Rules and Rules Chaining
—
Winston, Chapter 7

Michael Eisenberg and Gerhard Fischer
TA: Ann Eisenberg

AI Course, Fall 1997

Overview

rules:

- “if-then” rules
- production systems

forward chaining

- toy example: identify zoo animals
- influential, classic system: R1, Xcon ---> configures computer systems

backward chaining

- toy example: bag groceries
- influential, classic system: Mycin ----> diagnoses diseases

ideas and principles that support many **useful applications** of AI (e.g., expert systems)

Definitions of Concepts

assertion:

- a statement that something is true
- example: “Stretch is a giraffe” or “Stretch has long legs”

working memory:

- collection of assertions

deduction systems:

- “then” patterns specify new assertions to be placed into working memory
- antecedent: “if” pattern
- consequent: “then” pattern

reaction system:

- “then” patterns specify actions
- example: “put the item into the bag”

Forward Chaining and Backward Chaining

Forward Chaining:

- process of moving from the “if” patterns to the “then” patterns
- whenever an “if” pattern is observed to match an assertion: the antecedent is satisfied
- whenever all the “if” patterns of a rule are satisfied ---> the rule is triggered
- a triggered rule establishes a new assertion ---> it is fired

Backward Chaining:

- form a hypothesis
- use the antecedent-consequent rules to work backward towards hypothesis-supporting assertions

Example: ZOOKEEPER — Identify Animals

Z1 If ?x has hair
then ?x is a mammal

Z2 If ?x gives milk
then ?x is a mammal

Z3 If ?x has feathers
then ?x is a bird

Z4 If ?x flies
?x lays eggs
then ?x is a bird

Z5 If ?x is a mammal
?x eats meat
then ?x is a carnivore

Example: ZOOKEEPER — Continued

Z6 If ?x is a mammal
?x has pointed teeth
?x has claws
?x has forward-pointing eyes
then?x is a carnivore

Z7 If ?x is a mammal
?x has hoofs
then?x is an ungulate

Z8 If ?x is a mammal
?x chews cud
then?x is an ungulate

Z9 if ?x is a. carnivore
?x has tawny color
?x has dark spots
then?x is a cheetah

Example: ZOOKEEPER — Continued

Z10 If ?x is a carnivore
?x has tawny color
?x has black strips
then?x is a tiger

Z11 If ?x is an ungulate
?x has long legs
?x has long neck
?x has tawny color
?x has dark spots
then?x is a giraffe

Z12 If ?x is an ungulate
?x has white color
?x has black stripes
then?x is a zebra

Example: ZOOKEEPER — Continued

Z13 If ?x is a bird
?x does not fly
?x has long legs
?x has long neck
?x is black and white
then ?x is an ostrich

Z14 If ?x is a bird
?x does not fly
?x swims
?x is black and white
then ?x is a penguin

Z15 If ?x is a bird
?x is a good flyer
then ?x is an albatross

Analyze an Unknown Animal

working memory:

- Stretch has hair.
- Stretch chews cud.
- Stretch has long legs.
- Stretch has a long neck.
- Stretch has tawny color.
- Stretch has dark spots.

Analysis:

- Stretch has hair ---> Z1 fires ----> mammal
- Stretch is a mammal and chews cud ---> Z8 fires ----> ungulate
- all antecedents from Z11 are satisfied ---> Z11 fires ----> giraffe

Working Memory

A working memory is a representation in which

- Lexically, there are assertions and application-specific symbols. There are also patterns that contain application-specific symbols mixed with pattern symbols.
- Structurally, the assertions are lists of application-specific symbols.
- Semantically, the assertions denote facts in some world.

With constructors that

- Add an assertion to working memory

With readers that

- Produce a list of the matching assertions in working memory, given a pattern

Rule Base

A rule base is a representation in which

- there is a working memory.
- Lexically, there are rules.
- Structurally, the rules consist of patterns. Some of these patterns constitute the rule's if patterns; the others constitute the rule's then pattern.
- Semantically, rules denote constraints that enable procedures to seek new assertions or to validate a hypothesis.

With constructors that

- Construct a rule, given an ordered list of if patterns and a then pattern

With readers that

- Produce a list of a given rule's “if” patterns
- Produce a list of a given rule's “then” patterns

ZOOKEEPER (forward-chaining version)

To identify an animal with ZOOKEEPER

- Until no rule produces a new assertion or the animal is identified,
  - For each rule,
    - Try to support each of the rule's antecedents by matching it to known facts.
    - If all the rule's antecedents are supported, assert the consequent unless there is an identical assertion already.
    - Repeat for all matching and instantiation alternatives.

ZOOKEEPER (backward-chaining version)

ZOOKEEPER forms the hypothesis that Swifty is a cheetah. To verify the hypothesis, ZOOKEEPER considers rule Z9, which requires that Swifty is a carnivore, that Swifty has a tawny color, and that Swifty has dark spots.

ZOOKEEPER must check whether Swifty is a carnivore. Two rules may do the job, namely rule Z5 and rule Z6. Assume that ZOOKEEPER tries rule Z5 first.

ZOOKEEPER must check whether Swifty is a mammal. Again, there are two possibilities, rule Z1 and rule Z2. Assume that ZOOKEEPER tries rule Z1 first. According to that rule, Swifty is a mammal if Swifty has hair.

ZOOKEEPER must check whether Swifty has hair. Assume ZOOKEEPER already knows that Swifty has hair. So Swifty must be a mammal, and ZOOKEEPER can go back to working on rule Z5.

ZOOKEEPER must check whether Swifty eats meat. Assume ZOOKEEPER cannot tell at the moment. ZOOKEEPER therefore must abandon rule Z5 and try to use rule Z6 to establish that Swifty is a carnivore.

ZOOKEEPER must check whether Swifty is a mammal. Swifty is a mammal, because this was already established when trying to satisfy the antecedents in rule Z5.

ZOOKEEPER must check whether Swifty has pointed teeth, has claws, and has forward-pointing eyes. Assume ZOOKEEPER knows that Swifty has all these features. Evidently, Swifty is a carnivore, so ZOOKEEPER can return to rule Z9, which started everything done so far.

Now ZOOKEEPER must check whether Swifty has a tawny color and dark spots. Assume ZOOKEEPER knows that Swifty has both features. Rule Z9 thus supports the original hypothesis that Swifty is a cheetah, and ZOOKEEPER therefore concludes that Swifty is a cheetah.

ZOOKEEPER (backward-chaining version)

To identify an animal with ZOOKEEPER

- Until all hypotheses have been tried and none have been supported or until the animal is identified,
  - For each hypothesis,
    - For each rule whose consequent matches the current hypothesis,
      - Try to support each of the rule's antecedents by matching it to assertions in working memory or by backward chaining through another rule, creating new hypotheses. Be sure to check all matching and instantiation alternatives.
      - If all the rule's antecedents are supported, announce success and conclude that the hypothesis is true.

Rule-Based Reaction Systems

B1 If step is check-order
potato chips are to be bagged
there is no Pepsi to be bagged
then ask the customer whether he would like a bottle of Pepsi

B2 If step is check-order
then step is no longer check-order
step is bag-large-items

B2 (add-delete form)
If step is check-order
delete step is check-order
add step is bag-large-items

Rule-Based Reaction Systems — Continued

B3	If	step is bag-large-items
		a large item is to be bagged
		the large item is a bottle
		the current bag contains < 6 large items
	delete	the large item is to be bagged
	add	the large item is in the current bag

B4	if	step is bag-large-items
		a large item is to be bagged
		the current bag contains < 6 large items
	delete	the large item is to be bagged
	add	the large item is in the current bag

Conflict Resolution Strategies

- **Rule Ordering.** Arrange all rules in one long prioritized list. Use the triggered rule that has the highest priority. Ignore the others.

- **Context limiting.** Reduce the likelihood of conflict by separating the rules into groups, only some of which are active at any time.

- **Specificity ordering.** Whenever the conditions of one triggering rule are a superset of the conditions of another triggering rule, use the superset rule on the ground that deals with more specific situations.

- **Data ordering.** Arrange all possible assertions in one long prioritized list. Use the triggered rule that has the condition pattern that matches the highest priority assertion in the list.

- **Size ordering.** Use the triggered rule with the toughest requirements, where toughest means the longest list of conditions.

- **Recency ordering**. Use the least recently used rule.

Forward Chaining (detailed version)

To forward chain

- until no rule produces a new assertion,
  - For each rule,
    - try to match the first antecedent with an existing assertion. Create a new binding set with variable bindings established by the match.
    - Using the existing variable bindings, try to match the next antecedent with an existing assertion. If any new variables appear in this antecedent, augment the existing variable bindings.
    - Repeat the previous step for each antecedent, accumulating variable bindings as you go, until,
      - There is no match with any existing assertion using the binding set established so far. In this case, back up to a previous match of an antecedent to an assertion, looking for an alternative match that produces an alternative, workable binding set.
    - There are no more antecedents to be matched. In this case,
      - use the binding set in hand to instantiate the consequent.
      - Determine if the instantiated consequent is already asserted. If not, assert it.
      - Back up to the most recent match with unexplored bindings, looking for an alternative match that produces a workable binding set.
    - There are no more alternatives matches to be explored at any level.

Summary

- Rule-based systems were developed to take advantage of the fact that a great deal of useful knowledge can be expressed in simple if-then rules.
- Many rule-based systems are deduction systems. In these systems, rules consist of antecedents and consequents. In one example, a toy deduction system identifies animals.
- Deduction systems may chain together rules in forward direction, from assertions to conclusions, or backward, from hypotheses to questions. Whether chaining should be forward or backward depends on the problem.
- Many rule-based systems are reaction systems. In these systems, rules consist of conditions and actions. A toy reaction system bags groceries.
- Reaction systems require conflict-resolution strategies to determine which of many triggered rules should be allow to fire.
- Depth-first search can supply compatible bindings for both forward chaining and backward chaining.