



TRÍ TUỆ NHÂN TẠO

Bài 9: Tri thức và Lập luận

Nội dung



1. Các khái niệm cơ bản
2. Logic mệnh đề
3. Cơ sở tri thức
4. Suy diễn
5. Chứng minh bằng bác bỏ
6. Suy diễn lùi



Phần 1

Các khái niệm cơ bản

- **Ngữ nghĩa:**
 - Thành phần cơ bản nhất, cung cấp các ý nghĩa cho một hệ thống tri thức
 - Có thể có nhiều loại:
 - Cơ bản: không thể định nghĩa thông qua các ngữ nghĩa khác
 - Hệ quả: suy ra từ các ngữ nghĩa khác thông qua quá trình lập luận
 - Có thể chia theo mức độ phản ảnh chân lý:
 - Khẳng định: Đúng/Sai
 - Xác suất: Đúng theo tỉ lệ nào đó
 - Mờ
- **Cú pháp:** quy tắc liên hệ các kí hiệu, giúp cho việc xây dựng những ngữ nghĩa mới từ ngữ nghĩa đã có

- Cơ chế lập luận:
 - Quá trình tính toán (thuật toán)
 - Sử dụng ngữ nghĩa và cú pháp để tạo ra tri thức mới
- Quan điểm về AI:
 - Bài toán: Tập ngữ nghĩa và cú pháp cho trước
 - Giải bài toán: Tìm quá trình áp dụng cú pháp trên các ngữ nghĩa để ra được mục tiêu (cũng là một ngữ nghĩa)
- Đây là quan điểm theo trường phái “suy nghĩ hợp lý”
- Các phương pháp của trường phái này chịu ảnh hưởng lớn từ việc nghiên cứu ngôn ngữ hình thức (đầu TK20)
- Gần như mỗi môn khoa học cũng là một hệ thống tri thức và suy diễn



Phần 2

Logic mệnh đề

- Ngữ nghĩa: tập các mệnh đề (các khẳng định hoặc phủ định) và các kí hiệu logic
 - Hằng số: True / False
 - Biến: P, Q,...
 - Phép toán logic: \wedge (và) \vee (hoặc) \neg (phủ định) \Rightarrow (kéo theo) \Leftrightarrow (tương đương)
 - Các cặp ngoặc tròn () dùng kết hợp để thay đổi thứ tự tính toán

- Cú pháp:
 - Các biến mệnh đề là các công thức
 - Nếu A và B là công thức thì:
 - $(A \wedge B)$
 - $(A \vee B)$
 - $(\neg A)$
 - $(A \Rightarrow B)$
 - $(A \Leftrightarrow B)$
- cũng là các công thức

Logic mệnh đề



A	B	$A \wedge B$	$A \vee B$	$\neg A$	$A \Rightarrow B$	$A \Leftrightarrow B$
False	False	False	False	True	True	True
False	True	False	True	True	True	False
True	False	False	True	False	False	False
True	True	True	True	False	True	True

- Dạng chuẩn tắc:
 - Chuẩn tắc tuyển (hoặc)
 - Chuẩn tắc hội (và)
- Định lý: mọi công thức đều có dạng chuẩn tắc tuyển và chuẩn tắc hội
- Quy tắc:
 - Khử phép kéo theo
 - Khử phép tương đương



Phần 3

Cơ sở tri thức

- Hệ thống các tri thức ở 2 dạng:
 - Dữ liệu
 - Các sự kiện (hằng)
 - Các biến
 - Các hàm (logic tân từ)
 - Luật
 - Để ở dạng suy dẫn $P \Rightarrow Q$
- Thuật toán suy diễn: Nhiều chiến lược

- Cơ sở tri thức sử dụng logic mệnh đề:
 - Dạng cơ sở tri thức đơn giản nhất
 - Dữ liệu ban đầu: chỉ gồm các biến mệnh đề
 - Dữ liệu sự kiện: bổ sung các hằng
 - Luật: dạng suy dẫn đơn
- Động cơ (engine) suy diễn:
 - Ưu tiên chiều rộng
 - Ưu tiên chiều sâu

■ Ví dụ:

■ Tập luật:

- Nếu X tận cùng là 0, 2, 4, 6, 8 thì X chia hết cho 2
- Nếu X tận cùng là 0, 5 thì X chia hết cho 5
- Nếu X chia hết cho 2 và X chia hết cho 5 thì X chia hết cho 10

■ Sự kiện: $X = 100$, hỏi X có chia hết cho 10 không?

- Mã hóa:
 - Định nghĩa các biến mệnh đề:
 - P: X tận cùng là 0
 - Q: X tận cùng là 2
 - A: X tận cùng là 4
 - B: X tận cùng là 6
 - C: X tận cùng là 8
 - D: X tận cùng là 5
 - E: X chia hết cho 2

- Mã hóa:
 - Định nghĩa các biến mệnh đề (tiếp):
 - F: X chia hết cho 5
 - G: X chia hết cho 10
 - Tập luật:
 - $P \vee Q \vee A \vee B \vee C \Rightarrow E (*)$
 - $P \vee D \Rightarrow F (*)$
 - $E \wedge F \Rightarrow G$
 - Câu hỏi: Cho P, hỏi G



Phần 4

Suy diễn

- Sử dụng các hằng dữ kiện, tiến hành suy diễn từ các vế trái để nhận các giá trị vế phải, bổ sung vào kho dữ kiện & tiếp tục quá trình suy dẫn
- Ví dụ: Từ P ban đầu
 - Áp dụng: $P \vee Q \vee A \vee B \vee C \Rightarrow E$ thu được E
 - Áp dụng: $P \vee D \Rightarrow F$ thu được F
 - Áp dụng: $E \wedge F \Rightarrow G$

Thuật toán



T: Tập sự kiện ban đầu

R: Tập luật

D: Tập kết luận

do {

 L = {}

 for r in R do

 if (left(r) in T)

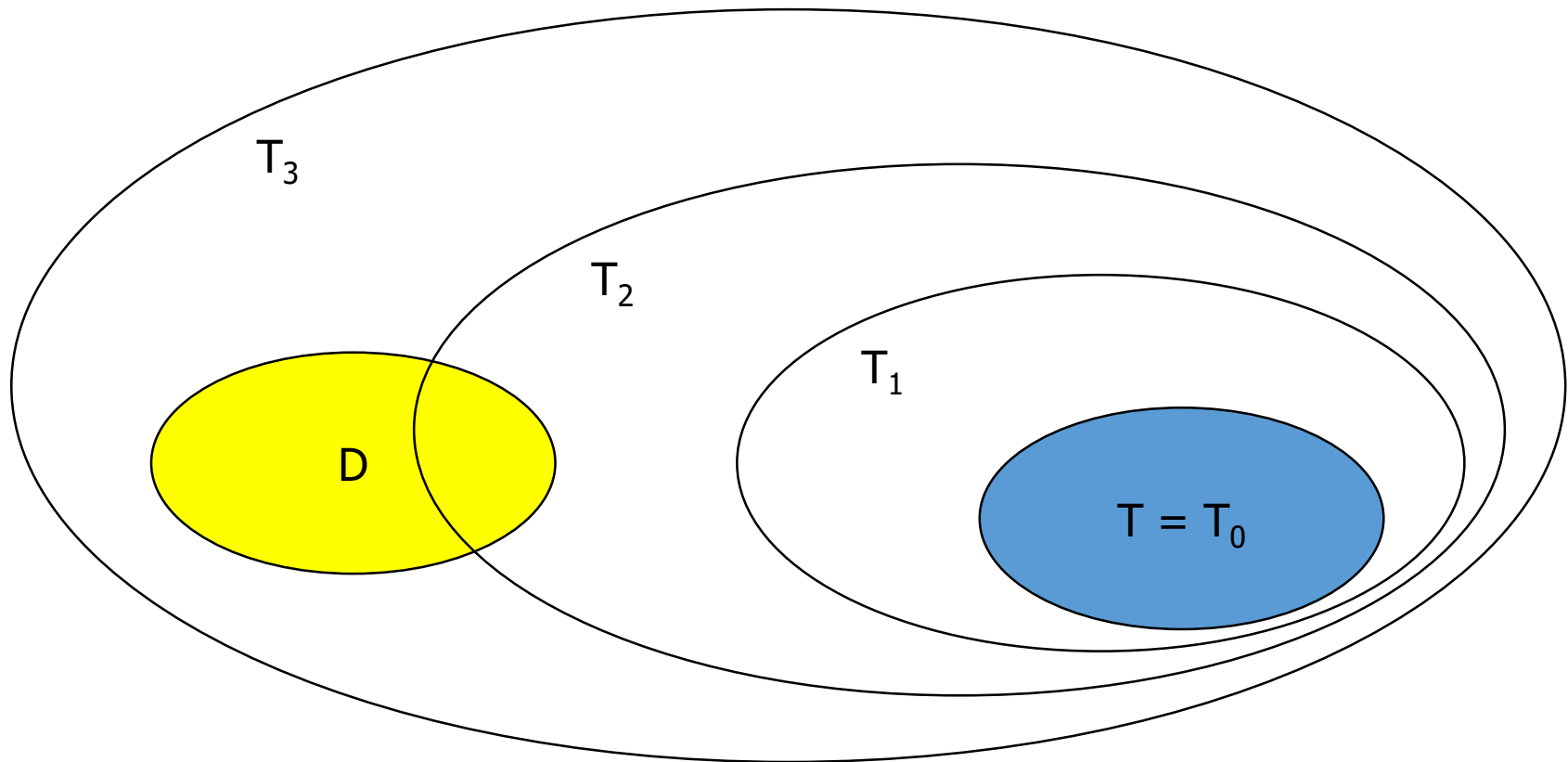
 L = L + right(r)

 T = T + L;

 if (D in T) break;

} while (L > {});

Mô tả quá trình suy diễn



Áp dụng thực tế



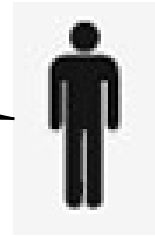
I. Mã hóa các biến mệnh đề

II. Mã hóa các luật

III. Chuẩn hóa các luật

IV. Xây dựng động cơ suy diễn

V. Xây dựng cơ chế diễn giải



- Đưa về dạng chuẩn
- Tách luật thành các luật nhỏ
- Ví dụ:
 - Nếu chia hết cho 2 và tận cùng bằng 0 hoặc 5 thì chia hết cho 10
 - A và $(B$ hoặc $C)$
 - $(A$ và $B)$ hoặc $(A$ và $C)$
 - Luật 1: Chia hết cho 2 và tận cùng bằng 0 thì...
 - Luật 2: Chia hết cho 2 và tận cùng bằng 5 thì...

Các vấn đề với suy diễn



- Cơ chế quen thuộc với con người nói chung
- Vấn đề:
 - Không có định hướng trong quá trình suy diễn
 - Có thể bị bùng nổ tổ hợp
 - Xuất hiện các sự kiện thừa (không cần thiết) trong quá trình diễn giải
- Định hướng:
 - Suy dẫn theo chiều sâu
 - Suy dẫn theo chiều rộng



Phần 5

Chứng minh bằng bác bỏ

Chứng minh bằng bác bỏ



- Phương pháp tương tự như việc chứng minh phản chứng mà con người thường sử dụng
- Nếu muốn chứng minh kết luận R , ta bổ sung $\neg R$ vào tập giả thiết
- Tiến hành suy diễn
 - Nếu đạt được \emptyset thì dừng và kết luận thỏa được
 - Nếu không đạt được \emptyset thì không chứng minh được

- Cho các mệnh đề đúng:

1. $\neg A \vee \neg B \vee C$

2. $\neg D \vee \neg E \vee C$

3. $\neg F \vee D$

4. A

5. F

6. E

- Chứng minh C

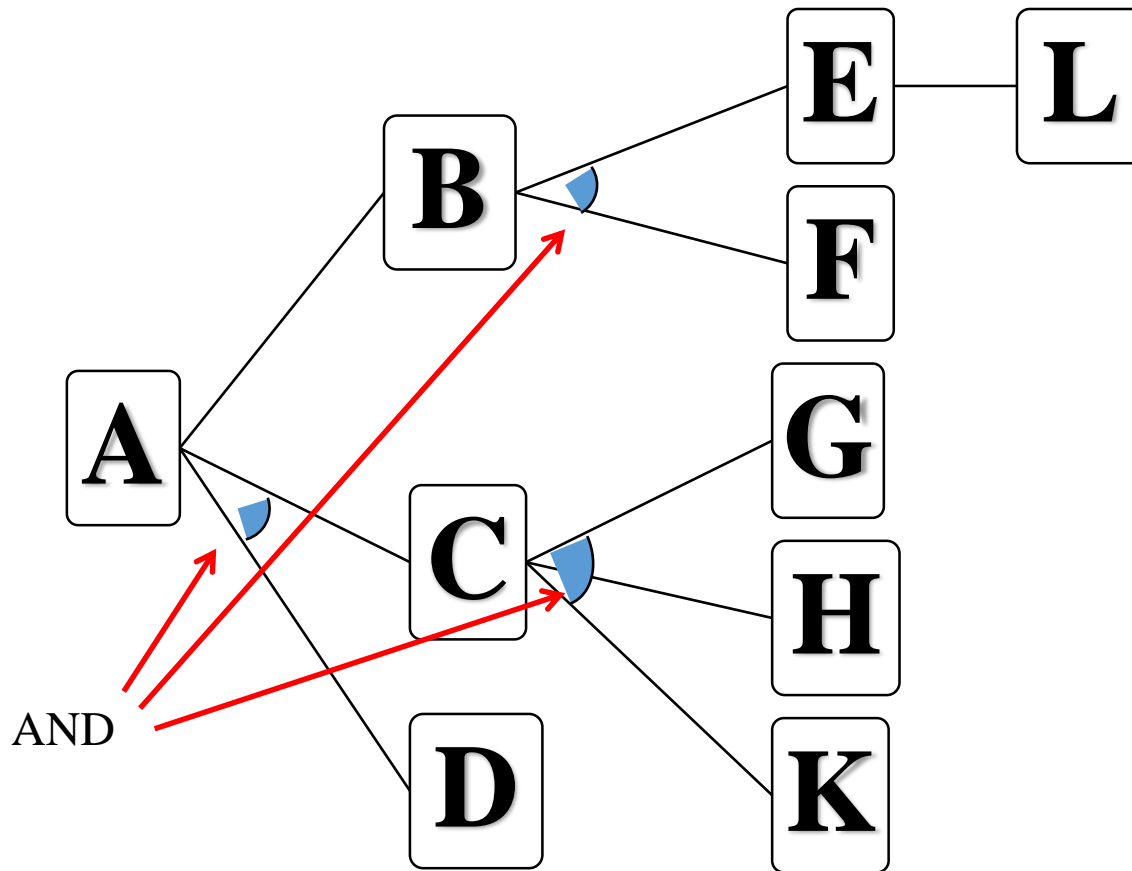
- Bổ sung thêm mệnh đề $\neg C$ (luật số 7)
- Kết hợp luật 2 và 7 ta được luật số 8
 - $(\neg D \vee \neg E \vee C) \vee (\neg C) \Rightarrow \neg D \vee \neg E$
- Kết hợp luật 6 và 8 ta được luật số 9
 - $(E) \vee (\neg D \vee \neg E) \Rightarrow \neg D$
- Kết hợp luật 3 và 9 ta được luật số 10
 - $(\neg F \vee D) \vee (\neg D) \Rightarrow \neg F$
- Kết hợp luật 5 và 10 ta được luật rỗng
 - $(F) \vee (\neg F) \Rightarrow \emptyset$



Phần 6

Suy diễn ì

Cây VÀ-HOẶC (AND-OR)



- Suy diễn tiến thực hiện việc suy diễn từ những giả thiết ban đầu và đi đến kết luận, suy diễn lùi làm theo hướng ngược lại:
 - Để chứng minh kết luận A_n , cần phải có ít nhất 1 tập các giả thiết A_{n-1} để suy ra được kết luận
 - Để chứng minh tập giả thiết A_{n-1} cần có chứng minh tập giả thiết A_{n-2}
 - ...
 - Kết thúc nếu tập giả thiết A_x được chứng minh

Thuật toán



T: Tập sự kiện ban đầu

R: Tập luật

D: Tập kết luận

```
function Back(D) {  
    if (D in T) return OK;  
    for (r in R) do  
        if (right(r) in D)  
            if (left(r) not in D)  
                if (Back(D - {right(r)} + {left(r)}))  
                    return OK;  
    return NotOK;  
}
```

Đặc điểm suy diễn lùi



- Thích hợp với những bài toán chuẩn đoán, tìm lỗi vì thường tập trung vào hiện tượng đích đã cho
- Đi theo chiều sâu, thường tiếp theo dòng suy diễn đã có thay vì rẽ qua nhánh khác