcepts corresponding to intersections of $\rho(S)$ with extensions of existing concepts on H excluding its superconcepts by procedure IntersectionConcept (H, S).
4. **For** each subconcept $(\rho(S), S)$ formed in the step 3, apply recursively OSHAM (O, A, H) with $O = \rho(S), A = A \setminus S$.

As OSHAM is applied recursively in forming concepts at different levels of the generality, the splitting conditions determine whether a concept is possible (condition 2.a) or worth (condition 2.b) to split further into subconcepts. The condition 2.b guarantees a consideration on concepts that cover at least some minimum number of objects. Algorithm OSHAM is refined by the following auxiliary procedures.

2.3 Auxiliary procedures

ClosedProperAttSet (O, A)
/* Determine there exist or not a closed proper subset of the attribute set A */

1. Determine a^* as attribute that satisfies $\varphi(\{a^*\}) = max_{a \in A}\varphi(\{a\})$.
2. Determine the subset of attributes $S = \{a \in A \mid \rho(\{a\}) = \rho(\{a^*\})\}$
3. **if** the coverage $\varphi(\{a^*\}) < 1$ **then** S is a proper subset of A, return *success* **else** return *failure.*

MaxCoverage (O, A, ω, a^*)
/* Find attribute a^* so that $\omega \cup \rho(\{a^*\})$ is the largest cover of O by the same way described in procedure ClosedProperAttSet (O, A). If many attributes a satisfy this condition, choose a^* arbitrarily that minimizes $card(\omega \cap \rho(\{a\}))$ (the minimum intersection condition) */

MaxClosedAttSet (O, A, a^*, S)
/* Find the closed attribute subset containing a given attribute a^* */
Let S = $\{a^*\}$.
for every $a \in A \setminus \{a^*\}$ **do**
 if $\rho(\{a\}) = \rho(\{a^*\})$ **then** $S = S \cup \{a\}$.

IntersectionConcept (H, S)
/* Form intersecting concepts from a given concept $(\rho(S), S)$ */

1. **for** every existing concept $(\rho(S'), S')$ on H, excluding superconcepts of $(\rho(S), S)$, **if** $\rho(S) \cap \rho(S') \neq \emptyset$ **then** create the intersecting concept $(\rho(S''), S'')$. The extension $\rho(S'')$ is the intersection of the extensions of two constituent concepts, $\rho(S'') = \rho(S) \cap \rho(S')$. The intension S'' is the closed attribute set found by procedure MaxCLosedAttSet (O, A, a^*, S) where a^* is one attribute chosen arbitrarily from $\lambda\rho(S'')$.
2. Apply recursively IntersectionConcept (H, S'').

2.4 Correctness of the algorithm

In this subsection we will prove some properties which shows that OSHAM forms correctly concepts according to the definitions in 2.1. First, we remark that each time when OSHAM is applied to an existing concept it generates subconcepts of this concept, and the procedure IntersectionConcept ensures that each level of the hierarchy corresponds to a partition of all initial objects of O. As OSHAM is based on its auxiliary procedures, we need to show that these procedures are correct.

Proposition 1: Procedure ClosedProperAttSet determines a closed attribute set.

Proof: Since $A \neq \emptyset$, so there exists $a^* \in A : \alpha(\{a^*\}) = max_{a \in A}\alpha(\{a\})$. Thus, according to procedure ClosedProperAttSet, $S = \{a \in A \mid \rho(\{a\}) = \rho(\{a^*\})\} \neq \emptyset$ since $a^* \in S$. We need to show that S is closed. It is proved in formal concept analysis that an attribute set is closed if and only if it corresponds to a maximal rectangle in the object-attribute matrix. The set S generated by procedure ClosedProperAttSet contains all attributes a covering the same set $\rho(\{a^*\})$, and as a^* covers a maximum number of objects, it is clear that S corresponds to a maximal rectangle in the object-attribute matrix. □

Proposition 2: The attribute set S determined by procedure MaxCoverage and MaxClosedAttSet is closed.

Proof: This follows directly from the description of these procedures as they employ the same way of finding a^* described in ClosedProperAttSet. □

Proposition 3: Procedure IntersectionConcept forms correctly concepts.

Proof: There exists at least one $a^* \in \lambda\rho(S'')$ because $\lambda\rho(S'') = \lambda(\rho(S) \cap \rho(S')) \supseteq S \cup S'$ as each object belonging to $\rho(S) \cap \rho(S')$ has all attributes of S and S'. The procedure MaxClosedAttSet ensures to find, from $a^* \in \lambda\rho(S'')$, the maximal rectangle in the object-attribute matrix. Thus, S'' is a closed attribute set corresponding to $\rho(S) \cap \rho(S')$. □

In formal concept analysis, when we construct a concept lattice from a context, the number of concepts can be exponential in the size of the context, and the complexity of finding concepts is exponential, since the number of concepts may be large. More precisely, the upper bound of complexity of determining all concepts of a given context (O, A, R) is $(|O||A|^2)$ times the number of concepts. The complexity of OSHAM is $\mathcal{O}(|O||A|)$, and the concept hierarchy is constructed by OSHAM in linear time in the number of objects and the total number of attributes.

3. Experimental results

OSHAM version 1.0 is written in ANSI C and is capable of running on a wide range of platforms. It has been tested using a number of databases, particularly those from the UC Irvine Repository of Machine Learning, including the Wisconsin breast cancer, Lung cancer, DNA prometer gene sequences, large soybean disease, and mushroom. Results from two of them are given below. The experiments have been designed as follows:

- divide the data set randomly into training and testing data;
- run OSHAM on the training data with different parameters without using the class information in the concept formation process; only use the class information to name the generated classes;
- match testing data set with the concept hierarchy and evaluate the prediction accuracy.

3.1 Wisconsin Breast Cancer Data

This data set contains 699 examples of patients with breast cancer (as of 15 July 1992) represented by 9 integer-valued attributes, respectively are *Clump Thickness, Uniformity of Cell Size, Uniformity of Cell Shape, Marginal Adhesion, Single Epithelial Cell Size, Bare Nuclei, Bland Chromatin, Normal Nucleoli, Mitoses.* Each instance lies in one of two possible classes, benign or malignant, with the class distribution being benign: 458 (65.5%) and malignant: 241 (34.5%).

559 instances are randomly chosen for the training set and the remaining 140 are used for testing. OSHAM is tested over 40 runs with different values of parameters on the training set. Following are some generated concepts extracted from the text output of OSHAM:

```
CONCEPT 42
Level 5
Superconcepts = {40}
Subconcepts = {47 48 49}
Defining_Feature = {(4,2)(8,1)(5,1)(7,1)(3,1)}
Unclassified_Instances = {318 428 457 466}

CONCEPT 48
Level 6
Superconcepts = {42}
Subconcepts = {53 54 55 56}
Defining_Feature = {(6,2)(4,2)(8,1)(5,1)(7,1)(3,1)}
Unclassified_Instances = {86 270 334 425 430 449}
```

The defining features, for example $\{(4,2)\ (8,1)\ (5,1)\ (7,1)\ (3,1)\}$ are understood as *Marginal Adhesion* = *value2* $\wedge$ *Normal Nucleoli* = *value1* $\wedge$ *Single Epithelial Cell Size* = *value1* $\wedge$ *Bland Chromatin* = *value1* $\wedge$ *Uniformity of Cell Shape* = *value1*. The average percentage of successful matched unknown instances in the testing data is 94.3%.

3.2 Mushroom data

OSHAM has been tested in a domain of mushroom. This data set includes 8124 instances each was described along 23 attributes all have been nominalized. Our experiments consisted of presenting a sequence of 1400 randomly chosen instances as training set, and the rest as testing set. Mushrooms are classified *a priori* into two poisonous and edible categories. The data is collected with missing values.

C1	C2	C3	C4	C5	C6	
234	753	1555	403	1309	1401	245
0.79	0.88	0.07	1.0	0.05	1.0	0.73
0.21	0.12	0.93	0.0	0.95	0.0	0.26

Table 2. Prediction distribution over testing data

The concept hierarchy at the first level consists of six superconcepts, two of which are mainly poisonous (C3, C5) and four are mainly edible (C1, C2, C4 and C6). Table 2 shows the prediction rate distribution over testing data. The first line of the table indicates the number of instances recognized as members of these concepts. These superconcepts are refined at lower levels of the hierarchy. The second and the third lines stand for the edible and poisonous probabilities estimated for these concepts. There are 245 instances which are not completely matched by the concept hierarchy. A flexible matching procedure is studying in order to give a prediction for these cases. The number of false negative and false positive cases are 166 and 119, respectively. The correct classification rate over the testing set is 95.2% which is approximately the best results reported by other methods in machine learning literature for the same data set.

4. Concluding Remarks

We have presented a concept formation method based on formal concept analysis. Formal concept analysis mathematically formulates the traditional understanding of a concept as a unit of thoughts consisting of two parts: the extension and the intension. It offers a natural tool for concept formation tasks with a strong algebraic structure of complete lattices. OSHAM differs mainly from the work of Wille on formal concept analysis in not carrying out an exhaustive search but generating a part of the concept lattice in a hierarchical form by a non-exhaustive manner. Once a level of the hierarchy is formed, the procedure is repeated recursively for each existing concept. OSHAM has been implemented and experimented. It performs reasonably well on various public data sets in machine learning. By the nature of the concept representation in formal concept analysis, OSHAM is effective even with sparse object-attribute distributions.

If compared to the related work, OSHAM has the following advantages:

- Concepts are formed with high utility score. Though the approach is different, OSHAM obtains, in terms of probability, the high utility score as other methods using probabilistic concepts, for example Fisher's COBWEB [3], [5]. COBWEB forms concepts with a high within-class similarity, this means the probabilities of attribute values for a given concept, $P(A_i = V_{ij}|C_k)$ are high. OSHAM forms concepts in a way such that all objects of each concept share the same set of attribute values with highest probabilities $P(A_i = V_{ij}|C_k)$.

- Many concept formation systems, e.g., COBWEB, are sensitive to the order of instance representation. One advantage of OSHAM is that it is completely independent with this order of instance representation.

- Concepts formed by OSHAM have a clear semantics and high coherence. Any concept in the hierarchy is a subconcept of some concepts with more properties (more specific), and is a superconcept of some other concepts with less properties (more general). OSHAM concepts naturally satisfy the condition of *vertically well placed* concepts in ARACHNE [8]. In a on-going work, by adding constraints on the dissimilarity between classes of objects, the OSHAM concepts are additionally modified to *horizontally well placed*, and as a result they become *well organized.*

As OSHAM is under elaboration, it may be possible to improve it in different directions:

- More evaluation of experts on constructed concept hierarchies and automatic parameter selection in order to quickly obtain acceptable results.

- Extension of the method to more complex formalisms, in particular to the predicate logic.

- OSHAM forms disjoint concepts as do most concept learning systems. By skipping the IntersectionConcept operator, OSHAM can be modified to form effectively overlapping concepts which are important in some domains.

- Extension of OSHAM into a hybrid system by combining it with some features of probabilistic, exemplar views and the similarity principle of categorization. In this on-going work, OSHAM is able to deal with exceptional cases and unmatched instances.

- OSHAM is a nonincremental concept learning method that accepts only symbolic attributes. It is necessary to extend it for numeric attributes and an incremental learning mode.

Acknowledgement

The author would like to thank Professor S. Ohsuga (University of Tokyo), Professor M. Kimura (JAIST-Hokuriku) for giving him the opportunity to pursue this research, and Professor H. Shimodaira (JAIST-Hokuriku) for helpful comments. This work is supported by Kokusai Electric Co., Ltd. and Japan Advanced Institute of Science and Technology-Hokuriku.

References

[1] Cheeseman P., Kelly J., Self M., Stutz J., Taylor W., Freeman D., "AUTOCLASS: A Bayesian classification system", *Proceedings of the Fifth International Conference on Machine Learning*, Morgan Kaufmann, 1988, pp. 54–64.

[2] Davey B.A., Priestley H.A., *Introduction to Lattices and Order*, Cambridge University Press, 1990.

[3] Fisher, D., "Knowledge Acquisition Via Incremental Conceptual Clustering", *Machine Learning*, Vol. 2 (1987), pp. 139–172.

[4] Fisher, D., Pazzani M., "Computer models of concept learning", in D. Fisher, M. Pazzani, P. Langley (Eds.) *Concept Formation: Knowledge and Experience in Unsupervised Learning*, Morgan Kaufmann 1991, pp. 3-44.

[5] Gennari, J.H., Langley, P., Fisher, D., "Models of Incremental Concept Formation", *Artificial Intelligence*, 40(1989), pp. 11–61.

[6] Ho, T.B., "A practical approach to the construction of hierarchical objects knowledge bases", *Proceedings of the Third Pacific Rim International Conference on Artificial Intelligence PRICAI'94*, Beijing, August 1994, pp. 311–316.

[7] Lebowitz M., "Experiments with incremental concept formation: UNIMEM", *Machine Learning*, Vol. 2 (1987), pp. 103-138.

[8] McKusick, K.B., Langley, P., "Constraint on Tree Structure in Concept Formation", *Proceedings of International Joint Conference on Artificial Intelligence*, IJCAI'91 (1991), pp. 810–816.

[9] Michalski R.S., Stepp R.E., "Learning from observation: conceptual learning", in R.S. Michalski, J.G. Carbonell, T.M. Michell (Eds.), *Machine Learning: An Artificial Intelligence Approach*, Morgan Kaufmann, 1983.

[10] Mitchell T., "Generalization as search", *Artificial Intelligence*, 18 (1982), pp. 203–226.

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

[12] Wille R., "Concept lattice and conceptual knowledge systems", *Computers and Mathematics with Applications*, Vol. 23 (1992), pp. 493–515.

[13] Wille R., "Lattice in data analysis: how to draw them with a computer", in *Algorithms and order*, NATO ASI Series 147, Reidel, 1989. pp. 33-58.

Tu Bao Ho is currently a visiting associate professor at the School of Information Science, Japan Advanced Institute of Science and Technology–Hokuriku. He received a B.Tech. degree in Computer Science and Applied Mathematics from the Hanoi Institute of Technology, in 1978, and M.S. and Ph.D. degrees from Université Pierre et Marie Curie, Paris, in 1984 and 1987. In 1979, he joined the Institute of Computer Science and Control, National Centre for Scientific Research of Vietnam where he is now a senior researcher and associate professor. His research interests include machine learning, knowledge-based systems, knowledge and data engineering.