Automatic Schema Acquisition in a Natural Language Environment

Gerald DeJong
Coordinated Science Laboratory
University of Illinois
Urbana, IL 61820

ABSTRACT

This paper outlines an approach to schema acquisition. The approach, called explanatory schema acquisition is applicable in problems solving situations and is heavily knowledge-based. Basically, learning is viewed as a fundamental part of the understanding process. Understanding a situation for which there is no existing schema involves generalizing the new event into a nascent schema. The new schema is then available to aid in future processing and can be further refined via that processing. This approach to learning is unique in several respects: it is not inductive and so is capable of one trial learning, it does not depend on failures to drive the learning process, and it is incremental and learns comparitively slowly. The learning procedure is outlined briefly with an example, a taxonomy of situations involving explanatory schema acquisition is given, and there is a brief discussion on the scope of the learning mechanism.

1. Introduction

The concept of knowledge chunks, variously termed schemas, scripts, frames or MOPs has emerged to organize world knowledge in artificial intelligence systems. They have been used to understand natural language, metaphor processing, memory organization, story summarizing, and planning. Yet there has been little work on how these constructs are acquired; most systems simply "build in" the requisite knowledge structures.

Now that we have had some experience with this knowledge representation form, it is appropriate to ask how these knowledge structures might be acquired by AI systems automatically. In the remainder of the paper I will use the least contentious term "schema" to refer to these knowledge chunks.

There are many reasons why learning should be a part of a natural language system. On the theoretical side, the ability to benefit from experience is a necessary and indeed a defining characteristic of intelligence. More practically, the ability to learn would be of great advantage to a system that deals with real world input texts. This was made apparent from experience with the FRUMP system [1]. While it was a very successful program, its weakest link was its limited number of schemas. Each of its schema structures (called sketch scripts) had to be added by hand.

This work was supported in part by the National Science Foundation under grant NSF IST 81-20254

2. Overview

Schemas are used in natural language processing to supply missing inferences to connect the input propositions logically. A text is input to the system. In addition, some systems (e.g. FRUMP) also used schemas to aid in word and sentence interpretation.

What can a natural language system do if it does not have an appropriate schema for understanding a new input text? In certain circumstances, the system can process the input anyway and in doing so acquire the missing schema. Most schemas are themselves composed of other schemas. If a text describes a new situation on the level of these constituent schemas then a natural language understander can process the input by relying on general goal and planning background knowledge. This is not contentious. Most planning-type natural language systems do precisely this (e.g., [7] [9]).

It is important to notice that the story representation arrived at though planning can be viewed as a schema itself, albeit a poor, narrow, and over-constrained one. That is, the process of re-reading this particular text would be immensely simplified if the system had access to the previously constructed representation. The predictions made by this "overly-specialized schema" would be exactly on target. The problem, of course, is that any modification, no matter how slight, of the story would directly contradict the new schema, making it inapplicable to the modified text.

The important insight, and the basis for explanatory schema acquisition, is that the "overly-specialized schema" for an event can be generalized by a knowledge-based system to be a plausible and useful schema. Moreover, the knowledge required for this generalization process is precisely the knowledge that is used by and present in planning systems. See [2] for a more complete overview and an example. Two large problems must be addressed. First, when is the generalization process invoked? and how does generalization occur?

3. Situations that Invoke Explanatory Schema Acquisition

There are four situations which when recognized in the text either individually or in combination ought to invoke the generalization routines. They are:

1) Schema Composition
2) Secondary Effect Elevation
3) Schema Alteration
4) Volitionalization

3.1 Schema Composition

Basically, schema composition involves connecting known schemas in a novel way. Typically, this will involve a primary schema, essentially unchanged, with one or more of its preconditions satisfied in a novel way by other known schemas.

For example, a RANSOM schema is a combination of THEFT and BARGAIN. The primary schema is BARGAIN. It is used in the normal way (i.e., to obtain something of value by trading something one values less). The THEFT schema satisfies the precondition of possessing the thing that one plans to trade away. Thus, THEFT is used, somewhat peculiarly, to obtain an object that is itself not necessarily valued by the thief.

3.2 Secondary Effect Elevation

Secondary effect elevation involves acquiring a new problem solving construct (schema) which is nearly the same as an existing schema but whose main effect is only a side effect in the original schema. Consider the following scenario in which Fred uses secondary effect elevation to acquire a schema to solve his problem.

> Fred wanted to date only Sue, but Sue steadfastly refused his overtures. Fred was on the verge of giving up when he saw what happened to his friend, John: John wanted to date Mary but she also refused. John started seeing Wilma. Mary became jealous and the next time he asked her, Mary eagerly accepted. Fred told Sue that he was going to make a date with Lisa.

Here Fred has has used an existing schema (DATE) in a new way. The main purpose of the DATE schema is to satisfy certain recurring social goals (like companionship, sex, etc.). DATE contains secondary effects as well. These are often undesirable effects accompanying the main, planned effects. For example, one is usually monetarily poorer after a date. Another secondary effect is that if one has an old girl friend, she may become jealous of a new date.

What Fred learned from observing John's experience is that it is occasionally useful to invoke the DATE schema in order to cause one of its secondary effects (jealousy) while completely ignoring the usual main goal.

3.3 Schema Alteration

Schema alteration involves modifying a nearly correct schema so that it fits the requirements of a new situation. The alteration process is guided by the system's world model. This is illustrated by the following brief anecdote:

> Recently I had occasion to replace temporarily a broken window in my back door with a plywood panel. The plywood sheet from which the panel was to be cut had a "good" side and a "bad" side (as does most raw lumber). The good side was reasonably smooth while the bad side had several ruts and knot holes. I automatically examined both sides of the sheet (presumably as part of my SAWING or CUTTING-A-BOARD-TO-FIT schema) and selected the good side to face into the house with the bad side to be exposed to the elements. After I had cut the panel and fitted it in place I noticed that several splinters had been torn out leaving ruts in the "good" side.

I immediately saw the problem. Hand saws only cut in one direction. With hand saws, the downward motion does the cutting while the upward motion only repositions the cutting blade for another downward motion. I had cut the wood panel with the "good" side facing down. The downward cutting action has a tendency to tear splinters of wood out of the lower surface of the board. This is not a problem on the upper surface because that wood is supported from below by other wood. Since the good side was the lower surface, it suffered the loss of splinters. If I had to perform the same action again, I would not make the same mistake. I would cut the board with the good side facing up. However, what I learned was not just a simple specialized patch to handle this particular instance of splintering. Since I knew the cause of the splintering, I knew that it would not always be a problem: it is only a problem when 1) the lumber is prone to splintering, 2) there is a "good" side of the board that is to be preserved, and 3) one is making a crosscut (across the wood's grain) rather than a rip cut (along the grain). Moreover, the solution is not always to position the wood with the good side up. My electric saber saw (also a reciprocating saw) cuts during the upward blade motion rather than the downward motion. Clearly, the solution when using the saber saw is the opposite: to position the board with the good side down. Now, these are not hard and fast rules: with a sufficiently poor quality sheet of plywood splintering would likely always be a problem. Rather, these are useful heuristics that lead to a refinement of the SAWING schema.

3.4 Volitionalization

As the name implies this situation involves transforming a schema for which there is no planner (like VEHICLE-ACCIDENT, ROULETTE, etc.) into a schema which can be used be a planner to attain a specific goal. Consider the following story:

> Herman was his grandfather's only living relative. When Herman's business was failing he decided to ask his grandfather for a loan. They had never been close but his grandfather was a rich man and Herman knew he could spare the money. When his grandfather refused, Herman decided he would do the old fellow in. He gave him a vintage bottle of wine spiked with arsenic. His grandfather died. Herman inherited several million dollars and lived happily ever after.

This story is a paraphrase of innumerable mystery stories and illustrates a schema familiar to all who-done-it readers. It might be called the HEIR-ELIMINATES-BENEFACTOR schema. It is derivable via volitionalization by modifying the existing non-volitional schema INHERIT. INHERIT is non-volitional since there is no active agent. The schema simply dictates what happens to a persons possessions when he dies.

In this guise volitionalization parallels schema composition. One of the preconditions to INHERIT is that the individual be dead. The ELIMINATE-BENEFACTOR schema uses the schema MURDER to accomplish this. One major difference is that schema composition requires all volitional schemas. This parallelism need not always be present, however. Non-volitional to volitional transformation is also applicable to removing stochastic causal steps from a schema resulting in a volitional one.

4. The Generalization Process

The generalization process is based on certain data dependency links established during understanding. After a story is understood, the understood representation can be viewed as an <u>explanation</u> of why the events are plausible. For example, take the case of a kidnapping. KIDNAP is an instance of schema composition, not unlike RANSOM. Thus, the first kidnapping story seen by the system is understood as a THEFT followed by a BARGAIN. If the kidnapper is successful, the ransom is paid. For a system to understand this, it must justify that the person paying values the safety of the kidnapped victim more that the ransom money. This justification is a data dependency [3] link to some general world knowledge (e.g., that a parent loves his children). Now the event can be generalized so long as these data dependency links are preserved. Clearly, as long as the data dependencies are preserved, the underlying events will still form a believable whole.

Consider again the secondary effect elevation example of Fred trying to date Sue. The observed specific instance is John's interactions with Mary. Notice, however, that Fred did not simply copy John's actions. John actually made a date with Wilma while Fred only expressed an intention to date Lisa. This is not an earth-shaking difference, but in the context of dating it is extremely significant. In the normal DATE situation expressing an intention to date someone is not nearly so satisfying as an actual date. Once modified for the purpose of causing jealousy, however, expressing an intention for a date and actually carrying it out can be equally effective. That is, they both maintain the data dependency link for why we believe that Sue is in fact jealous.

5. Conclusion

There are several concluding points

1) Unlike most learning systems explanatory schema acquisition does not depend on correlational evidence. Thus, it is capable of one trial learning. It is somewhat similar to Soloway's view of learning [8].

2) The approach is heavily knowledge-based. A great deal of background knowledge must be present for learning to take place. In this respect explanatory schema acquisition follows the current trend in AI learning and discovery systems perhaps traceable to Lenat [5].

3) The learning mechanism is not "failure-driven" as is the MOPs approach [6]. In that view learning takes place in response to incorrect predictions by the system. In explanatory acquisition learning can also be stimulated by positive inputs which encounter no particular problems or prediction failures.

4) The absolute representation power of the system is not enhanced by learning new schemas. This statement is only superficially surprising. Indeed, Fodor [4] implies that this must be true of all self-consistent learning systems. Explanatory schema acquisition does, however, increase processing efficiency. Since all real-world systems are resource limited, this learning technique does, in fact, increase the system's processing power. Furthermore, it may indicate how Socratic method learning is possible and why the psychological phenomenon of functional fixedness is adaptive.

ACKNOWLEDGEMENT

I am indebted to Paul O'Rorke for several discussions on explanatory schema acquisition.

REFERENCES

[1] DeJong, G. (1979) Skimming stories in real time: an experiment in integrated understanding. Research Report 158, Yale Computer Science Department, New Haven, CT.

[2] DeJong, G. (1979) Prediction and substantiation: a new approach to natural language processing. <u>Cognitive Science</u> 3, 251-273.

[3] Doyle, J. (1978) Truth maintenance systmes for problem solving. MIT AI Technical report TR-419, MIT, Cambridge, MA

[4] Fodor, J. (1975) <u>The Language of Thought</u> Thomas Y. Crowell Company, New York

[5] Lenat, D., (1976) AM: an artificial intelligence approach to discovery in mathematics as heuristic search. AIM-286, AI Laboratory, Stanford University, Stanford, CA.

[6] Schank, R., (1980) Language and memory. Cognitive Science 4, 243-283.

[7] Schmidt, C. and Sridharan, N. (1977) Plan recognition using a hypothesize and revise paradigm: an example. Proceedings of the Fifth International Joint Conference on Artificial Intelligence, pp. 480-486.

[8] Soloway, E., (1977) Knowledge directed learning using multiple levels of description. Ph. D. dissertation, Computer Science Department, University of Massachusetts, Amherst, MA.

[9] Wilenski, R. (1978) Understanding goal-base stories, Ph.D. dissertation, Yale Computer Science Report 140, September, Yale University, New Haven, CT.