LEARNING RELATIVE ATTRIBUTE WEIGHTS FOR INSTANCE-BASED CONCEPT DESCRIPTIONS

David W. Aha Dale M. McNulty
Department of Information & Computer Science
University of California, Irvine, CA 92717 U.S.A.
(714) 856-8779 (714) 631-3021
aha@ics.uci.edu mcnulty@ics.uci.edu

ABSTRACT

Nosofsky recently described an elegant instance-based model (GCM) for concept learning that defined similarity (partly) in terms of a set of attribute weights. He showed that, when given the proper parameter settings, the GCM model closely fit his human subject data on classification performance. However, no algorithm was described for learning the attribute weights. The central thesis of the GCM model is that subjects distribute their attention among attributes to optimize their classification and learning performance. In this paper, we introduce two comprehensive process models based on the GCM. Our first model is simply an extension of the GCM that learns relative attribute weights. The GCM's learning and representational capabilities are limited – concept descriptions are assumed to be disjoint and exhaustive. Therefore, our second model is a further extension that learns a unique set of attribute weights for each concept description. Our empirical evidence indicates that this extension outperforms the simple GCM process model when the domain includes overlapping concept descriptions with conflicting attribute relevancies.

Keywords: concept learning, concept-dependent attribute weights, instance-based concept descriptions, Generalized Context Model, categorization

1. MOTIVATION

Research on supervised learning algorithms and categorization models share the goal of designing an intelligent concept learning model and associated concept description representation. Of course, the concentrations in the two disciplines differ (i.e., computational tractability and psychological plausibility respectively). We believe that these two perspectives are complementary and mutually beneficial. For example, supervised learning strategies can be used to support processing components for models of categorization.

In this paper we introduce two process models for categorization, each of which is an extension of Nosofsky's (1986, 1987) Generalized Context Model (GCM). Nosofsky posited that an attention-focusing process learns the GCM's attribute weights, but no such algorithm was described. Our first model, GCM-SW (Single set of Weights), is a principled process model that learns a single set of attribute weights for the GCM (but not the GCM's scale or concept bias parameters). Our second model, GCM-MW (Multiple sets of Weights), combines separate progress described in the machine learning and categorization theory literatures. GCM-MW learns a separate set of attribute weights and concept description for each concept (Aha, 1989) in Nosofsky's GCM.

Nosofsky (1986; 1987) showed that the GCM model closely fit human subject data on several simple concept learning tasks. However, its performance degrades when the concepts being learned require conflicting attribute settings to optimize categorization of

their instances. Therefore, our contribution is twofold. First, we describe an algorithm for learning the GCM's attribute weights. Second, we show that, by using separate sets of attribute weights, the process model learns more quickly and accurately during complex concept learning tasks.

GCM-SW and GCM-MW are examples of *instance-based learning* (IBL) algorithms. We introduce the methodology and framework for IBL algorithms in the following section.

2. INSTANCE-BASED LEARNING AND THE PROCESS FRAMEWORK

IBL algorithms process a sequence of *training instances* and output a set of *concept descriptions* whose accuracy can be evaluated by a disjoint set of *test instances*. IBL algorithms process instances incrementally; concept descriptions are updated after each classification attempt. Each instance is represented with a set of n attribute-value pairs. Instances represent unique points in an *instance space*, where each attribute represents one dimension in the space.[1] Concept descriptions are updated after each instance is classified. These descriptions map instances to an interpretation of the instance space called the *psychological space* (Shepard, 1987). *Concepts* are unions of regions in psychological space.

IBL algorithms represent each concept description with a set of instances rather than with abstractions derived from them (i.e., rules, decision trees). The extension of the concept description in the psychological space is made with respect to a similarity function and a classification function. IBL algorithms neither modify nor discard informative instances – they are simply added to a concept description. Therefore, IBL algorithms have low learning (updating) costs and retain the concept-describing information present in specific instances.

IBL algorithms are specified by a framework consisting of three components.

1. First, a *similarity function* computes the continuously-valued *similarity* of a training instance i to an instance in a concept description.
2. Next, a *classification function* inputs the similarity function's results on all concept description instances and yields a probabilistic classification of i for that concept.
3. Finally, a *concept description updater* is used to maintain summary information, such as attribute weight settings. Inputs include i, the similarity results, the classification results, and the concept descriptions. It yields the modified concept descriptions.

These components are sufficient to support the application and acquisition of concept descriptions. This framework specifies a spectrum of IBL models, obtained by varying these components. Our models will be described with respect to this framework.

3. LEARNING THE GCM'S ATTRIBUTE WEIGHTS

This section introduces a process model interpretation of Nosofsky's GCM (1986, 1987). This model is described using the framework introduced in Section 2.

[1] In this paper, we restrict attribute values to be either Boolean, nominal, or continuously-valued.

Table 1: The attribute weight updating algorithm, where the inputs are the current training instance x, current concept description D, and Classguess(x) (the predicted classification of x). Variable λ is the higher relative observed class frequency of x's actual and its predicted class. Variable y is x's most similar neighbor that is also in x's predicted class. Since the instances are normalized, step 3 yields a value in $[0, 1]$.

1. **LET** λ = max(ObservedRelativeFrequency(Class(x)),ObservedRelativeFrequency(Classguess(x)))
2. **LET** $y = \{d \in D | \forall d' \in D$ {Class(d')=Classguess(x) & Similarity(x,d) $\geq$ Similarity(x,d')}}
3. **LET** difference = $|x_a - y_a|$
4. **IF** (x's classification was correctly predicted)
   THEN increment = (1-λ)$\times$(1-difference)
   ELSE increment = (1-λ)$\times$difference
5. total-attribute-weight$_a$ = total-attribute-weight$_a$ + increment
6. total-possible-attribute-weight$_a$ = total-possible-attribute-weight$_a$ + (1-λ)

3.1 GCM-SW: An Interpretation of the Generalized Context Model

Our interpretation of Nosofsky's Generalized Context Model is named *GCM-SW*.

3.1.1 Similarity Function: The distance between instances x and y in an n-dimensional instance space is defined as:[2]

$$\text{Distance}(x, y) = \sqrt{\sum_{a=1,n} \text{Weight}_a (x_a - y_a)^2} \tag{1}$$

Similarity is subsequently defined as:

$$\text{Similarity}(x, y) = e^{-\text{Distance(x,y)}^2} \tag{2}$$

3.1.2 Classification Function: The probability of classifying x in concept c is defined as:[3]

$$\Pr(c|x) = \frac{\sum_{y \in cd(c)} \text{Similarity}(x, y)}{\sum_{c \in C} \sum_{y \in cd(c)} \text{Similarity}(x, y)} \tag{3}$$

where $cd(c)$ is the set of instances in c's concept description and C is the set of concepts to be learned.

3.1.3 Concept Description Updater: All training instances are saved in a single concept description. GCM-SW's attribute weights are defined as follows: (for each attribute a)[4]

$$\text{Weight}_a = max(\frac{\text{total-attribute-weight}_a}{\text{total-possible-attribute-weight}_a} - 0.5, 0). \tag{4}$$

The attribute weights are updated after each training instance x is classified. Its most similar neighbor y in the concept description is used to update the weights, as described in Table 1. The total-attribute-weight is incremented by a fraction of that added to the total-possible-attribute-weight$_a$. The total-attribute-weight's reward is high when it assists

[2] Missing is Nosofsky's *scale parameter*, which reflects overall discriminability in a psychological space (i.e., it increases with increasing domain knowledge). This parameter's effect is an emergent property of GCM-SW's learning behavior.

[3] Missing are Nosofsky's *concept bias parameters*, which involve other topics of attention that we do not address here.

[4] Attribute weight values are defined in the range $[0, 0.5]$ instead of $[0, 1]$ because (1) an irrelevant attribute's total attribute weight is expected to be half its total possible attribute weight and (2) we wanted irrelevant attributes to have 0 weight.

making a correct classification decision and is low otherwise. More specifically, its increment is high when either (1) a correct classification occurs and the instances' attribute values are similar to each other or (2) an incorrect classification occurs and they are dissimilar. Otherwise, the total-attribute-weight's addend is small since the attribute's value did not assist in predicting the correct classification. This algorithm attends to classes with low observed relative frequency in order to overcome highly skewed concept frequency distributions.

Finally, the weights are linearly scaled to sum to 1. This simulates the distribution of resource-limited "attention" across attributes (Nosofsky, 1986; 1987).

The weight-learning algorithm is best explained with an example. For this purpose we will study GCM-SW's ability to learn the concept "Ph.D. student" from instances (people) described with three Boolean attributes ("is enrolled","has M.S. degree", and "is married"). Suppose that GCM-SW has been trained on 4 instances, only one of which was a Ph.D. student (with attribute values <True,True,True>), the resultant total-attribute-weights settings are (0.65,0.65,0.65), and the total-possible-attribute-weights are all 0.75. If the fifth instance is incorrectly classified as a Ph.D. student and has attribute values <False,False,True> (i.e., not enrolled, no M.S., married), then the new total-attribute-weight settings are (0.8,0.8,0.65) and the total-possible-attribute-weights are all 1.0. Finally, if the sixth instance is correctly classified as a Ph.D. student and has attribute values <True,False,False> (i.e., enrolled, no M.S., unmarried), then the new total-attribute-weight settings are (1.6,0.8,0.65) and the new total-possible-attribute-weights are 1.8. This indicates that GCM-SW has learned that the attribute "is enrolled" is more predictive of the Ph.D. class than either "has M.S. degree" or "is married." Good attribute predictors will have higher attribute weights than less relevant attributes.

3.2 The Utility of the GCM-SW Model

The GCM-SW algorithm increases concept learning rate by allowing relevant attributes to have greater influence in similarity calculations and, subsequently, classification decisions. When learning tasks involve concepts with conflicting attribute relevancies, GCM-SW learns concept descriptions more quickly than does the equivalent algorithm that weights all attributes equally. We studied GCM-SW's learning behavior, with and without learning attribute weights, in a domain described by twelve Boolean attributes. Instances were randomly drawn from a uniform distribution of the instance space. Only positive instances had values of 1 for their first attribute. The resulting learning curve (Figure 1) indicates that GCM-SW learns the concept far more quickly when attribute weights are learned.

GCM-SW assigns the same initial weight to each attribute. As training progresses, the relevant attribute's weight increases while the irrelevant attributes' weights decrease. Figure 2 summarizes GCM-SW's weight-learning behavior for this concept learning task.

3.3 A Limitation of the GCM-SW Model

Nosofsky (1986, 1987) did not apply the GCM to complex concept learning tasks. In particular, concepts were assumed to be disjoint. They were also assumed to exhaust the instance space. These simplifications limit the GCM's learning capabilities.

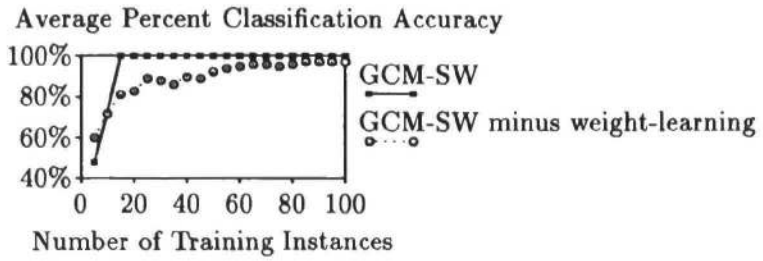

Figure 1: Average percent classification accuracies (over 25 trials) of GCM-SW with and without its attribute weight-learning capability. The rate of learning is increased when attribute weights are learned.

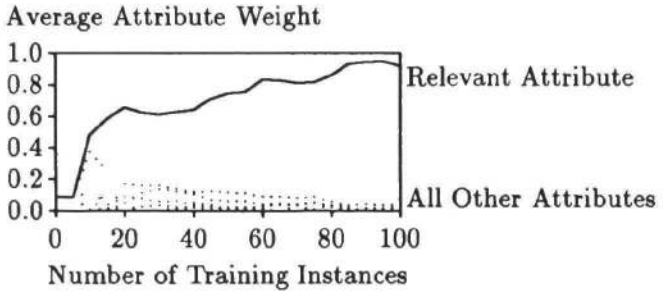

Figure 2: GCM-SW's average (25 trials) attribute weight settings during training. The relevant attribute's weight is quickly distinguished from the irrelevant attributes' weights.

The GCM was designed to model the *attention-optimization hypothesis*, which posits that humans optimize their classification performance by distributing their attention among the given attributes. To satisfy this assumption, the GCM's representation needs to be extended to simultaneously learn overlapping concepts which require conflicting attribute weight settings to optimize classification performance. In Section 4, we present GCM-MW, an extension of the GCM that can learn overlapping, non-exhaustive concept descriptions, where each concept is associated with a unique set of attribute weights. We show that GCM-MW has a faster learning rate than GCM-SW for these concept learning tasks.

4. LEARNING MULTIPLE PSYCHOLOGICAL SPACES

Our extension of the GCM-SW algorithm is *GCM-MW*, which learns a separate set of attribute weights for each concept description being learned. Like GCM-SW, GCM-MW is best described as an instantiation of our instance-based process framework.

4.1 GCM-MW: An Extension of the Generalized Context Model

4.1.1 Similarity Function: GCM-MW's similarity function is concept-dependent, but otherwise identical to that used in GCM-SW. The distance between instances x and y in an n-dimensional instance space with respect to a concept c is defined as:

$$\text{Distance}(c, x, y) = \sqrt{\sum_{a=1,n} \text{Weight}_{c_a}(x_a - y_a)^2} \tag{5}$$

Similarity is subsequently defined as:

$$\text{Similarity}(c, x, y) = e^{-\text{Distance(c,x,y)}^2} \tag{6}$$

Similarity is concept-dependent. For example, we would expect that, for any tiger t and cat c, Similarity$(animal, t, c)$ is greater than Similarity(pet, t, c).

4.1.2 Classification Function: The probability of classifying x in concept c is:

$$\Pr(c|x) = \frac{\sum_{y \in cd(c)} \text{Similarity}(c, x, y)}{\sum_{c \in C} \sum_{y \in cd(c)} \text{Similarity}(c, x, y)} \tag{7}$$

where $cd(c)$ is the set of instances in c's concept description and C is the set of concepts to be learned. (Note that instances in $cd(c)$ are either positive or negative.)

4.1.3 Concept Description Updater: Finally, weight learning in GCM-MW is the same as in GCM-SW except that *each concept description has a separate set of weights.*

$$\text{Weight}_{c_a} = max(\frac{\text{total-attribute-weight}_{c_a}}{\text{total-possible-attribute-weight}_{c_a}} - 0.5, 0). \tag{8}$$

In our previous example, we saw that being enrolled in a Ph.D. program was highly diagnostic of the Ph.D. class. However, if GCM-SW was simultaneously trying to learn the concept of being married, the attribute weight for "is enrolled" would decrease. Subsequently, this attribute would have less impact on classifying people as Ph.D. students. GCM-MW avoids this conflict by maintaining a separate set of attribute weights and description for each concept being learned.

Updating the attribute weights after each classification continuously changes a concept's similarity function. This notion was originally captured by Salzberg's (1988) exemplar-based system, which inspired our work on weight-learning algorithms. We extended Salzberg's algorithm by (1) removing an ad-hoc parameter that required different settings for each domain, (2) defining how it learns weights for continuously-valued attributes, and (3) extending it to learn attribute weights separately for each concept. Thus, GCM-MW learns similarity functions independently for each concept description. This is an important capability for distinguishing different contexts during classification and other problem solving tasks.

In summary, GCM-MW is an incremental concept learning algorithm that updates its concept descriptions after classifying each training instance and learns the appropriate attribute weight settings *for each concept.* Furthermore, GCM-MW's concept descriptions need not exhaust the instance space nor be disjoint. Finally, since GCM-MW employs a separate interpretation of the instance space for each concept, it can represent independent and overlapping concept descriptions.

4.2 The Utility of the GCM-MW Model

GCM-MW models the attention-optimization hypothesis even when the learned concept descriptions overlap and differ in their attribute weight settings. We applied GCM-SW and GCM-MW to an instance space described by five numeric-valued attributes whose values

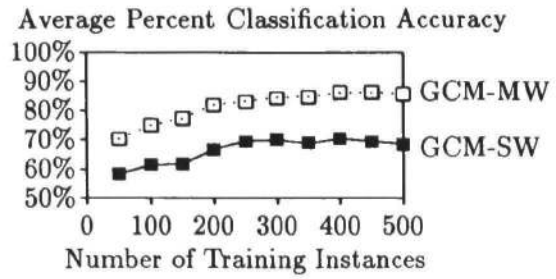

Figure 3: Average learning curves (over 25 trials) on a domain with five overlapping and non-exhaustive concepts. GCM-MW learns more quickly by building independent descriptions for each concept.

range in $[0, 100]$. This space contains five overlapping and non-exhaustive concepts. Each concept is defined in terms of a single attribute. For each $n \in [1, 5]$, the n^{th} concept's instances have values greater than 50 in their n^{th} attribute. Instances are randomly drawn from a uniform distribution over the instance space and can be members of any subset (possibly empty) of the five concepts.

The learning curve in Figure 3 summarizes the applications of these two algorithms to this domain. In summary, GCM-MW learns the concepts' descriptions more quickly because it builds an independent interpretation of the instance space for each concept.[5] In effect, GCM-MW simultaneously learns multiple psychological spaces.

5. SOME LIMITATIONS AND FUTURE WORK

While GCM-MW performs well along a number of dimensions, it has several limitations. First, we have incorrectly defined similarity such that the similarity between two instances cannot increase with additional attribute comparisons, even when the additional attribute values indicate that these instances are similar. We plan to experiment with variants of Tversky's (1977) contrast model, which defines similarity both in terms of attribute value commonalities *and* (directed) differences, in an attempt to solve this problem.

Also, GCM-MW's classification function's behavior in the presence of millions of attributes and/or instances is needlessly expensive. Some control is needed so that similarities are computed only for relevant instances (i.e., those in the concept's description that are most similar to the instance being classified). We plan to explore the role of attention and intelligent indexing schemes in the future (McNulty, 1988).

We plan to demonstrate how IBL models can learn non-normal category distributions. Neumann (1977) presented evidence that people can learn such distributions. Moreover, Fried and Holyoak (1984) argue that only instance-based models can describe non-normal category distributions.

We would also like to determine whether the GCM-SW and GCM-MW models simulate human categorization behavior on complex concept learning tasks. While Nosofsky (1986, 1987) showed that the GCM model can closely fit human subject data on simple tasks, he

[5] Preliminary experiments indicate that GCM-SW's classification accuracy begins to approach GCM-MW's after processing several thousand instances from this domain.

did not describe how it behaves when concept descriptions overlap and have conflicting, optimal attribute-weight settings.

6. SUMMARY

In this paper, we introduced two instance-based process models, GCM-SW and GCM-MW. Both models are based on Nosofsky's (1986, 1987) generalized context model. We introduced a simple algorithm for learning GCM's attribute weights. We have also argued that a separate set of relative attribute weights for each concept description, as used in GCM-MW, is needed to represent and accurately learn complex concept descriptions (i.e., overlapping). GCM-MW can learn independent and overlapping concept descriptions by developing a separate psychological space for each concept to be described.

However, our model has several limitations. For example, similarity should not increase monotonically with fewer numbers of attributes. Also, our instance-based model needlessly computes the similarity of an instance with all previously observed instances for each classification. It should instead compute similarities for only a relevant subset of the instances. We plan to extend our model in these and other directions in the future.

ACKNOWLEDGEMENTS

We would like to thank Dennis Kibler, John Gennari, David Ruby, and our reviewers for their suggestions on earlier drafts of this paper.

REFERENCES

Aha, D. W. (1989). Incremental, Instance-Based Learning of independent and graded concept descriptions. To appear in *Proceedings of the Sixth International Workshop on Machine Learning.* Ithaca, NY: Morgan Kaufmann.

Fried, L. S., & Holyoak, K. J. (1984). Induction of category distributions: A framework for classification learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 10,* 234–257.

McNulty, D. M. (1988). Extending moment analysis with directed attention to handle structural variations in character recognition. In *Proceedings of the Seventh Biennial Conference of the Canadian Society for the Computational Studies of Intelligence* (pp. 206–212). Edmonton, Canada: Morgan Kaufmann.

Neumann, P. G. (1977). Visual prototype formation with discontinuous representation of dimensions of variability. *Memory & Cognition, 5,* 187–197.

Nosofsky, R. M. (1986). Attention, similarity, and the identification-categorization relationship. *Journal of Experimental Psychology: General, 15,* 39–57.

Nosofsky, R. M. (1987). Attention and learning processes in the identification and categorization of integral stimuli. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 13,* 87–108.

Salzberg, S. (1988). *Exemplar-based learning: Theory and implementation* (Technical Report TR-10-88). Cambridge, MA: Harvard University, Center for Research in Computing Technology.

Shepard, R. N. (1987). Toward a universal law of generalization for psychological science. *Science. 237,* 1317–1323.

Tversky, A. (1977). Features of Similarity. *Psychological Review, 84,* 327–352.