Context-Sensitive, Distributed, Variable-Representation Category Formation*

Mirsad Hadzikadic and Paul Elia**

Department of Computer Science
University of North Carolina
Charlotte, NC 28223

Abstract

This paper describes INC2, an incremental category formation system which implements the concepts of family resemblance, contrast-model-based similarity, and context-sensitive, distributed probabilistic representation. The system is evaluated in terms of both the structure of categories/hierarchies it generates and its categorization (prediction) accuracy in both noise-free and noisy domains. Performance is shown to be comparable to both humans and existing learning-from-example systems, even though the system is not provided with any category membership information during the category formation stage.

Introduction

The issues of category formation and categorization represent an important research topic due to the fact that categories lie at the core of our thought, perception, speech, and action. Researchers from several diverse disciplines (psychology, philosophy, linguistics, anthropology, and computer science) actively work in this area. Computer science, in particular, offers wealth of results under the common term of *concept formation*.

Fisher & Langley (in press) describe concept formation as the incremental unsupervised acquisition of categories. A system which can accomplish this task can be used both as an aid in organizing and summarizing complex data and as a retrieval system which can predict properties of previously unseen objects. Such a system will be useful in domains where knowledge is incomplete or classifications and/or human experts do not exist.

This paper describes INC2, an incremental concept formation system which combines the effectiveness of similarity-based learning methods (computer science) with the plausibility of the modified contrast-model (psychology) and family-resemblance (philosophy) theories. The system uses a context-sensitive, distributed probabilistic representation to store the knowledge about categories, their descriptions, and members. The system has been evaluated in the domains of soybean disease, breast cancer, and primary tumor cases. When compared to humans and existing learning-from-examples systems, INC2 shows comparable performance in terms of prediction accuracy.

* This work was supported by a grant from the UNCC, College of Engineering.

** Currently with the Energy Management Associates, Inc, Atlanta, Georgia.

INC2

INC2 shares the object/category representation formalism with its predecessor, the INC system (Hadzikadic & Yun 1989). Also, they both incrementally build a hierarchy (tree) of disjoint categories (although an object may match descriptions of more than one category) in an unsupervised fashion. That, however, is where the similarities end. We will now concentrate on the description of the INC2 system.

Representation

As already mentioned, INC2 builds a hierarchy of non-disjoint category descriptions. The leaves of the hierarchy are objects (singleton categories). The root of this hierarchy has associated with it a description which is a summary of all descriptions of the objects seen by the system to date. As one traverses the hierarchy, downward pointers lead to nodes with more specific descriptions, while upward pointers lead to nodes with more general descriptions.

A description of each category C is defined as a set of features f (attribute-value pairs). Each feature has a conditional probability $p(f \mid C)$ associated with it. Thus, representing the color feature of red apples would take the form (*color red* 0.25). The 0.25 means that members of this category are red 25% of the time. Consequently, singleton categories will have probabilities equal to 1.0. In addition to nominal attributes, INC2 supports the structural ones as well. Representing a structural fact such as the fact that object a is inside object b would take the form (*contains* (b a) 1.0). Finally, INC2 supports structured domains, i.e., (*shape triangle* 1.0) will match (*shape square* 1.0), provided the knowledge that both triangles and squares are specializations of polygons.

The notion of *probabilistic concept representations* was introduced by Smith & Medin (1981). However, since members of a given category may reside in distinct portions of the hierarchy, the adopted representation formalism is referred to as *distributed probabilistic concept hierarchies* (Fisher & Langley in press).

Family Resemblance

In addition to its features and hierarchical pointers, each category description contains an estimate of its cohesiveness given in the form of *family resemblance*. Family resemblance, first advanced by Wittgenstein (1953), is defined here as the average similarity between all possible pairs of objects in a given category. More formally, we define the family resemblance FR of a given category, C, to be

$$FR(C) = \frac{\sum_{a,b} [s(a,b) + s(b,a)]}{2 \times \binom{n}{2}}$$

where $a \neq b$, a and b are members of C, $s(a,b)$ is any similarity function defined for two objects described with feature sets, n is the number of children of the node associated with the category C, and $\binom{n}{2}$ is the number of distinct two-element sets of objects in C.

We interpret family resemblance as a measure of the cohesiveness of a category. To save processing time, INC2 compares pairs from the children of C rather than from all the objects stored in the subtree headed by C to approximate the family resemblance for a given category, C. A special case arises for categories which do not have any pairs, namely singleton categories. The value we used for the family resemblance of a singleton category is the family resemblance of its least compact sibling. If there are no siblings, the family resemblance of the parent is used instead. In essence, this means that two objects will form a new category if their similarity is greater than either the average similarity associated with the least compact sibling or the average similarity between all pairs of objects within the given context (assuming that no other category represents a better host for the object).

Similarity Function

The similarity function s used by INC2 represents a variation of the contrast model (Tversky 1977) which defines the similarity between an object and a category as a linear combination of both common and distinctive features. Our modification of the contrast model includes: (a) elimination of a referent since comparisons take place between nodes at the same level of the hierarchy, (b) consequent introduction of the symmetricity property, thus reducing the number of comparisons needed by 50%, and (c) normalization of the function so that the values fall into the (-1.0, 1.0) range, with 1.0 denoting identical objects/categories and -1.0 indicating completely dissimilar ones. The new function is now formally defined as

$$s(A,B) = \frac{\lambda - \alpha - \beta}{\lambda + \alpha + \beta}$$

$$\lambda = \frac{m_A \times \sum_{f_{AB}} p(f_{AB} \mid A) + m_B \times \sum_{f_{AB}} p(f_{AB} \mid B)}{m_A + m_B}$$

$$\alpha = m_A \times \sum_{f_A} p(f_A \mid A)$$

$$\beta = m_B \times \sum_{f_B} p(f_B \mid B)$$

where A and B are (possibly singleton) categories, λ represents the contribution of the common features, while α and β introduce the influence of the features of A not shared by B and vice versa, respectively, m_A is a number of objects stored under the node associated with the category A, m_B is similarly interpreted for the category B, f_{AB} is the set of features shared by A and B, f_A is the set of features present in the description of A but not B, and f_B is the set of features present in the description of B but not A.

Operators

INC2 uses four operators to guide the category formation process: create, extend, merge, and delete. *Create* forms a new category for an object found to be dissimilar to all examined categories, while *extend* adds a new object to the most similar category found.

Merge unites two or more categories at the same level in the hierarchy that are found to be similar to a new object to form a new category. The object is then recursively classified with respect to the category which maximizes the increase in its cohesiveness upon incorporating the object.

When the family resemblance of a given category is less than the family resemblance of its parent, that category does not represent a proper specialization of its parent. Such a situation is likely to occur in noisy domains upon application of the merge operator. The *delete* operator rectifies this problem by removing the category and promoting its specializations.

Algorithm

Figure 1 presents the classification procedure of INC2. It implements a hill-climbing strategy which encourages advancement toward the maximal improvement of the hierarchy as measured by the increase in the family resemblance of every candidate host category.

The algorithm can be paraphrased as follows. To begin, pass to the procedure both an object, a, and the root of the hierarchy, C. The first action is to update the description of the category C based on the description of the object a. Next, find the change in the family resemblance measures (ΔFR) that would result from temporarily placing a in each of $C's$ specializations (subcategories).

If only one subcategory experiences improvement (an increase in its family resemblance measure), then either extend this subcategory (if singleton) or call the classification procedure recursively on a and that subcategory, the *besthost*. If two or more subcategories experience improvement, then find the subcategories which are at least as similar to a as the family resemblance of C, i.e., those subcategories which are more similar to a than the degree of compactness in the given context. These categories are merged together to form a new category, with a continuing the classification process recursively with respect to the new best host (maximizing the increase in its family resemblance). If no subcategory experiences improvement, then a is unique and a new singleton category is created.

Once a home for a is found and all category descriptions affected have been updated, the system updates the family resemblance measures of the categories on the path from a to the root.

Classify(a,C)
Update the description of the category C using a.
Compute ΔFR for each of $C's$ subcategories.
IF a single subcategory (*besthost*) has a positive ΔFR
THEN

IF the best host is singleton

THEN call **Extend(a, best host)**

ELSE call **Classify(a, best host)**

ELSE

IF many subcategories have a positive ΔFR
THEN

(a) find the subcategories which are at least as similar to a as the FR of the parent category

(b) call **Merge(a, similar subcategories)**

(c) compute the description of the new category

(d) determine the subcategory *(best host)* that maximizes ΔFR

(e) call **Classify(a, best host)**

ELSE

(no subcategory has a positive ΔFR)
call **Create(a)** (create a new singleton category for a)

Update the FRs of the categories on the path from a to the root.
IF a merge was done
THEN

search that subtree for any class with a FR less than the FR of its parent, **deleting** if found, promoting its children one level higher, and recomputing the FR of the parent of the deleted category.

Figure 1: The classification algorithm.

Figure 2 presents the retrieval procedure of INC2. We begin by passing the procedure an object a and the root of the hierarchy. The next step is to compute the similarity between a and each of $C's$ subcategories. Then, the procedure is called recursively with the subcategory that maximizes the similarity function. The process stops after reaching a singleton category. Note that the retrieval procedure utilizes the similarity function rather than the concept of family resemblance since the latter is needed mainly for plausible clustering of objects.

Retrieve(a,C)
IF C is a singleton category
THEN return C
ELSE

(a) compute the similarity between a and all the subcategories of C

(b) find the subcategory (best candidate) that maximizes the similarity

(c) call **Retrieve(a, best candidate)**

Figure 2: The retrieval algorithm.

Drop Threshold

Although it is not specified in the algorithms, both the classification and retrieval procedures rely on the *drop threshold*. This threshold allows for category descriptions to be either probabilistic or logical. It can be set between 0.0 and 1.0 and means that any feature which falls below this threshold in conditional probability should be dropped from the description of the given category. A value of 1.0 for this threshold would yield a logical category description.

However, the nature of the classification process calls for a dynamically adjustable threshold rather than a fixed

one. For example, at the top level of the hierarchy all features are important no matter how low their probabilities might be, due to the potential noise in object descriptions as well as the diversity of objects in the domain. Therefore, the drop threshold should be set close to 0.0. At the lower levels of the hierarchy, however, certain patterns have been detected, resulting in high conditional probabilities for 'participating features' and, consequently, lower probabilities for the ones not significantly present in those patterns. However, since all categories at the lower levels have few members, all the features found in their descriptions will have relatively high conditional probabilities (1 out of 2 still gains the probability of 0.5). To avoid the interference of unimportant features with the retrieval process, the drop threshold should be set close to 1.0. The intermediate categories will, then, require the drop threshold somewhere between 0.0 and 1.0, depending on the level of the hierarchy (the lower the level, the higher the drop threshold).

In order to alleviate this problem, we rely on family resemblance to provide an estimate of the drop threshold. As we stated earlier, family resemblance can be interpreted as an estimate of the category compactness. It is naturally close to 0.0 at the root (summarizing the whole universe) and to 1.0 at the leaves. Therefore, during both classification and retrieval INC2 sets the drop threshold to the value of the family resemblance of the parent category. It increases with the object traversing the hierarchy downward.

This dynamically adjustable drop threshold and the fact that the expand, merge, and delete operators depend on the family resemblance of the parent category (as well as the candidate subcategory's siblings in the case of merge) represent two important features introduced by INC2. As a result, INC2 performs a context-sensitive classification/retrieval due to its adaptive behavior that changes from level to level of the hierarchy. In that process, consequently, INC2 uses different representations to describe objects/categories at different levels of the hierarchy, possibly moving from the probabilistic representation (drop threshold = 0.0) at the top level to the logical one (drop threshold = 1.0) at the leaves.

Performance Evaluation

The performance of INC2 was evaluated in the domains of soybean disease, breast cancer, and primary tumor cases. Although a true expert in these domains would have access to much richer data, we show that the knowledge representation and algorithm used by INC2 yield a useful hierarchy of categories. The soybean disease domain was selected as a representative of noise-free domains, while the breast cancer and primary tumor cases included a lot of data with incorrect or missing values (a noisy domain). In each experiment, the information concerning the ideal category for a given object was not given to the system and training sets were randomly both selected and ordered.

The soybean disease domain consisted of forty-seven cases with four ideal categories represented. A high number of features used to describe the cases were common to all forty-seven cases, making the domain very compact. Training sets of sizes five to twenty-five in increments of five were selected randomly from the domain. For assessing prediction accuracy (determining category-membership for previously unseen objects), twenty cases were randomly selected from the set of remaining cases. Table 1 shows the range, mean, and sample standard deviation prediction accuracies for five experiments per training set size.

Training Size (#)	Range (%)	Mean (%)	Sample Std. Deviation
5	75-95	82	10.4
10	95-100	98	2.7
15	95-100	99	2.2
20	95-100	99	2.2
25	95-100	99	2.2

Table 1: INC2's prediction accuracies for five experiments per training set size in the soybean disease domain.

With four ideal categories in the soybean disease domain, there is a 25% chance of simply guessing the correct diagnosis. INC2's prediction accuracy was consistently above chance, even at low levels of experience. From the results summarized in the table 1 it is obvious that INC2's performance improves with experience and that it needs a small portion of objects from a domain to make a plausible decision with respect to category membership of previously unseen objects.

The breast cancer domain[1] (prediction of cancer recurrence five years later) consisted of 286 cases with two ideal categories represented, *yes* and *no*. The domain itself can be characterized as very noisy. A total of nine features were randomly missing from the 286 cases, with no more than two features missing from any one case. There were cases from two different categories bearing the same exact description. Five specialists[2] were presented this data and then tested for diagnostic accuracy. They were correct 64% of the time.

[1] Data from this domain were provided by the Institute of Oncology of the University Medical Center in Ljubljana, Yugoslavia.
[2] The specialists were from the Institute of Oncology, Ljubljana.

The other oncology-related example, the primary tumor domain[1] (prediction of tumor locations), consisted of 339 cases with 20 ideal categories represented. A total of 224 features were randomly missing from the 286 cases. Once again, there were cases from two different categories having the same exact description. Four internists and four specialists[2] were tested for diagnostic accuracy. Internists were correct 32% of the time, specialists 42%.

Since the results obtained in the soybean disease domain suggest that 25% of the total number of objects in the domain is sufficient for estimating a maximum achievable performance by the system, we have decided to use 25% cases for training and the remaining 75% cases for prediction. This is in sharp contrast with other learning systems which used 70% of the objects for training and the remaining 30% for prediction.

Table 2 presents the prediction accuracies for two learning-from-examples systems (AQ15 [Michalski et al 1986] and Assistant-86 [Cestnik et al 1987]), human experts, and INC2. Note that in a system which learns from examples, training cases are associated with the correct response, and the goal of the system at that point is to find a set of rules which will cover that data and be useful for classifying previously unseen objects.

System	Breast Cancer (%)	Primary Tumor (%)
AQ15	66	39
Assistant-86	78	44
INC2	69.2	30
Human Experts	64	42

Table 2: Mean prediction accuracies for the oncology domains. The INC2's accuracy is obtained by averaging values achieved in five runs on randomly chosen sets of objects.

There is a 50% chance of simply guessing the correct diagnosis for the breast cancer domain and 5% for the primary tumor domain. INC2's prediction accuracy was shown to be significantly above chance. Furthermore, INC2 compares to both human experts and learning-from-examples systems given the same task (significantly better in the breast cancer domain than in the primary tumor domain). This is despite the fact that INC2 is never given any help from a teacher.

In order to evaluate the effectiveness of the fixed-value drop threshold compared to its variable, context-sensitive alternative, we have carried out a sequence of experiments in both soybean and breast cancer domains. In the soybean-disease experiments we fixed the size of the training set to ten (25% of the complete set), while varying the drop threshold from 0.0 to 1.0 with the increment of 0.25. The results are summarized in the table 3.

Drop Threshold	Range (%)	Mean (%)
0.00	90-100	98
0.25	100-100	100
0.50	95-100	98
0.75	90-100	98
1.00	95-100	99

Table 3: Prediction accuracies for the fixed-value drop threshold in the soybean disease domain.

The results are obviously equal or slightly better than that of the variable drop threshold. That is especially true for the value of 0.25 with its perfect score. However, when we applied INC2 with the drop threshold = 0.25 to the breast cancer domain its prediction accuracy dropped from 69.2% (for the variable drop threshold) down to 63.3%. To make sure that this was no accident we checked the basic case (drop threshold = 0.0), where all the features are taken into consideration at all times, and recorded a similarly reduced performance, 63.6%. It seems that the variable, context-sensitive drop threshold provides a more robust alternative for noisy domains while retaining a good performance in the noise-free ones.

Finally, in order to estimate the effect of ordering of objects on the resulting classification, we evaluated prediction accuracies for five different random orderings of the same set of objects. Table 4 presents the obtained results.

Ordering (#)	Accuracy (%)
1	71.0
2	64.5
3	61.7
4	64.5
5	68.2

Table 4: Prediction accuracies for five random orderings of the set of input breast cancer cases.

If we compare the sample standard deviation for the data presented in the table 4 (3.63) and for the data used to derive the value reported for the breast cancer domain for INC2 (4.53; Table 2), a decrease of 20%, then we can conclude that INC2 represents a relatively robust incremental category formation system with respect to

the ordering of input objects, which has been a major drawback of all incremental systems. Furthermore, we suggest that the performance of different hierarchies rather than their form should be measured when evaluating the effect of object orderings on the system's performance.

Previous Work

Most existing category formation systems use hill-climbing methods to find suboptimal clusterings of objects to be characterized and create nondisjoint category descriptions. Five existing systems which share these features are COBWEB (Fisher 1987), CLASSIT (Gennari, Langley, & Fisher 1989), UNIMEM (Lebowitz 1987), CYRUS (Kolodner 1984), and WITT (Hanson & Bauer 1989).

Out of the aforementioned incremental concept formation systems, INC2 is most similar to COBWEB. They utilize similar distributed probabilistic concept representation formalism and operators. However, INC2 possesses several features that significantly distinguish it not only from COBWEB but from all other systems as well:

I. Family resemblance provides an estimate of category compactness, which is used by INC2 to: (a) introduce an objective, domain-adaptive, context-sensitive bias for classification; (b) implement a variable, context-sensitive representation of concepts, thus focusing only on important features in the current context; (c) measure an improvement over a single category in addition to the improvement over the whole level of the hierarchy, thus utilizing finer-grained information to guide classification; and (d) achieve a relative robustness with respect to object orderings.

II. INC2 seeks tree-structures that optimize not only the top level of the hierarchy, but the hierarchy as a whole.

III. It introduces no constraints on input objects (they may be hierarchies of objects on their own).

Conclusion

INC2 incrementally builds a hierarchy of category descriptions based on a set of objects described with nominal and/or structural attributes. The system is based on our own interpretation of both family resemblance and contrast model theories. It uses a context-sensitive threshold to eliminate all irrelevant features from concept descriptions, thus effectively introducing an adaptive, context-dependent representation of concepts. Performance has been shown to be comparable to both human experts and learning-from-example systems.

Future research will involve three directions: (a) reducing the number of steps required to compute family resemblance for categories with many children, (b) enhancing the representation with continuous (linear) attributes, and (c) performing further experiments in order to objectively evaluate the INC2's results in the light of both fan and typicality effects as well as the structure of categories.

References

Cestnik, G. et al. 1987. Assistant-86: A Knowledge-Elicitation Tool for Sophisticated Users. *Progress in Machine Learning*, 31-45, Sigma Press.

Fisher, D. (1987). Knowledge Acquisition Via Incremental Conceptual Clustering. *Machine Learning* 2(2):139-172.

Fisher, D. and Langley, P. In Press. The Structure and Formation of Natural Categories. *The Psychology of Learning and Motivation*, G. Bower (Ed.).

Gennari, J. H., Langley, P., and Fisher D. 1989. Models of Incremental Concept Formation. *Artificial Intelligence* 4(1-3):11-61.

Hadzikadic, M. and Yun, D. Y. Y. 1989. Concept Formation by Incremental Conceptual Clustering. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, August 20-26, Detroit, Michigan.

Hanson, S. J. and Bauer, M. 1989. Conceptual Clustering, Categorization, and Polymorphy. *Machine Learning* 3(4):343-372.

Kolodner, J. L. 1984. *Retrieval and Organizational Strategies in Conceptual Memory: A Computer Model.* Lawrence Erlbaum Associates, Publishers, London.

Langley, P., Gennari, J. H., and Iba, W. 1987. Hill-Climbing Theories of Learning. In *Proceedings of the Fourth International Workshop on Machine Learning*, 312-323, University of California, Irvine.

Lebowitz, M. 1987. Experiments with Incremental Concept Formation: UNIMEM. *Machine Learning* 2(2):103-138.

Michalski, R. S. et al. 1986. The AQ15 Inductive Learning System: An Overview and Experiments. Technical Report, Intelligent Systems Group, University of Illinois at Urbana-Champaign.

Smith, E. E. and Medin, D. L. 1981. *Categories and Concepts.* Harvard University Press, Cambridge, MA.

Tversky, A. 1977. Features of Similarity. *Psychological Review* 84:327-352

Wittgenstein, L. 1953. *Philosophical Investigations.* MacMillan, New York.