PURPOSE-DIRECTED ANALOGY

Smadar Kedar-Cabelli

Department of Computer Science
Rutgers University
New Brunswick, NJ 08903

Abstract

Recent artificial intelligence models of analogical reasoning are based on mapping some underlying causal network of relations between analogous situations. However, causal relations relevant for the purpose of one analogy may be irrelevant for another. We describe here a technique which uses an explicit representation of the purpose of the analogy to automatically create the relevant causal network. We illustrate the technique with two case studies in which concepts of everyday artifacts are learned by analogy*.

[A]ny two things which are from one point of view similar may be dissimilar from another point of view.

-K. Popper, The Logic of Scientific Discovery

I Introduction

A recent development in artificial intelligence (AI) research on analogical reasoning has been to recognize that analogy involves mapping some underlying causal network of relations between analogous situations [2, 3, 5, 26]. Often, however, there are numerous causal networks describing the situations. Which are the relevant ones to map when performing a particular analogy? Causal relations relevant for the purpose of one analogy may be irrelevant for another. A more robust model of analogical reasoning cannot always reason from predefined causal networks, but needs the ability to automatically generate the appropriate network based on the purpose of the analogy being performed.

This paper describes a new technique, *Purpose-Directed Analogy*, which is designed to address the above limitation using a specialized notion of 'purpose' to automatically generate the relevant causal network. In particular, we are developing a system to learn concepts of everyday artifacts by reasoning analogically from a known example of an artifact to an unknown example. The specialized notion of 'purpose' is the purpose for which these artifacts will be used.

*This research is supported by GTE Laboratories, under Contract No. GTE840917.

Artifacts can be viewed as objects designed to enable people to perform certain actions (chairs to sit on, pens to write with, and so on). If the goal of an agent is to perform an action, often the agent may need to recognize an artifact which will enable him to perform that action (or a plan of actions leading to the goal). One way to recognize such an artifact is by reasoning analogically from a known example of the artifact to an unknown example. The central idea is that:

> *Two examples will be considered analogous if they share a network of relations which demonstrates how both can be used for the same purpose.*

Thus, performing an analogy can be viewed as a subprocess of a more global problem solving process (as in [11]): to enable an agent to proceed with an action he desires to perform by being able to recognize objects that facilitate that action.

For example, suppose an agent is thirsty, and would like to drink hot liquids. Assume that as a result, the agent wants to learn the concept HOT-CUP: objects whose purpose is to enable the drinking of hot liquids. One way to learn the concept is to be able to determine if a new example (a styrofoam cup, say) is analogous to a known, prototypical example (a ceramic mug) in ways relevant for the purpose of a HOT-CUP. If a cup were needed for a different purpose (ornamental or religious, say), a different network of relations would be relevant.

Section II presents a unifying framework for concept learning by analogy in order to compare existing models and point to a key limitation. Section III describes the Purpose-Directed Analogy technique. Section IV illustrates the technique with two case studies. One case study involves learning the concept of a cup for the purpose of drinking hot liquids. The second case study involves learning the concept of a vehicle in the context of identifying vehicles violating the legal statute "A vehicle is prohibited in a public park". We conclude in section V with a discussion of limitations of the technique, future work, and a summary.

II Related Research and a Limitation

A. Discussion

A common view is that analogy is powerful because it allows us to learn about an unfamiliar situation by

mapping over many aspects of a familiar situation with a dramatic savings in reasoning. To highlight the directionality in the mapping, the familiar situation is often referred to as the *base situation* from which aspects are mapped over to the unfamiliar, or *target situation*, [5]. Thus in the analogy "Science is like a jigsaw puzzle", the less well-understood process of scientific discovery is likened to the working out of a jigsaw puzzle--a more familiar activity. As a result, many of the properties of scientific inquiry are highlighted, without needing separate explanation.

We present in this section a simple, four-stage unifying framework that describes existing AI models of concept learning by analogy. (In fact, this framework encompasses other forms of analogical reasoning such as problem-solving by analogy and metaphor comprehension [8].) We then discuss three such models [5, 2, 26] from this common perspective. We examine the limitation which we address in this paper: the inability of these models to automatically generate a causal network relevant for the purpose of a particular analogy.

B. Concept Learning by Analogy: Unifying Framework

The problem of concept learning by analogy, and the four-stage unifying framework for solving it, is stated in figure II-1.

We illustrate each stage by the analogy "The hydrogen atom is like our solar system" from [5]. The framework we present is slightly more general than the models it describes: most of these models simplify the reasoning by supplying the base example instead of retrieving it. In this analogy, the potentially analogous base concept 'solar system', is provided as input, rather than *retrieved*.

First, independent relations and causal networks of relations describing the base concept are *derived*. By a *causal network of relations*, we mean a set of relations related by any higher order relations such as 'physical-cause(ri,rj)', 'logically-implies(ri,rj)', 'enables(ri,rj)' and so on. (This is a broader sense of 'causal' than is sometimes used [5].) *Independent relations* are those not belonging to a causal network. The causal network of relations in this example describes that 'the sun attracting the planets *causes* the planets to orbit the sun'. Next, the causal network is *mapped* from the base concept over to the target, to explain why the electrons orbit the nucleus of the atom. Finally, the correctness of the mapping is *justified*: that in fact 'the nucleus attracting the electrons causes the electrons to revolve around the nucleus.'

The unifying framework does not perform any concept learning, in the sense that it does not modify the the system's representation of the target concept in any way. In order to model concept learning following the analogical reasoning, this framework is used in conjunction with (possibly) three subsequent stages. First, the concept may be learned by simply *retaining* the causal structure which was mapped to the target concept. For instance, more is learned about the atom

Figure II-1: Unifying Framework for Concept Learning by Analogy

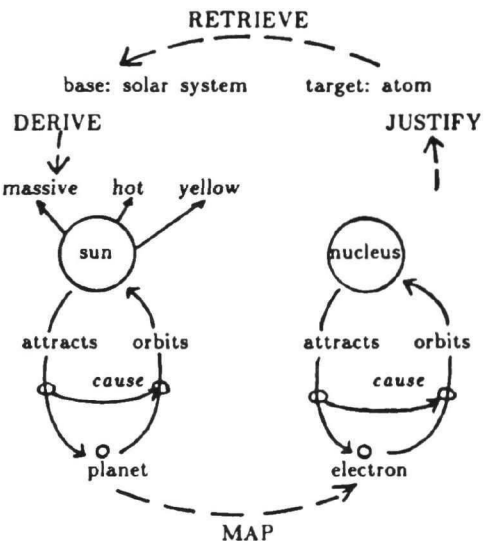
Given:

- a new, target concept, (e.g. the atom)

Find:

- a familiar, base concept, (e.g. the solar system)
- causal networks of relations of the base concept, and
- causal networks of relations of the target concept derived from the base concept

Process:



by retaining the causal structure describing why the electrons orbit the nucleus. In addition, concept learning might involve forming a *generalization* of the target and base, as in [26]. A generalized concept of 'attractive force' may be learned as a result of the above analogy. Furthermore, learning could involve *debugging* or refining a 'faulty' causal structure or generalization, by repeated analogical reasoning with the same, or different, base concepts [2, 28]. For example, the description of the atom's physical mechanisms may only be partially correct, and may be revised by analogy to other concepts.

Given the above framework, we can now discuss three recent models of concept learning by analogy [5, 2, 26]. (See [6, 8] for surveys of other work on analogy.)

C. Gentner's Domain-Independence Relevance Criterion

The central idea in Gentner's structure-mapping theory [5] is that a syntactic (domain-independent) principle can be used to select the relevant aspects of

situations for any analogy. This *systematicity principle* states that, in general, causal networks of relations are relevant to the analogy between situations, while independent relations are not. The justification is that analogy is defined as a reasoning process which maps over a "...system of connected knowledge, not a mere assortment of independent facts" [5, p.162]. Thus, as we saw earlier, in the analogy "The hydrogen atom is like our solar system" more is understood about the atom by mapping over causal relations. Specifically, the causal network describing why the planets orbit the sun is mapped to explain why the electrons orbit the nucleus of the atom. Note that the analogy is not intended to teach us that the nucleus of the atom is 'yellow, hot or massive' like the sun. These independent relations, not involved in the causal network, are considered irrelevant to the analogy.

Gentner's model assumes that the *relevant* causal network is given. For a different purpose, a different causal network may be relevant. Consider, for example, a different analogy with the sun: the metaphor "Juliet is the sun", from Shakespere's *Romeo and Juliet* (also discussed in [5]). We know the context in which this metaphor is conveyed: that Juliet is a woman and Romeo loves her. The purpose of this metaphor is to analogically *convey positive qualities about Juliet*, not to *convey anything about physical mechanisms!* Thus the causal network about the sun which was supplied for the previous analogy is no longer relevant.

D. Burstein's Automatic Indexing Into the Relevant Network

Burstein's model [2] also relies on the domain-independent criterion stated above. However, his model is a step closer to automatically selecting the relevant causal network among many candidate networks: it is provided with a relation used to index into the relevant causal network. One specific analogy he uses to illustrate his work is: "A variable is like a box, in that numbers can be inside variables in some ways similar to the way objects can be inside boxes". [2]. The action by which 'variable' is analogous to 'box' is explicitly supplied: they are analogous by the fact that things can be 'put inside' them. This eliminates considering many irrelevant actions involving boxes (such as stacking boxes, playing with boxes, etc.). Given the 'put-inside' action, the system is able to automatically retrieve the relevant goal/plan structure related to it: the 'store' plan is retrieved, which describes related actions such as putting things in boxes, taking things out of boxes, etc. (Actions can be thought of as relations, and the plan structure which connects actions in a higher order 'enable' relation can be viewed as the causal network of relations.) This goal/plan structure is then mapped to 'variable', to learn about storing things in variables, taking things out of variables, and so on, by analogy to boxes.

If the relevant action were not supplied, however, many actions and goal/plan structures associated with 'box' could be considered when trying to understand the analogy. Consider a student trying to understand this analogy. He will immediately eliminate many of these as being irrelevant. He is not likely to infer that variables can be 'stacked' like boxes, or that variables can be 'played with' like boxes. Why is that? A student learning about variables knows the purpose of the analogy: *to learn a command in a computer language*, and commands in a computer language enable the computer to manipulate numbers and symbols. Given several goal/plan structures, the student might dismiss 'play' or 'stack' as irrelevant for the purpose of the analogy. 'Put inside' might finally be focused on as the relevant action, and 'store' as the related goal/plan structure. So although Burstein's model is provided with an action 'put-inside' which can be used to automatically index into the relevant goal/plan structure 'store', it is supplied with exactly the relevant action, and cannot reason from the purpose of the analogy to select that action automatically.

E. Winston's Learning from Precedents and Exercises

The main scenario for learning and reasoning in Winston's work on analogy is one of guided learning (e.g. [26]). Here, a teacher supplies the system with a precedent. For instance, the system is provided with part of the Macbeth plot, describing Macbeth's relationship to Lady Macbeth, and what causes him to aspire to become king. The system is also given an exercise which describes personalities and relationships among some people. The task is to show that in the exercise 'the noble may want to be king,' by analogy to the precedent. This is accomplished by mapping a portion of the causal network shared by the precedent and the exercise. If in the precedent these relations are causally connected to the relation 'Macbeth may want to be king', then it can be (plausibly) concluded that in the exercise 'the noble may want to be king'.

Although Winston admits that "...the way things are matched depends on purpose as well as on experience" [25, p.6] currently just the appropriate causal structure needed to make the analogy was supplied. If, however, an analogy between the Macbeth story and the exercise were performed not for the purpose of *understanding Macbeth's motives*, but rather to *to understand Lady Macbeth's motives*, say, different causal relations would be considered important.

III Purpose-Directed Analogy

A. Discussion

We have argued above that Gentner's systematicity principle, Burstein's indexing into the relevant causal network, and Winston's analogies between precedents and exercises are all limited in their ability to automatically generate the network relevant for the purpose of a particular analogy, since explicit knowledge of purpose is not supplied as an input in these models.

Purpose-Directed Analogy attempts to overcome this limitation by making a specialized notion of 'purpose' an explicit input to the analogy. It uses this 'purpose' to automatically generate the relevant causal network for learning concepts by analogy. In this section we present the statement of the general problem, and the technique introduced to solve it. Section IV illustrates the technique by solving this problem in two case

studies of learning concepts of everyday artifacts. We are illustrating an initial design and partial implementation, not a fully implemented system. We have recently begun an implementation of a prototype system in PROLOG, a logic programming language [12]. (A PROLOG program consists of a set of horn clauses, a subclass of logical implications. The computation is based on resolution theorem-proving.)

B. Statement of the General Problem

We first introduce some terminology. A *concept* is a set of elements. The *goal concept* is the concept currently being learned by analogy. A *concept definition* provides a specification of logically necessary and sufficient conditions for being an element of the set, while a *sufficient concept definition* provides sufficient conditions only. An *example* of a concept is defined as an element of the set. The *domain theory* consists of default IF-THEN rules (axioms) and action operators which represent what is typically true in a real world domain. An *explanation* of how an example is a member of a concept is a proof that the example is an element of the set. The explanation can be viewed as a causal network of relations, consisting of domain-theory rules which link properties of examples, actions, and goals with the relation 'enables(ri,rj)' and 'logically-implies(ri,rj)'. An explanation *relevant for a particular purpose* can be viewed as a causal network of relations all of whose relations are related, either directly or by transitivity, to relations representing the purpose.

Concept learning by analogy as considered here differs slightly from that studied by Gentner, Burstein, or Winston. The analogy is *not* made between base and target *concepts*, but rather between base and target *examples* of the concept.

The problem, and the four-stage technique for solving it, is stated in figure III-1.

The system first *retrieves* a known, base example of the goal concept. The system then *explains* to itself how this example satisfies the purpose of the concept using the domain theory. (We make the simplifying assumption that there is a single purpose, which is given.) More precisely, using AI planning terminology, if the purpose of an artifact is to *enable an agent to perform a goal action*, then the artifact will satisfy the purpose if its structural features *enable a plan of actions* leading to the goal. It will enable a plan of actions if it *satisfies those preconditions of the actions in which it is involved*. So for example, a ceramic mug will enable an agent to drink hot liquids if it enables those preconditions of actions in a plan leading to DRINK in which it is involved: that is, if it enables the agent to PUTIN the hot liquids (i.e. pour), KEEP the hot liquid in the cup for some interval of time, GRASP the cup with the hot liquids in order to PICKUP, and finally if it enables the agent to DRINK the hot liquids. The prototypical ceramic mug clearly satisfies these preconditions with its open concavity, its non-porous, insulating material, its flat bottom, handle, and light weight.

The styrofoam cup will be considered analogous to

Figure III-1: Purpose-Directed Analogy

Given:

- goal concept (e.g. HOT-CUP)
- purpose of goal concept (e.g. enable an agent to drink hot liquids)
- domain theory (e.g. axioms such as '∀x has-part(x,handle) ⇒ graspable(x)')
- a new, target example (e.g. styrofoam-cup1)

Find:

- a familiar, base example (e.g. ceramic-mug1),
- an explanation of how the base example is a member of the goal concept (e.g. how ceramic-mug1 is a HOT-CUP), and
- an explanation of the target example is a member of the goal concept derived from the explanation of the base example (e.g. how styrofoam-cup1 is a HOT-CUP)

Process:

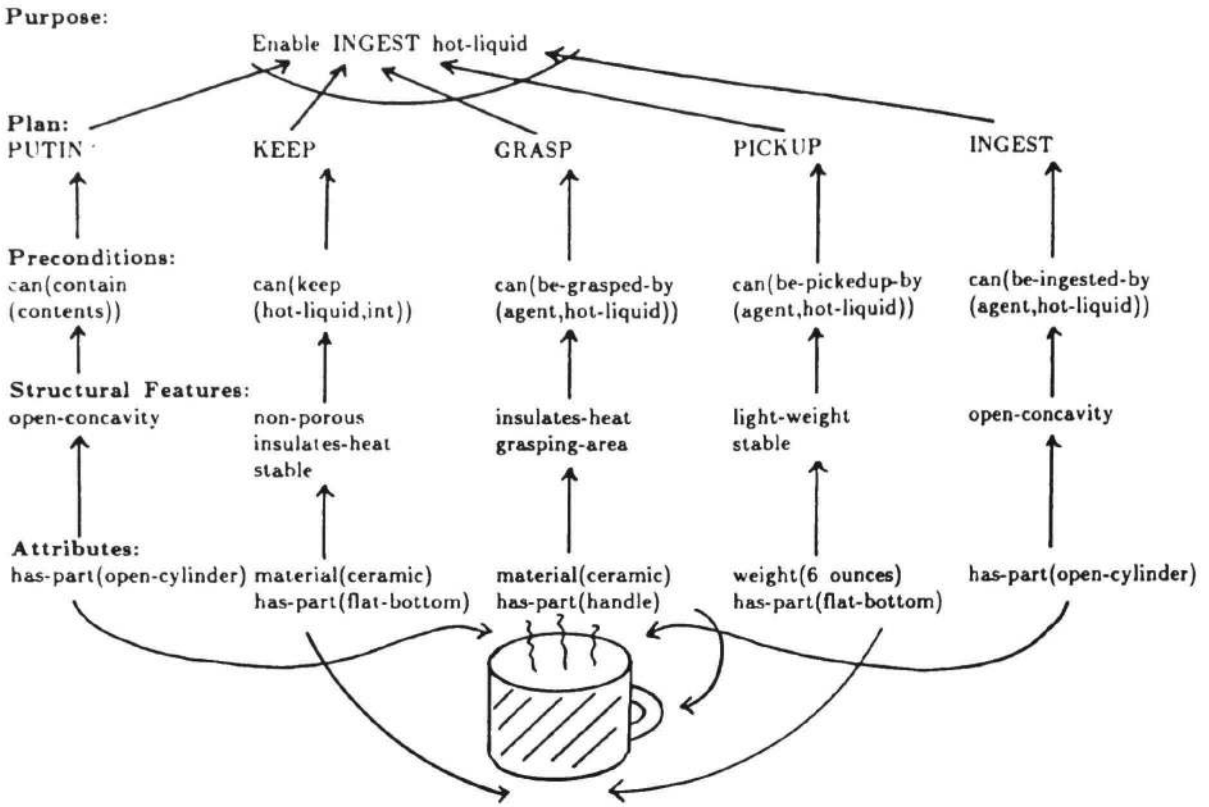
---

the ceramic mug if it too can be used for the stated purpose. To show that, the system *maps* the explanation derived for the ceramic mug, and attempts to *justify* that it is satisfied by this example. The styrofoam cup satisfies the explanation, although with slightly different structural characteristics. It differs structurally in that the styrofoam, not ceramic material, provides insulation; and the conical shape, rather than the handle, makes it graspable.

C. Relationship to Explanation-Based Generalization

The research described here adapts recent techniques for performing goal-directed and explanation-based generalization [4, 11, 14, 17, 18, 19, 27]. One key feature of these techniques is that the relevant aspects of a single example can be extracted by generating an explanation of how the example satisfies a particular goal, or purpose.

Figure IV-1: Explain How the Ceramic Mug is a HOT-CUP



In adapting these techniques to analogy, the distinction between analogy and generalization has somewhat blurred. While in analogy the explanation is mapped from a known example and modified to fit the new example; in generalization, the explanation is generated anew for each example. Is there an advantage to modifying explanations rather than generating them each time? Although it seems plausible that modifying explanations is computationally more efficient, we do not yet have experimental data to support this. One can argue, however, that observing multiple examples and modifying the explanation slightly each time provides a principled way of learning alternate ways of satisfying a particular goal or purpose (see also [13]). Current generalization techniques which analyze a *single example* do not have this capability.

The work described here is most closely related to Winston's [27], where the relevant *structural* features of an example of an artifact are extracted by explaining how the example satisfies some pre-defined *functional* features. We extend this work by providing the ability to automatically derive relevant *structural and functional features* from an explicitly given *purpose*.

IV Case Studies

A. Discussion

In this section we illustrate the technique by two case studies. Our case studies illustrate the problem of refining concepts of artifacts, by analogy, based on the specialized purpose for which these artifacts are to be used. Often when learning a concept, some notion of the concept is already known, and the task is to modify it slightly as it is used in a different context. To simplify our technique conceptually, we assume that the known purpose of the artifact, constrained by the specialized purpose for which it is intended, is the 'purpose' input to the system. For example, if the system is to learn the concept of 'vehicle' in the context of prohibiting vehicles from being driven in the park, we assume that in the known purpose of vehicles (to enable transportation), constrained by the context (interfering with park use) is the 'purpose' input to the system: i.e. vehicles that 'enable transportation but interfere with park use'.

B. Case Study 1: A Cup for Drinking Hot Liquids

In the case study described below, a system for performing Purpose-Directed Analogy takes as input the goal concept (HOT-CUP), its purpose (to enable an agent to drink hot liquids), a target example (a styrofoam cup), and domain theory (typical actions an agent can perform, a structural and functional model of the artifact). Then, by analogical reasoning to a known base example of a HOT-CUP (a ceramic mug), the system determines how the target example (styrofoam cup) is a member of the concept (HOT-CUP), derived from the explanation of how the base example is a member of the concept.

We now detail each step of the technique.

1. RETRIEVE step

Given the goal concept, this step retrieves a prototypical base example of the goal concept. Specifically, a prototypical example of a HOT-CUP (a ceramic mug) is retrieved. To simplify the problem, we assume that a prototypical example is known, and stored in such a way that it can be easily retrieved (as an instance of the general concept in an instance/class hierarchy).

2. EXPLAIN step

The next step uses the domain theory to explain how the base example (ceramic mug satisfies the purpose (to enable an agent to drink hot liquids) (see figure IV-1). This is the crux of Purpose-Directed Analogy: in this step the relevant explanation is automatically derived, given explicit purpose for which the artifact is used.

The explanation step consist of two parts: first, *derive a general explanation* of how an example can satisfy the purpose of the goal concept; second, *recognize* that features of the example in fact satisfy the explanation.

Derive a General Explanation: First, given the purpose of the goal concept and domain theory, a general explanation of how an example satisfies the purpose of the goal concept is derived. The purpose of HOT-CUP can be stated in a PROLOG-like representation as follows:

purpose(object,
  enable-action(object,Ingest(agent, hot-liquid, object)))
⇐ hot-cup(object)

In words: if something is a HOT-CUP, its purpose is to enable an agent to 'ingest' hot liquids. If the purpose of a HOT-CUP is to enable ingesting hot liquids, then an example of a HOT-CUP will satisfy the purpose if it enables a plan of action leading to the goal. A planner (as in [12]) generates a prototypical plan (or 'script') which leads to the goal action 'ingest': The plan is:

Putin(agent, liquid, object)

Keep(agent, liquid, object, time-interval)

Grasp(agent, object, liquid)

Pickup(agent, object, liquid)

Ingest(agent, liquid, object)

Each action has a list of preconditions which must be true in order to enable the action. The *object preconditions* are those preconditions which must be true of the *object* in order to enable the action. For an artifact to enable the plan of actions, it is expected to satisfy the object preconditions of each of the actions in the plan. The object preconditions are:

object preconditions for 'Putin':
  can(contain(object, contents))

object preconditions for 'Keep':
  can(keep(object, hot-liquid, time-interval))

object preconditions for 'Grasp':
  can(be-grasped-by(object, agent, hot-liquid))

object preconditions for 'Pickup':
  can(be-pickedup-by(object, agent, hot-liquid))

object preconditions for 'Ingest':
  can(be-ingested-with(object, agent, hot-liquid))

In general, the preconditions are collected together by a method such as goal regression [22] which collects only those preconditions not directly enabled by previous actions, and keeps track of other constraints among the preconditions.

The output of this first part is a general explanation of the preconditions which an example is expected to satisfy in order to fulfill the purpose of a HOT-CUP.

Recognize Example as Satisfying the Explanation: Given the general explanation, the base example, and domain theory, this step verifies that, in fact, the base example (a ceramic mug) satisfies the explanation for membership in HOT-CUP (closely related to plan recognition [23].) An artifact can satisfy these preconditions, or *functional requirements*, by certain structural characteristics. These can be satisfied, in turn, by particular attributes of the artifact.

The example is represented as a frame, with attributes represented by slots and values. A frame in a PROLOG-like representation is a list of binary predicates [12]. The frame describing ceramic-mug1 is:

manufacturer(ceramic-mug1, abc-co))
serial-number(ceramic-mug1, 72118))
color(ceramic-mug1, blue)
material(ceramic-mug1, ceramic)
weight(ceramic-mug1, 6-ounces)
has-part(ceramic-mug1, flat-bottom)
has-part(ceramic-mug1, open-cylinder)
has-part(ceramic-mug1, handle)

The domain theory contains 'default' rules, representing typical structural and functional characteristics of an artifact. An *enable structure* rule expresses how a general structural attribute can typically be satisfied by a particular attribute of an object. For instance:

structure(object, open-concavity) ⇐
  has-part(object,open-cylinder)

The *enable function* rule expresses how a functional requirement can typically be satisfied by a structural attribute. For example:

enable-function(object,
  can(contain(object, contents)) ⇐
  structure(object, open-concavity)

Recognition proceeds as a search for rules to generate a proof that attributes of the ceramic mug satisfy the functional requirements. The resulting explanation is as follows: (see Figure IV-1 for an illustration): Since the shape of the mug is an open cylinder, it has an open concavity which allows hot liquids to be PUTIN it (i.e. poured in). The ceramic material of the mug provides a non-porous material which also insulates the heat, and the flat shape of its bottom makes it stable--all which enable the cup to KEEP the hot liquid for some interval. Its handle and insulating material makes it GRASPable. Its weight (6 oz.) makes it light-weight, and that, along with its stability, enable an agent to PICK it UP. Finally, enabling all the previous actions, along with having an open concavity, allows the agent to perform his goal action of INGESTing the hot liquids from the ceramic mug.

The output of this step, then, is an explanation of how the base example (the ceramic mug) satisfies the purpose of HOT-CUP.

3. MAP Step

This step copies the explanation of the base example over to the target example (the styrofoam cup).

4. JUSTIFY Step

This step takes as input the explanation mapped over, the target example, and domain theory, and attempts to justify that the explanation is satisfied by the target example. If it cannot justify it using the explanation as its stands, it modifies the explanation to show that the target example is a member of the goal concept in a slightly different way.

First, it attempts to show how the attributes of the styrofoam cup satisfy the structural and functional requirements of something that is a HOT-CUP in the same way as the ceramic mug. If it fails to do that, it attempts to modify a portion of the explanation to show that the functional requirements are satisfied by alternative structural features. If it is unable to do that, it attempts to show that alternative actions satisfy the agent's goal action. If that is unsuccessful, the justification step fails. This processing is similar in spirit to derivational analogy [3], and partial provisional planning [24].

In this example, the styrofoam cup satisfies most of the structural and functional requirements in the same way as the ceramic mug. It differs structurally only in that it is the styrofoam material, not ceramic, which insulates the heat; and it is the conical shape, rather than a handle, which makes it graspable. Since the styrofoam cup also fits these relevant functional requirements and therefore the purpose, even if with different structural characteristics, it is considered analogous to the ceramic mug, and may also be classified as a HOT-CUP.

The result of this step is a (possibly modified) explanation of how the styrofoam cup satisfies the purposes of a HOT-CUP.

5. Learning

Given the two explanations as input, learning is achieved first by retaining the two explanations derived by the system. This provides the system with the ability to classify the target example as a member of the concept. Next, the system proceeds to form a generalization based on the explanations generated for two examples. Given these two explanations, the system can summarize the common structural characteristics (and when finding none in common--the functional ones) to form a sufficient definition of the goal concept. Thus the output of this step is a sufficient definition of a HOT-CUP: an object which can have an open concavity, can be made of nonporous, insulating material, can be stable, lightweight, and can be graspable. This sufficient definition can be used from now on to recognize examples of a HOT-CUP more easily, since it is described in more *operational* terms [20], i.e. in terms of structural, observable characteristics (see section V.A for further discussion of 'operationality').

C. Case Study 2: Vehicle in Park

We are also applying Purpose-Directed Analogy to a more complex case study, that of forming legal concepts by legal reasoning from precedents (initiated within the TAXMAN II project [16, 21]). (For other research on AI and legal reasoning see [9]). Given the legal statute "A vehicle is prohibited in a public park" [7], the task is to learn the concept DISTURBING-VEHICLE, an object which enables driving but interferes with park use. We do not present a detailed solution here. Rather, we sketch it briefly.

A case is brought before the court for violating the statute 'A vehicle is prohibited in a public park'. It is the case of Tommy, an 18-year old, who was found speeding through the park on a bicycle by a policeman. The system performing Purpose-Directed Analogy can be viewed as modelling the task of the prosecuting lawyer. The lawyer will argue that riding a bicycle in the park is analogous to a case where a passenger car was driven into the park, a clear example of a vehicle prohibited in the park. (This style of argumentation from precedents is a common form of legal argumentation.) The argument involves presenting the relevant facts that justify why for this law, the bicycle case is analogous to the case involving a passenger car. Knowledge of the purpose of the vehicles that the law intends to prohibit (DISTURBING-VEHICLES, objects which enable driving but interfere with park use) is used to derive the relevant explanation used in this analogy. The problem of learning the legal concept DISTURBING-VEHICLE by argumentation from precedents, guided by knowledge of legislative intent, more specifically the purpose of DISTURBING-VEHICLE, is thus an instance of the general problem of learning concepts by Purpose-Directed Analogy.

First, the system *retrieves* the clear precedent case (involving a passenger car). We assume that clear precedent cases are known, and can easily be retrieved. Next, the system *explains* why the precedent case has violated this law, and thus involves a DISTURBING-

VEHICLE. The statute's intent is to prohibit driving those vehicles which would interfere with people's use of the park, such as enjoying the serene setting, and the natural habitat provided by the park. Driving a passenger car clearly interferes with these aspects of the park: it makes noise, and thus disturbs the serene setting. It pollutes the air, and may trample the lawn and even small animals, and thus destroys the natural habitat. Next, the system *maps* this explanation to argue (*justify*) that the case involving a bicycle is a DISTURBING-VEHICLE in the same relevant respects. Because the bicycle trampled the lawn, flowers, and sped by park users, it similarly interfered with the serene setting and the natural habitat of the park, and is therefore analogous to the case of a passenger car in the aspects relevant for these purposes, and may also be classified as a DISTURBING-VEHICLE. Finally, the system generalizes to a sufficient definition of DISTURBING-VEHICLE, based on the explanations, so that future cases can be identified more easily as having violated the legal statute.

V Conclusion

A. Limitations and Future Research

We plan to complete the implementation of the system in the near future. In addition, we expect to experiment with the system using case studies of increasing complexity. We also plan to test the system on case studies for learning concepts with alternative purposes. Further, several major theoretical issues still need to be addressed before we have a robust Purpose-Directed Analogy technique.

Generalizing the technique to other domains: We provided an initial design of a technique to learn concepts of everyday artifacts. Can this technique be generalized to other domains such as those studied by Gentner, Burstein, and Winston? We use a very specific notion of purpose, the purpose for which an artifact is intended to be used. 'Purpose' in analogy can express many different intentions. It can refer to the purpose of the agent forming the analogy, the purpose of the agent understanding the analogy, the purpose of the analogy process itself, the purpose of the agent using the concept learned as a result of the analogy, and so on. One important open problem to be solved before the technique can be generalized is to represent classes of purposes, and their relationship to one another.

Deriving the Goal Concept: Currently, the system is given a single concept to learn, and a single purpose. Yet in most real-world forms of learning, there are many concepts to learn. In addition, there are many, sometimes conflicting, purposes for learning. Can a system arrive at the desired concept(s) to learn, and infer the purpose(s) automatically from context? In his research on *contextual learning*, Keller has demonstrated a scenario for this in the context of heuristic search [11]. An important issue for future research is to examine the problem of formulating concepts and purposes in these domains.

Adequacy of The Domain Theory: The explanations derived by Purpose-Directed Analogy are only as adequate as the underlying domain theory used. In the case studies presented here, the system had all the correct domain theory rules needed to derive the explanations. Yet theories of the domains of commonsense artifacts and law are both *inexact* and *information-incomplete*. In fact, the representation of most real-world domains will always be lacking. Learning in these domains will have to account for a weak underlying theory (see [15, 16] for one approach).

The domain theory is inexact in that the axioms represent what is only approximately true. For example, a rule such as 'has(vehicle,Motor) ⇒ pollutes(vehicle,Air)' is not infallible: what if the motor is dead? One issue to deal with is how to learn concepts when exceptions to these rules arise (see also [1]). Both analytic and empirical techniques that deal with exceptions will need to be developed ([28] is one such analytic technique). In addition, the theory in these domains is information-incomplete in that we cannot hope to represent all the needed information about these domains. For example, our representation might be missing the rule 'has-part(x, handle) ⇒ graspable(x)' needed to generate the explanation of how something is a cup. Thus the issue of when to approximate when generating an explanation will come into play. In the long term, techniques will need to be developed to learn these axioms from empirical techniques, or from reasoning from first principles.

Operationality of Definitions: Intuitively, we want our concept definitions to enable agents to easily recognize members of the concepts. When is a concept definition 'operational' in these domains? When is it 'non-operational' [20]? Currently we assume that structural definitions allow a human agent to easily classify examples as being members of a concept (hence are operational), while functional and purposive definitions do not (and hence are non-operational). The notion of concept operationalization has been investigated in [10]. Keller advocates defining operationality in terms of the intended use of the concept. The intended use in the case studies examined here seems to be one of classification. An interesting open issue to explore is whether knowledge of the type of domain (artifacts, say) and the intended use of a concept (classification, say) can be used to automatically define operational and non-operational languages for defining concepts in a given domain.

B. Summary

To summarize, we have outlined a framework of existing models of analogical reasoning, within which we discussed three existing models of concept learning by analogy. We argued that a key limitation is that these models cannot automatically generate the causal network of relations relevant for the purpose of a particular analogy. We then introduced an initial design for Purpose-Directed Analogy in concept learning, which addresses this limitation by using a specialized notion of purpose to automatically derive the relevant explanation (causal network). This specialized purpose is the

purpose for which an artifact is intended to be used. Given explicit knowledge of the purpose of the artifact, two examples are considered analogous by the system if they share an explanation which proves that both can be used for the same purpose. We illustrated the technique with two case studies of learning concepts of everyday artifacts.

Building a machine that learns by analogy and reasons in commonsense domains is still beyond our abilities, yet we are slowly progressing toward that goal.

VI Acknowledgments

Thanks go to Tom Mitchell, my thesis advisor, who strongly influenced the work and its presentation. I would also like to thank Thorne McCarty for his guidance. Rich Keller's thesis research influenced the goal and context-directed nature of this investigation. I would like to thank Jack Mostow for his thoughtful comments on the research and drafts of this paper. Thanks also go to Rich Keller, Sridhar Mahadevan, Mike Sims, Chuck Schmidt, Louis Steinberg, N.S. Sridharan, Saul Amarel, Prasad Tadepalli, Keith Williamson, and other Rutgers and GTE colleagues who provided useful suggestions during the course of the research, and helpful comments on drafts of this paper.

References

1. Borgida, A., Mitchell, T. and Williamson, K. E. Learning Improved Constraints and Schemas from Exceptions in Data and Knowledge Bases. In *On Knowledge Base Management Systems*, Brodie, M. L. and Mylopoulos, J., Eds., Springer Verlag, New York, NY, 1985. forthcoming.

2. Burstein, M. H. A Model of Learning by Incremental Analogical Reasoning and Debugging. Proceedings AAAI-83, Washington, D.C., August, 1983, pp. 45-48.

3. Carbonell, J. G. Derivational Analogy and Its role in Problem Solving. Proceedings AAAI-83, Washington, D.C., August, 1983, pp. 64-69.

4. DeJong, G. Acquiring Schemata Through Understanding and Generalizing Plans. Proceedings IJCAI-8, Karlsruhe, West Germany, August, 1983, pp. 462-464.

5. Gentner, D. "Structure Mapping: A Theoretical Framework for Analogy". *Cognitive Science 7*, 2 (April-June 1983), 155-170.

6. Hall, R. P. Analogical Reasoning in Artificial Intelligence and Related Disciplines. Department of Information and Computer Science,University of California, Irvine, February, 1985.

7. Hart, H. L. A. "Positivism and the Separation of Law and Morals". *Harvard Law Review 71* (1958), 593-629.

8. Kedar-Cabelli, S. Analogy - From a Unified Perspective. DCS-TR-146, Laboratory for Computer Science Research, Rutgers University, July, 1985. forthcoming.

9. Kedar-Cabelli, S. Analogy with Purpose in Legal Reasoning from Precedents. LRP-TR-17, Laboratory for Computer Science Research, Rutgers University, July, 1984.

10. Keller, R. M. Learning by Re-expressing Concepts for Efficient Recognition. Proceedings AAAI-83, Washington, D.C., August, 1983, pp. 182-186.

11. Keller, R. M. Sources of Contextual Knowledge for Concept Learning. Unpublished Thesis Proposal, July 1984, Rutgers University Department of Computer Science.

12. Kowalski, R.. *Logic for Problem Solving*. Elsevier North Holland, Inc., New York, NY, 1979.

13. Lebowitz, M. Concept Learning in a Rich Input Domain: Generalization-Based Memory. In *Machine Learning: An Artificial Intelligence Approach, Vol. 2*, Michalski, R. S., Carbonell, J. G., and Mitchell, T. M., Ed., Morgan Kaufmann, Los Altos, CA, 1985. forthcoming.

14. Mahadevan, S. Verification-Based Learning: A Generalization Strategy for Inferring Problem-Decomposition Methods. Proceedings IJCAI-9, Los Angeles, CA, August, 1985. .

15. McCarty, L. T. and Sridharan, N. S. The Representation of an Evolving System of Legal Concepts: II. Prototypes and Deformations. Proceedings IJCAI-7, Vancouver, B.C., Canada, August, 1981, pp. 246-253.

16. McCarty, L. T. and Sridharan, N. S. A Computational Theory of Legal Argument. LPR-TR-13, Laboratory for Computer Science Research, Rutgers University, January, 1982.

17. Minton, S. Constraint-Based Generalization: Learning Game-Playing Plans From Single Examples. Proceedings AAAI-84, Austin, TX, August, 1984, pp. 251-254.

18. Mitchell, T. M. Learning and Problem Solving. Proceedings IJCAI-8, Karlsruhe, West Germany, August, 1983, pp. 1139-1151.

19. Mitchell, T. M., Mahadevan, S. and Steinberg, L. LEAP: A Learning Apprentice for VLSI Design. Proceedings IJCAI-9, Los Angeles, CA, August, 1985.

20. Mostow, D. J. Machine Transformation of Advice into a Heuristic Search Procedure. In *Machine Learning*, Michalski, R. S., Carbonell, J. G. and Mitchell, T. M., Eds., Tioga, Palo Alto, CA, 1983.

21. Nagel, D. Concept Learning by Building and Applying Transformations Between Object Descriptions. LPR-TR-15, Laboratory for Computer Science Research, Rutgers University, June, 1983.

22. Nilsson, N. J.. *Principles of Artificial Intelligence*. Tioga, Palo Alto, CA, 1980.

23. Schmidt, C. F, Sridharan, N. S. and Goodson, J. L. "The Plan Recognition Problem: An Intersection of Psychology and Artificial Intelligence". *Artificial Intelligence 11* (1978), 45-83.

24. Schmidt, C. F. Partial Provisional Planning: Some Aspects of Commonsense Planning. In *Formal Theories of the Commonsense World*, Hobbs, J. and Moore, R., Eds., Ablex Publishing Co., Norwood, NJ, 1985, pp. 227-250.

25. Winston, P. H. Learning New Principles from Precedents and Exercises: The Details. M.I.T. AI Lab, May, 1981.

26. Winston, P. H. "Learning New Principles from Precedents and Exercises". *Artificial Intelligence 19*, 3 (November 1982), 321-350.

27. Winston, P. H., Binford, T. O., Katz, B., and Lowry, M. Learning Physical Descriptions from Functional Definitions, Examples, and Precedents. Proceedings AAAI-83, Washington, D.C., August, 1983, pp. 433-439.

28. Winston, P. H. Learning by Augmenting Rules and Accumulating Censors. In *Machine Learning: An Artificial Intelligence Approach, Vol. 2*, Michalski, R. S., Carbonell, J. G., and Mitchell, T. M., Ed., Morgan Kaufmann, Los Altos, CA, 1985. forthcoming.