20IS603 Architecture of Intelligent Systems

Handling Uncertainty in Expert Systems

Certainty Theory

Lecture #6

Certainty Theory

- An alternative to Bayesian reasoning - an attempt to overcome some of the shortcomings of Bayesian updating
- An attempt to formalize the heuristic approach to reasoning with uncertainty – NOT probabilities.
- Instead of using probabilities, each assertion has a certainty value associated with it.
- For a given hypothesis H, its certainty value C(H) is given by

$C(H) = 1.0$ if H is known to be true;
$C(H) = 0.0$ if H is unknown;
$C(H) = -1.0$ if H is known to be false.

Certainty Theory

- Each rule also has a certainty associated with it, known as its certainty factor, CF.
- Certainty factors serve a similar role to the affirms and denies weightings in Bayesian systems.
- The maximum value of the certainty factor was +1.0 (definitely true) and the minimum -1.0 (definitely false).
- A positive value represented a degree of belief and a negative a degree of disbelief.
- Identical measures of certainty are attached to rules and hypotheses.

```
uncertainty_rule
    if <evidence>
    then <hypothesis>
    with certainty factor <CF>.
```

Certainty Theory

- The certainty factor of a rule is modified to reflect the level of certainty of the evidence, such that the modified certainty factor CF′ is given by

$$CF' = CF \times C(E)$$

- If the evidence is known to be present, that is, C(E) = 1, then

$$CF' = CF$$

- The technique for updating the certainty of hypothesis H, in the light of evidence E, C(H|E), involves the application of the following composite function:

$$\text{if } C(H) \geq 0 \text{ and } CF' \geq 0 \text{ then } C(H|E) = C(H) + [CF' \times (1 - C(H))]$$

$$\text{if } C(H) \leq 0 \text{ and } CF' \leq 0 \text{ then } C(H|E) = C(H) + [CF' \times (1 + C(H))]$$

- if C(H) and CF′ have opposite signs then

$$C(H|E) = \frac{C(H) + CF'}{1 - \min(|C(H)|, |CF'|)}$$

Certainty Theory

- In certainty theory, a rule can only be applied if the certainty of the evidence C(E) is greater than 0, i.e., if the evidence is more likely to be present than not.

Properties of updating certainty values:

- Function is continuous and has no singularities or steps
- The updated certainty C(H|E) always lies within the bounds −1 and +1.
- If either C(H) or CF′ is +1 (i.e., definitely true), then C(H|E) is also +1.
- If either C(H) or CF′ is −1 (i.e., definitely false), then C(H|E) is also −1.
- When contradictory conclusions are combined, they tend to cancel each other out, that is, if C(H) = − CF′, then C(H|E) = 0.
- Several pieces of independent evidence can be combined by repeated application of the function, and outcome is independent of the order in which evidence are applied.
- If C(H) = 0, that is, the certainty of H is at its a priori value, then C(H|E) = CF′
- If the evidence is certain i.e., C(E) = 1, then CF′ = CF.
- Absence of evidence can be taken into account by allowing rules to fire when C(E) < 0

Certainty Theory

Logical Combinations of Evidence

- In Bayesian updating systems, each piece of evidence that contributes toward a hypothesis is assumed to be independent and is given its own affirms and denies weights.
- In certainty theory based systems, the certainty factor is associated with the rule as a whole rather than with individual pieces of evidence.
- For this reason, certainty theory provides a simple algorithm for determining the value of the certainty factor that should be applied when more than one item of evidence is included in a single rule.
- The algorithm covers the cases where evidence is conjoined (i.e.,joined by *and*), disjoined (i.e., joined by *or*), and negated (using *not*).

Certainty Theory

Logical Combinations of Evidence

- Conjunction

```
uncertainty_rule
    if <evidence E1>
    and <evidence E2>
    then <hypothesis>
    with certainty factor <CF>
```

The certainty of the combined evidence is given by $C(E_1 \textit{ and } E_2)$

$$C(E_1 \textit{ and } E_2) = \min[C(E_1), C(E_2)]$$

Certainty Theory

Logical Combinations of Evidence

- Disjunction

```
uncertainty_rule
    if <evidence E1>
    or <evidence E2>
    then <hypothesis>
    with certainty factor <CF>
```

The certainty of the combined evidence is given by $C(E_1\ \mathit{or}\ E_2)$

$$C(E_1\ \mathit{or}\ E_2) = \max[C(E_1), C(E_2)]$$

Certainty Theory

Logical Combinations of Evidence

- Negation

```
uncertainty_rule
    if not <evidence E>
    then <hypothesis>
    with certainty factor <CF>
```

The certainty of the negated evidence, C(E), is given by C(~E)

$$C(\sim E) = -C(E)$$

Example #1

- Control of a power station boiler

```
uncertainty_rule r3_1c
  if release_valve is stuck
  then task becomes clean_release_valve
  with certainty factor 1.0.

uncertainty_rule r3_2c
  if warning_light is on
  then release_valve becomes stuck
  with certainty factor 0.2.

uncertainty_rule r3_3c
  if pressure is high
  then release_valve becomes stuck
  with certainty factor 0.9.

uncertainty_rule r3_4c
  if temperature is high
  and water_level is not low
  then pressure becomes high
  with certainty factor 0.5.
```

Example #1

- set of input data

```
water_level is not low.
warning light is on.
temperature is high.
```

assume that the rules fire in the following order:

$$r3\_4c \rightarrow r3\_3c \rightarrow r3\_2c \rightarrow r3\_1c$$

Example #1

uncertainty_rule r3_4c — CF = 0.5

H = pressure is high; — C(H) = 0

E_1 = temperature is high; — $C(E_1) = 1$

E_2 = water_level is low; — $C(E_2) = -1$, $C(\sim E_2) = 1$

$C(E_1 \& \sim E_2) = \min[C(E_1), C(\sim E_2)] = 1$

$CF' = CF \times C(E_1 \& \sim E_2) = CF$

$C(H | (E_1 \& \sim E_2)) = CF' = 0.5$

/* Updated certainty of "pressure is high" is 0.5 */

uncertainty_rule r3_3c — CF = 0.9

H = release_valve is stuck; — C(H) = 0

E = pressure is high; — C(E) = 0.5

$CF' = CF \times C(E) = 0.45$

$C(H | (E)) = CF' = 0.45$

/* Updated certainty of "release_valve is stuck" is 0.45

uncertainty_rule r3_2c — CF = 0.2

H = release_valve is stuck; — C(H) = 0.45

E = warning_light is on; — C(E) = 1

$CF' = CF \times C(E) = CF$

$C(H | (E)) = C(H) + [CF' \times (1 - C(H))] = 0.56$

/* Updated certainty of "release_valve is stuck" is 0.56

uncertainty_rule r3_1c — CF = 1

H = task is clean_release_valve — C(H) = 0

E = release_valve is stuck; — C(E) = 0.56

$CF' = CF \times C(E) = 0.56$

$C(H | (E)) = CF' = 0.56$

/* Updated certainty of "task is clean_release_valve" is 0.56 */