A Model of Learning by Incremental Analogical Reasoning and Debugging

Mark H. Burstein
Department of Computer Science
Yale University
New Haven, Connecticut 06520

Abstract

This paper presents a model of analogical reasoning for learning. The model is based on two main ideas. First, that reasoning from an analogy presented by a teacher while explaining an unfamiliar concept is often determined by the causal abstractions known by the student to apply in the familiar domain referred to. Secondly, that such analogies, once introduced, are extended incrementally, in attempts to account for new situations by recalling additional situations from the same base domain. Protocols suggest that this latter process is quite useful but extremely error-prone. CARL, a computer program that learns semantic representations for assignment statements of the BASIC programming language is described as an illustration this kind of analogical reasoning. The model maps and debugs inferences drawn from *several* commonly used analogies to assignment, in response to presented examples.

1 Introduction

It has often been said among AI researchers that learning one new concept requires knowing an enormous amount beforehand. In learning by analogy this is particularly true. This paper outlines an approach to learning by analogy showing one way that the organization of that knowledge can also be important. In developing the model presented here, I concentrated particularly on how analogies presented by a teacher or text might be used by a student in forming an initial model of events in an unfamiliar domain. The model was motivated in part by observations derived from recorded protocols of the behavior of several students I tutored in introductory computer programming in the BASIC language.

A number of models of analogical reasoning in AI have been developed around forms of partial pattern matching, under which objects in one domain are first associated with those in another, and ordered relations or frames with case slots of some kind are then placed in correspondence [Evans 68, Brown 77, Winston 80, Winston 82]. These algorithms were based on the assumption that a best partial match could be found by first accumulating evidence for all possible object-to-object mappings between two situations and then choosing the one that placed the largest number of attributes and relations in correspondence.

This approach has several major drawbacks in a model of analogical learning. First, it presupposes that well defined, bounded representations exist for the situations in both the *base* or familiar domain and in the *target* domain. The partial matching model thus requires some prior representation of the objects and relations in the target domain when, in fact, there may not be any such representation when the analogy is first presented. Equivalently, the student's knowledge of the domain may be wrong or inconsistent with the analogy, making matching difficult or useless. The point of presenting an analogy to a student is to aid him in *constructing* a representation of a target situation, or to correct problems in a prior representation. Since a poor match does not indicate how such a representation might be constructed, it cannot form the basis of a general theory of how students learn by analogy.

A related problem with theories of analogical reasoning based principly on description matching is that rich situation descriptions often contain many objects and relations not taking part in analogies to those situations. As the amount of detail in the representation of a situation increases, the combinatorial complexity of a bottom-up analogical matching process also dramatically increases. Larger and larger numbers of extraneous object-object mappings must be tried and discarded. Winston [Winston 80, Winston 82] suggested that attention to important relations, such as those connected by causal links, can reduce the computational complexity of the matching process to some degree. Yet, even in strictly causal models, sub-systems can quite often be expanded to greater levels of detail [deKleer and Brown 81, Collins and Gentner 82], thereby introducing objects which not playing direct roles in analogies based on those systems.

What is required to address these objections is an approach based on analogical *mapping* that includes a set of strong heuristics for delimiting what is to be "imported" from the base to the target domain. One important way the analogical mapping process can be usefully constrained is by focusing attention on the abstractions embodying the inferences made when first interpreting a presented base domain situation. The hierarchical structure of these representations can help in determining a useful mapping without prior specification of *all* of the objects potentially involved in complete descriptions of two related situations.

It has been argued [Schank and Abelson 77, Schank 82] that abstract causal, plan and goal structures play an important role in the interpretation of natural language texts. If this is the case, then they must also be available when interpreting texts containing analogies. Winston's system used causal links to guide its matching process, but this generally occurred only if those links were were present in the input representations. At no time did the system look at the rules supporting represented causal links to find features or relations that might have useful correspondences. I suggest that the causal abstractions underlying represented base domain situations provide important constraints on the analogical mapping process, particularly when little or nothing is known of the target domain.

The work reported here was supported in part by the Advanced Research Projects Agence of the Department of Defense, and monitored by the Office of Naval Research under contract No. N00014-75-C-111.

Many aspects of this approach to analogical learning are illustrated the behavior of a computer program, CARL, that attempts to learn about various kinds of assignment statements by building conceptual models drawn from several different analogies. CARL makes a number of the errors I found my students made as they tried to use the same analogies [Burstein 81]. The following is the English equivalent of one portion of a dialogue that CARL engages in (actual inputs are in Conceptual Dependency [Schank and Abelson 77]) :

```
>> The computer remembers a number by putting it
     in a variable.
>> A variable is like a box.
>> To put the number 5 in the variable named x type 'X=5'.
>> How do you get the computer to remember the number 9?
CARL: Type X = 9
>> Suppose I then type 'X=6'.
>> Now what's in the variable X?
CARL: 6 is in X and  9 is in X
>> No, the 9 replaces the 6 in X.
CARL: OK
>> Suppose I type 'Y=5' and then 'X=Y'.
>> Now what's in the variable X?
CARL: Y is in X
>> No, it means put the number in Y into X.
CARL: OK
>> Now what's in the variable Y?
CARL: Nothing.
>> Guess again.
CARL: 5 is in Y    (by analogy to algebraic equality)
>> Correct.
CARL: OK
```

In this example, CARL is first introduced to assignment primarily in terms of an often used analogy between variables and boxes. The initial statement of the analogy includes both a statement of the principle object class correspondence (i.e. box::variable), and an example of an assignment, described in terms of a parallel "box domain" situation.

These statements taken together provides CARL with enough information to form an initial description of the relation between variables and numbers formed by assignment, principly that this new relation is "like" physical containment in some specific ways. CARL concludes that the relation between the variable X and the number 5 *results* from the assignment action requested by typing "X=5", in the way "the object is in the box" results from the action "put an object in a box". Most attributes of boxes, physical objects, and many possible relations between the two are completely ignored in this process.

Other consequences of this analogy, including many erroneous ones, are discovered during further exercises in the new domain. As new examples are presented, other, related situations are retrieved from CARL's memory of the "box" domain, the mapping process is repeated, using the object and predicate correspondences formed initially. Errors are discovered and corrected either by explicit statements of the tutor, or, when possible, by the use of internally generated alternative hypotheses. CARL uses several other analogies in this process, including the similarity between assignment and equality, and the common belief that computer actions mimic human actions. As in the above example, secondary analogies are used to discover alternate hypotheses about the effects of statements when errors are detected.

When a class of assignment statements is successfully modeled by a structure formed in the mapping process, parsing and generation rules are also associated with the new structure. In this way, CARL learns to manipulate most common forms of assignment statements.

2 An Initial Structure Mapping Theory

The mapping algorithm developed for CARL was influenced by a model of analogical reasoning called *structure mapping*, developed for a series of psychological studies Gentner [Gentner 82, Gentner and Gentner 82]. This model was used to describe the effects of human reasoning when given scientific or "explanatory" analogies, such as those below. [Gentner 82]

> The hydrogen atom is like the solar system.
>
> Electricity flows through a wire like water through a pipe.

The mapping algorithm Gentner proposed for reasoning from analogies of this type circumvented some of the problems with match-driven models in that it did not require a prior representation of the target domain. However, it was underspecified as a cognitive process model. By her model, first-order relations, or predicates involving several objects or concepts, are mapped *identically* from one domain to the other under a prespecified object-object correspondence. After identical first-order relations have been used to relate corresponding objects in a target situation, second-order predicates, such as causal links between first-order relations, were also mapped. While this does suggest a way to map new structures into an unfamiliar domain, it does not give a good account of how corresponding objects are first identified, nor does it constrain *which* relations are mapped. It also does not allow for mappings between non-identical relations, which I will argue is often necessary.

The need to constrain the set of relations mapped can be seen from Gentner's representation of the solar system model, and the mapping that her system suggests to a model for the atom. In that representation, the sun is related to a planet by the predicate HOTTER-THAN, as well as the predicates ATTRACTS, REVOLVES-AROUND and MORE-MASSIVE-THAN, three relations which are causally linked in the description of the orbiting system. Gentner claimed that the HOTTER-THAN relation was *not* mapped to the atomic model, in accord with most people's intuitions. However, her formal mapping algorithm could not predict this. Many other attribute comparisons, such as BRIGHTER-THAN, could also have been present in a description of the solar system. Presumably, these relations would not be mapped either.

The explanation given for this phenomenon was in terms of a general condition on the mapping process she termed *systematicity*. This condition is essentially that "predicates are more likely to be imported into the target if they belong to a system of coherent, mutually constraining relationships, the others of which are mapped." [Gentner and Gentner 82]

In CARL, this condition appears as the top-down heuristic delimiting the relations considered for mapping. When a causally connected structure is in memory to describe a base domain situation, only relations taking part in that structure are considered for mapping. Within that set of relations, simple attribute comparisons like HOTTER-THAN, LARGER-THAN are not mapped if no corresponding attributes can be found in the target domain. The result of mapping a causally-connected structure under these conditions is a new, parallel causal structure characterizing the target example. Objects in the target example are made to fill roles in that structure using stated object correspondences between the domains when available, and otherwise by their appearance in roles of actions and other relations with objects that do have known correspondences. Thus, in the analogy between variables and boxes, an indirect correspondence is formed between the situational role of a physical object that is INSIDE a box, and a number assigned to a variable.

This causally directed mapping process thus forms new structures in domains where none existed before, while allowing relations to be mapped with some consideration of what *is* known of the objects and relations in the target domain. The process is also top-down in that classes of objects are only placed in correspondence across domains explicitly by statement of the teacher, or, indirectly, during mapping by the appearance of an object in a role of the target structure where a different object was described in same role in the base domain structure.

3 Mappings between non-identical relations

Another problem with both Winston's and Gentner's models of analogical reasoning can be found in the claim that all relations are mapped "identically" from one situation description to another. Winston's matcher embodied this claim since it only found correspondences between identical predicates in two representation. This condition may apply in analogies where the domains overlap or are closely related, as in the standard geometric analogies dealt with by Evans, and many scientific analogies. However, it is much too strong a claim in general. When analogies are formed between physically realizable situations and purely abstract ones, like those of mathematics or computer programming, it is impossible to maintain the "identical predicate" mapping position.

Probably the most important thing implied by the analogy between boxes and variables is that variables can "*contain*" things. That is, the relationship that exists between a box and an object inside the box is, in some ways, similar to the relationship between a variable and the number associated with that variable. The important inference that must be preserved in the mapping is that since one can *put* things *in* boxes, then there must be a way to "*put*" numbers "*in*" variables as well.

The problem from the standpoint of Gentner's model, is that the relationship which gets mapped from the "box world" to the "computer world" is precisely that of *physical containment*. That is, the interpretation that results from copying this relation into the programming domain is that a number is physically INSIDE a variable. Unfortunately, not *all* of the inferences which the relation INSIDE takes part in apply to numbers "in" variables. For example, there is no primitive action in BASIC that corresponds precisely to the action "take out of". CARL determines that target relation may have some differences from the physical relation INSIDE by noting that numbers violate a typical constraint on the object slot of the INSIDE relation. Since a number is not a physical object, it is not in the class of objects normally "contained" in other objects. Thus, the relation suggested between variables and numbers cannot be the standard notion of containment, but instead must be some (as yet undetermined) analogical extension of that relationship.

When an attempt to map a relation directly results in such a constraint violation, a *virtual relation* is formed in the target domain that is a "sibling" of the corresponding base domain relation, or an ancestor at some higher level in is-a the hierarchy of known relational predicates. The constraints placed on the roles in new virtual relations are determined primarily from the classes of the objects related in the target domain example. Thus, from the analogy to boxes, a new predicate INSIDE-VARIABLE is formed to relate variables and their "contents", initially constrained to be numbers. Inferences are associated with this new relation as they are successfully mapped from the base domain, learned independently in the new domain, or inherited from other analogies.

4 Overview of CARL's analogical reasoning process

CARL develops simple causal or inferential structures in the target domain by retrieving structures in memory for the familiar domain, and adapting them to the new domain, using a top-down mapping process that preserves the causal/temporal links explicitly specified in those structures. In the retrieval process, base domain objects are substituted for target domain objects when those correspondences can be determined from the presented description of the analogy. The mapped predicates are subject to transformation within an abstraction hierarchy, as described above. Subsequent use of the same analogy may either be for the purpose of mapping a related structure - a related type of action situation, or a more context-specific version of the originally mapped structure - using the mapping developed initially, or an extension of that mapping to include new predicates.

The structure first mapped when CARL is given the box analogy is a simple conceptual description of the action of putting an unspecified object in a box, together with the standard result of that action, that the object is then INSIDE that box. The result relation INSIDE is transformed in the mapping process to a new relation which I will refer to here as INSIDE-VARIABLE. The action PTRANS representing a change of physical location is replaced with the more general predicate TRANS, indicating any state change, and causally connected to the result, that the OBJECT of the TRANS is INSIDE-VARIABLE after the action is completed. The standard precondition on putting an object in a box, that the object fit in the box, is ignored since neither variables nor numbers are known to have a physical SIZE.

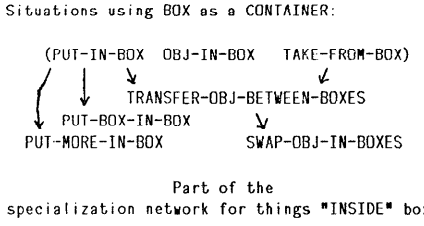
5 Incremental analogical reasoning for learning

Even when analogies are based on simple actions, the specific inferences to be made may vary considerably depending on the context in which the action occurs. For example, throwing a rock at a brick wall and throwing one at a glass window are known to have very different consequences. Though an analogy to a thrown rock might imply indirectly that the specific inferences valid in such alternate contexts will apply in some target domain situation, in practice each such situation must be explored to determine the true extent to which the analogy is valid. Extending analogies in this fashion is an error-prone process as often as it is useful. CARL attempts to extend an analogy to such alternate-context inferences only as they are required to interpret new situations in the target domain, making a number of errors in the process.

In the protocols I examined, these errors appeared only when the context in the target domain made them potentially useful inferences. For example, when a statement like "X=Y" was first introduced, it was necessary to explain that this meant X was given the value that Y had previously. One student then inferred that Y was must contain "nothing", since that was what happened with boxes. Finding and correcting these special-case errors in the inference rules mapped is treated here as an *incremental* process that requires observation and consideration of a number of examples in the new domain.

This behavior is modeled in CARL by retrieval and mapping of related causal structures from a base domain (the box domain in this case). One interpretation found for "X=Y" in terms of the box analogy is that it is like moving an object from one box to another. Information saved about the mapping of the prototype "put an object in a box" is used both in finding the structure representing this more specific action and subsequently to map it back to the programming domain.

CARL's memory organization for knowledge of simple action-based domains involving familiar objects is in part an extension of an object-based indexing system developed by Lehnert [Lehnert 78, Lehnert and Burstein 79] for natural language processing tasks. So that CARL can retrieve a variety of special case situations, the retrieval process was augmented using a discrimination system as in the specification hierarchy model of episodic memory used by Lebowitz and Kolodner [Lebowitz 80, Kolodner 80]. Precondition/result-based indexing was also added so that actions and simple plans could be retrieved in response to requests to achieve specific goals. Any of these forms of indexing may be used in finding a suitable structure to map. For familiar domains, the system assumes a large set of fairly specific generalized situations exists, detailing the causal inferences expected in each case. As an example, CARL contains the following structures describing the effects of some simple actions involving containers.



Once an initial mapping is formed from the causal structure PUT-IN-BOX, specializations of that structure are available when new examples presented to CARL. Also, once the containment relation is formed for variables, expectations are established for the other "primitive" situations involving containers. Thus, from the fact that variables can "contain" numbers, it is *expected* that they can be "put in" and "removed".

The result of mapping these additional structures is the formation of a corresponding set of structures for assignment statements. Many of these new structures contain erroneous inferences, which are "debugged" locally, by observation of the actual results entailed, or simply thrown out. Information for recognizing each structure, in this case parsing and generation information for program statements, is attached to each new structure during the analysis of the examples that caused them to be mapped. Corrections propagate downward from the initial prototype formed, to the variants mapped subsequently. Thus, for example, the fact that 'X=5' removes old values of X also automatically applies to 'X=Y'.

6 Summary

In developing CARL, I have been concerned with a number of related issues in the learning of basic concepts in a new domain by a combination of incremental analogical reasoning from multiple models. I have tried here to motivate the need for top-down use of known abstractions in this process. This was found to be necessary both to limit the analogical reasoning required to form some initial concepts in the new domain, and to allow for incremental debugging of the many errors that can result from the use of analogies. The process described here is heavily teacher-directed, but allows for fairly rapid development of a working understanding of basic concepts in a new domain.

Acknowledgements: I would like to thank Dr. Chris Riesbeck and Larry Birnbaum for many helpful comments on drafts of this paper.

References

1. Brown, Richard. Use of Analogy to Achieve New Expertise. Tech. Rept. 403, M.I.T. A.I. Memo, May, 1977.

2. Burstein, Mark H. Concept Formation through the Interaction of Multiple Models. Proceedings of the Third Annual Conference of the Cognitive Science Society, Cognitive Science Society, August, 1981, pp. 271-274.

3. Collins, Allan and Gentner, Dedre. Constructing Runnable Mental Models. Proceedings of the Fourth Annual Conference of the Cognitive Science Society, Cognitive Science Society, August, 1982, pp. 86-89.

4. de Kleer, J. and Brown, J. S. Mental Models of Physical Mechanisms and their Acquisition. In *Cognitive Skills and Their Acquisition*, Anderson, John R., Ed.,Lawrence Ehrlbaum and Assoc., Hillsdale, NJ, 1981, pp. 285-309.

5. Evans, Thomas G. A Program for the Solution of Geometric Analogy Intelligence Test Questions. In *Semantic Information Processing*, Marvin L. Minsky, Ed.,M.I.T. Press, Cambridge, Massachusetts, 1968.

6. Gentner, Dedre. Structure Mapping: A Theoretical Framework for Analogy and Similarity. Proceedings of the Fourth Annual Conference of the Cognitive Science Society, Cognitive Science Society, August, 1982, pp. 13-15.

7. Gentner, D. and Gentner, D. R. Flowing waters or teeming crowds: Mental models of electricity. In *Mental Models*, Gentner, D. and Stevens, A. L., Ed.,Lawrence Erlbaum Associates, Hillsdale, New Jersy, 1982.

8. Kolodner, Janet L. Retrieval and Organizational Strategies in Conceptual Memory: A Computer Model. Tech. Rept. 187, Yale University. Department of Computer Science, 1980. Ph.D. Dissertation

9. Lebowitz, M. *Generalization and Memory in an Integrated Understanding System* . Ph.D. Th., Yale University, October 1980.

10. Lehnert, W. G. Representing Physical Objects in Memory. Tech. Rept. 131, Yale University. Department of Computer Science , 1978.

11. Lehnert, W.G. and Burstein, M.H. The Role of Object Primitives in Natural Languag. Processing. Proceedings of the Sixth International Joint Conference on Artificial Intelligence, IJCAI, August, 1979, pp. 522-524.

12. Schank, R.C.. *Dynamic memory: A theory of learning in computers and people.* Cambridge University Press, 1982.

13. Schank, R.C. and Abelson, R.. *Scripts, Plans, Goals and Understanding.* Lawrence Erlbaum Associates, Hillsdale, New Jersy, 1977.

14. Winston, P. Learning and reasoning by analogy: the details. Tech. Rept. 520, M.I.T. A.I. Memo, May, 1980.

15. Winston, P. "Learning new principles from precedents and exercises." *Artificial Intelligence 19* (1982), 321-350.