ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

BÁO CÁO ĐỒ ÁN 02

LOGIC

Môn học: Cơ sở trí tuệ nhân tạo

Giáo viên hướng dẫn: Lê Hoài Bắc
Nguyễn Duy Khánh
Nguyễn Ngọc Băng Tâm

Sinh viên thực hiện: Bùi Quốc Trung - 20120023

Thành phố Hồ Chí Minh, tháng 12 năm 2022

Mục Lục

1. Đánh giá mức độ hoàn thành

STT	Đặc tả tiêu chí	Mức độ hoàn thành
1	Đọc dữ liệu đầu vào và lưu trong cấu trúc dữ liệu phù hợp	100%
2	Cài đặt giải thuật hợp giải trên logic mệnh đề	100%
3	Các bước suy diễn phát sinh đủ mệnh đề và kết luận đúng	100%
4	Tuân thủ mô tả định dạng của đề bài	100%
5	Báo cáo test case và đánh giá	100%

2. Mô tả xử lý dữ liệu đầu vào

Đầu tiên, với mỗi mệnh đề ta sẽ lưu nó dưới dạng là danh sách các literal (list<string>). Vì thế mỗi đênh đề đọc từ file ta cần xóa đi các khoảng trắng và tác chúng ra thông qua OR (vì các literal cách nhau bởi OR). Đầu đầu tiên ta sẽ đọc vào α. Sau đó ta đoc vào số lượng KB (nKB). Rồi ta sẽ đọc các mệnh đề vào ta cũng sẽ xử lý và cho vào danh sách các mệnh đề của chúng ta.

Sau cùng, để chứng minh ta sẽ đi chứng minh phủ định của nó là sai. Với trường hợp α nhiều literal thì khi phủ định lại phép OR thành phép AND nên ta sẽ tách các phủ định của literal trong mệnh đề α ra thành các mệnh đề để đưa vào danh sách mệnh đề của chúng ta.

3. Mô tả thuật toán

Trước tiên, để lưu mệnh đề ta sẽ xây dựng một class Clause để quản lý mệnh đề. Trong class này ta sẽ quản lý mệnh đề như sau:

- Hàm normalize: hổ trợ việc loại bỏ bớt các literal trùng nhau chỉ để lại 1 (ví dụ: A OR B OR A thì là A OR B) và Các literal trong cùng mệnh đề (đối với cả dữ liệu đầu vào và đầu ra) được xếp theo thứ tự chữ cái (Ví dụ: B OR A thì thành A OR B)
- Hàm check: để kiểm tra xem mệnh đề có phải chân lý hay không (Vi những mệnh đề như thế này vô ích cho việc suy dẫn và do đó có thể bỏ đi).
- Hàm P_Resolve(clause1, clause2): hàm hợp giải 2 mệnh đề. Nếu 2 mện đề cần hợp giải mà có xuất hiện 1 cặp literal đối kháng thì ta lấy 2 cái đó ra và hợp giải những cái còn lại của 2 mệnh đề (ta gộp 2 mệnh đề lại với nhau). Nếu không có cặp đối kháng nào thì ta không hợp giải.

Và thuật toán chính để hợp giải 2 mệnh đề ta sẽ sử dụng **PL_Resolution** (Thuật toán được tham khảo từ sách Sách Artificial Intelligence: A Modern Approach, Third Edition, Chương 7, Hình 7.12, hàm PL-RESOLUTION):

function PL-RESOLUTION(KB, α) **returns** *true* or *false*
  inputs: KB, the knowledge base, a sentence in propositional logic
    α, the query, a sentence in propositional logic

  clauses $\leftarrow$ the set of clauses in the CNF representation of $KB \wedge \neg\alpha$
  new $\leftarrow \{\}$
  loop do
    for each pair of clauses C_i, C_j **in** *clauses* **do**
      resolvents $\leftarrow$ PL-RESOLVE(C_i, C_j)
      if *resolvents* contains the empty clause **then return** *true*
      new $\leftarrow$ *new* $\cup$ *resolvents*
    if *new* $\subseteq$ *clauses* **then return** *false*
    clauses $\leftarrow$ *clauses* $\cup$ *new*

Figure 7.12 A simple resolution algorithm for propositional logic. The function PL-RESOLVE returns the set of all possible clauses obtained by resolving its two inputs.

Trong thuật giải này ta duyệt hết các mệt đề và hợp giải từng cặp với nhau. Nếu hợp giải cho ra được mệnh đề mới (khác rỗng và chưa xuất hiện trong danh sách mệnh đề trước hoặc chưa được tạo ra trong quá trình hợp giải thì ta thêm nó vào danh sách các mệnh đề mới. Nếu danh sách mệnh đề mới rỗng thì không thể hợp giải nữa nên ta in ra NO và kết thúc quá trình hợp giải. Còn nếu trong mệnh đề mới xuất hiện (đã xuất hiện cặp mệnh đề đối kháng) thì ta sẽ dừng hợp giải và in ra YES. Nếu không trong 2 mệnh đề trên ta sẽ thêm danh sách mới vào dưới danh sách mệnh đề vào tiếp tục việc hợp giải cho đến khi có kết quả.

4. Kiểm thử và đánh giá

4.1. Test-case01

input1.txt	output1.txt	Ghi chú
A	7	
5	A OR B	(B OR C OR A) hop giai voi (A OR -C)
B OR C OR A	B OR C	(B OR C OR A) hop giai voi (B OR -A)
A OR -C	A OR C	(B OR C OR A) hop giai voi (-B OR C)
B OR -A	B OR -C	(A OR -C) hop giai voi (B OR -A)
-B OR C	A OR -B	(A OR -C) hop giai voi (-B OR C)
-B	-C	(A OR -C) hop giai voi (-A)
	-A OR C	(B OR -A) hop giai voi (-B OR C)
	3	
	A	(A OR -C) hop giai voi (A OR C)
	B	(B OR -A) hop giai voi (A OR B)
	C	(-B OR C) hop giai voi (B OR C)
	1	
	{}	(-B) hop giai voi (B)
	YES	KB entails α vì tồn tại mệnh đề rỗng trong KB

Với test-case này cho ta kết quả đúng tuy nhiên nó cần hợp giải nhiều mệnh đề. Trong thực tế thì ta chỉ cần hợp giải cần thiết để ra kết quả: (B OR C OR A) hợp giải với (-B) ra (A OR C). (A OR C) hợp giải (A OR -C) ra (A) (đối kháng với (-A)) => YES.

4.2. Test-case02

input2.txt	output2.txt	Ghi chú
C	2	
5	A OR F	(A OR B) hop giai voi (-B OR F)
A OR B	{}	(E) hop giai voi (-E)
E	YES	KB entails α vì tồn tại mệnh đề rỗng trong KB
-B OR F		
A OR D		
-E		

Ở test case này ta cần chứng minh (C) try nhiên trong mệnh đề có cặp đối kháng (E) và (-E) nên đưa ra YES. Vì đây là các căp gây nhiễu nên khi ta thu thập thì lun cho ra YES.

4.3. Test-case03

input3.txt	output3.txt	Ghi chú
H	2	
4	B OR C OR D OR E	(B OR D OR -A) hop giai voi (E OR A OR C)
B OR D OR -A	-A OR B	(B OR D OR -A) hop giai voi (-D)
E OR A OR C	1	
F OR C	B OR C OR E	(E OR A OR C) hop giai voi (-A OR B)
-D	0	
	NO	KB KHONG entail α vi khong phat sinh duoc menh de moi va khong tim thay menh de rong

Với test case này ta cần chứng minh (H) tuy nhiên trong mệnh đề không có xuất hiên 1 literal (H) nào cả và lại cũng k có căp gây nhiểu vì thế rõ ràng là chắc chắn không suy diễn dẫn đến (H) được nhưng thuật toán phải hợp giải nhiều lần rồi mới kết luận.

4.4. Test-case04

input4.txt	output4.txt	Ghi chú
A OR B	3	
3	A OR D	(A OR C OR D) hop giai voi (-C)
A OR C OR D	A OR C	(A OR C OR D) hop giai voi (-D)
-C	C OR D	(A OR C OR D) hop giai voi (-A)
-D	3	
	A	(-C) hop giai voi (A OR C)
	D	(-C) hop giai voi (C OR D)
	C	(-D) hop giai voi (C OR D)
	1	
		(-C) hop giai voi (C)
	YES	KB entails α vì tồn tại mệnh đề rỗng trong KB

Với việc α có nhiều literal thì ta thêm các phủ định của các literal đó vào trong mệnh đề. Ở trên ta có (A OR B) phủ định thành (-A) AND (-B) thì ta lần lượt (-A) và (-B) vào trong mệnh đề để hợp giải.

4.5. Test-case05

input5.txt	output5.txt	Ghi chú
A OR B	3	
4	-A OR -B OR E	(-A OR C) hop giai voi (-C OR -B OR E)
-A OR C	-B OR -C OR D	(-C OR -B OR E) hop giai voi (-E OR D)
-C OR -B OR E	D OR F	(E OR F) hop giai voi (-E OR D)
E OR F	1	
-E OR D	-A OR -B OR D	(-A OR C) hop giai voi (-B OR -C OR D)
	0	(-C) hop giai voi (C OR D)
	NO	KB KHONG entail α vi khong phat sinh duoc menh de moi va khong tim thay menh de rong

5. Đánh giá chung thuật toán

5.1. Ưu điểm

Thuật toán của chúng ta duyệt hết tất cả các mệnh đề để hợp giải nên lun cho ra kết quả đầy đủ. Thuật toán dùng các luật dễ hiểu và dễ dàng cài đặt.

5.2. Nhược điểm

vì thuật toán vét cạn nên sẽ dẫn đến dư thừa việc hợp giải không liên quan và việc hợp giải nếu có nhiều mệnh đề và tạo ra nhiều mệnh đề mới thì sẽ làm cho chương trình chạy chậm. Chưa xử lý đươc việc các cặp nhiễu trong input. Và thuât toán chỉ thực hiện được trên các mệnh đề chuẩn CNF.

5.3. Đề xuất phương pháp cải tiến

Ưu tiên chọn các mệnh đề hợpj giải liên quan đến α. Ngoài ra, Sử dụng các thuật toán suy diễn với mệnh đề Horn: suy diễn tiến và suy diễn lùi.

6. Mô tả file run.sh

File run.sh sẽ chứa thông tin của các thành viên trong nhóm và câu lệnh gọi tới file main.py nằm trong thư mục source để chạy toàn bộ chương trình.
Khi được gọi chạy, main,py sẽ thực hiện đọc các tập tin input<x>.txt bên trong thư mục INPUT và đưa ra các tập tin kết quả output<x>.txt tương ứng trong thư mục OUTPUT

Tài liệu tham khảo

[1] Slies bài giảng của thầy Lê Hoài Bắc và hướng dẫn thực hiện đồ án của thầy Nguyễn Duy Khánh

[2] Sách Artificial Intelligence: A Modern Approach, Third Edition): https://zoo.cs.yale.edu/classes/cs470/materials/aima2010.pdf