

CASE-BASED REASONING VÀ KHẢ NĂNG TIẾP CẬN BẰNG LOGIC VỊ TỪ

Nguyễn Việt Hà, Phạm Ngọc Hùng, Hồ Sĩ Đàm

Trường Đại học Công nghệ, Đại học quốc gia Hà Nội

Trần Vũ Việt Anh

Đại học quốc gia Singapore

1. Đặt vấn đề

Phương pháp tìm lời giải dựa trên các ca suy luận mẫu đã có (Case-based reasoning - CBR) được khởi xướng từ thập kỷ 80 và ngày càng thu hút mối quan tâm của nhiều nhà nghiên cứu [1, 2, 3, 4]. CBR hoạt động theo cơ chế so sánh bài toán cần giải với các bài toán mẫu đã có lời giải và tìm cách đưa ra một lời giải dựa trên lời giải mẫu. Đã có nhiều hệ thống CBR được phát triển như các hệ chuyên gia chẩn đoán y học, lập kế hoạch hay lập luận về luật.

Tuy vậy, cơ chế suy luận của CBR vẫn chưa được hình thức hóa chặt chẽ. Nhiều nghiên cứu và ứng dụng chỉ chú trọng vào việc tìm kiếm bài toán cũ tương tự và đưa ra lời giải mới bằng cách thay thế từ khóa. Đặc thù của CBR là sử dụng lại lời giải cũ, do đó một số lớn trong các bước suy diễn để đưa ra lời giải mới bị loại bỏ. Điều này dẫn đến vấn đề khó đảm bảo được tính đúng đắn về mặt logic của lời giải mới. Thêm nữa, cài đặt CBR thường khó tổng quát hóa do phải phối hợp giữa các tri thức kinh nghiệm là các lời giải đã biết và các tri thức tổng hợp về miền bài toán. Chưa tồn tại ứng dụng CBR nào tận dụng được năng lực suy diễn và tính tường minh của logic vị từ. Trong bài báo này, chúng tôi phân tích khía cạnh suy diễn của CBR và đề xuất một phương thức cài đặt CBR bằng logic vị từ cấp một. Trong phương thức này, lời giải cũ đóng vai trò là các kinh nghiệm suy luận giúp tăng tốc độ tìm lời giải mới và tính đúng đắn của lời giải mới được đảm bảo bằng suy diễn.

2. Case-based reasoning

2.1. Khái niệm

CBR là phương pháp tìm lời giải dựa trên việc sử dụng lại các lời giải (ca suy luận) trong quá khứ. CBR là phương pháp tương đối mới và chưa được hình thức hóa một cách chặt chẽ. Về cơ bản, CBR tìm lời giải như sau (hình 1) [2]:

- ◆ Lưu các *ca suy luận* đã gặp trong quá khứ vào CSDL ca suy luận.
- ◆ Khi tìm lời giải cho một bài toán mới, trước tiên tìm kiếm các bài toán tương tự trong CSDL ca suy luận.

- ◆ Đưa ra lời giải mới bằng cách hiệu chỉnh lời giải của ca suy luận tìm được.
- ◆ Lưu ca suy luận mới tạo ra vào CSDL ca suy luận.

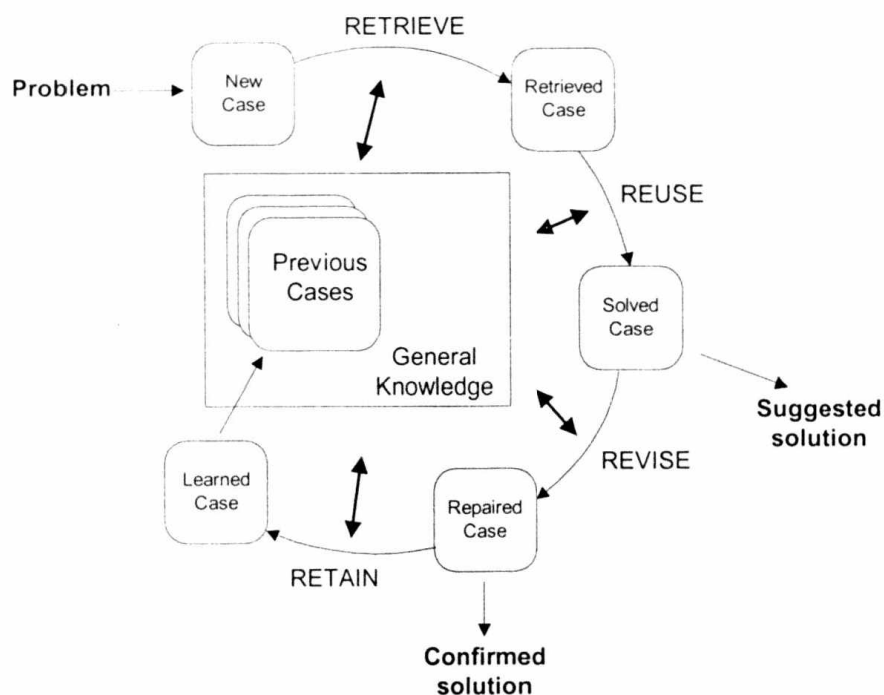
Ca suy luận thông thường được biểu diễn dưới dạng một bộ gồm *đặc tả bài toán* và *lời giải*. Quá trình tìm lời giải cho bài toán mới được bắt đầu bằng việc tìm kiếm ca suy luận tương tự. Đặc tả được tạo ra bằng cách chỉ số hóa các đặc trưng và việc tìm kiếm thông thường được thực hiện bằng cách so sánh các chỉ số này.

Lời giải của ca suy luận thường được biểu diễn bằng các khung tri thức (frame). Các khung tri thức phù hợp với việc mô tả nhiều khái niệm tri thức khác nhau, nhất là các tri thức dưới dạng kinh nghiệm.

Quá trình tạo ra lời giải mới từ lời giải đã có thường được tiến hành theo hai bước: tái sử dụng và hiệu chỉnh. Trước tiên, lời giải cũ được tái sử dụng thông qua các phương pháp đơn giản như thay thế từ khóa. Sau đó, cần hiệu chỉnh lời giải này sao cho thực sự phù hợp với bài toán cần giải. Việc hiệu chỉnh phụ thuộc rất nhiều vào miền ứng dụng và hiện tại thì với mỗi hệ CBR cụ thể thì người ta lại phát triển một cơ chế riêng. Có thể nói hiệu chỉnh chính là nhân tố quyết định sự thành công của CBR và chưa có một cơ chế hiệu chỉnh có thể áp dụng một cách tổng quát.

Lời giải mới được tạo ra có thể được cập nhật vào CSDL ca suy luận để sử dụng lại trong tương lai. Việc cập nhật không nhất thiết là thêm ca suy luận mới mà có thể là hiệu chỉnh ca suy luận cũ để cho nó mang tính đại diện hơn. CSDL ca suy luận cũng không nhất thiết chỉ lưu các ca suy luận thành công mà cũng có thể chứa cả các ca suy luận không thành công.

CBR đang được nghiên cứu và ứng dụng một cách rộng rãi, đặc biệt là với những hệ chuyên gia cần tới các tri thức dạng kinh nghiệm. Một số các nghiên cứu điển hình là hệ chuyên gia tạo món ăn Trung Hoa CHEF, chẩn đoán bệnh PROTOS, giải quyết xung đột quốc tế MEDIATOR, tư vấn luật HYPO. CBR cũng bắt đầu được ứng dụng hiệu quả trong công nghiệp và kinh doanh như hệ trợ giúp khách hàng SMART của Compaq, hệ trợ giúp thiết kế CLAVIER của Lockheed [3]. Một hướng nghiên cứu mới là phát triển các hệ *CBR hội thoại* cho phép người sử dụng tham gia vào quá trình hiệu chỉnh lời giải [5].



Hình 1. Case-based reasoning.

2.2. Tìm hiểu CBR thông qua hệ chuyên gia CHEF

CHEF [4] là một hệ chuyên gia trợ giúp việc tạo cách chế biến món ăn Trung Hoa. CHEF chứa 10 ca suy luận về cách chế biến món ăn và một số tri thức tổng hợp về nấu ăn. Hình 2 mô tả cách chế biến món “đậu xào thịt bò”.

A half pound of beef
 Two tablespoons of soy sauce
 One teaspoon of rice wine
 A half tablespoon of corn starch
 One teaspoon of sugar
 A half pound of green bean
 One teaspoon of salt
 One chunk of garlic

Chop the garlic into pieces the size of matchheads
 Shred the beef
 Marinate the beef in the garlic, sugar, corn starch, rice wine and soy sauce
 Stir fry the spices, rice wine and beef for one minute
 Add the green bean to the spices, rice wine and beef
 Stir fry the spices, rice wine, green bean and beef for three minutes
 Add the salt to the spices, rice wine, green bean and beef

Hình 2. Cách chế biến “đậu xào thịt bò”.

Giả sử, chúng ta cần cách chế biến “xúp lơ xào thịt bò”. Khi đó từ các thông tin “thịt bò”, “xào” và tri thức “xúp lơ” và “đậu” cùng là “rau”, CHEF tìm ra cách chế biến gần giống nhất là “đậu xào thịt bò”. Nếu chỉ đơn thuần thay “xúp lơ” vào vị trí của “đậu” thì chưa thể gọi là tạo ra một cách chế biến mới. CHEF vận dụng các tri thức về nấu ăn khác để hiệu chỉnh cách chế biến này. Trước tiên, từ sự khác nhau về kích thước giữa “đậu” và “xúp lơ”,

CHEF sẽ thêm công đoạn “thái xúp lơ” trước khi xào. Thêm nữa, từ các tri thức “xúp lơ chứa nhiều nước”, “thịt bò không nên xào với nhiều nước”, và “nếu xào chung không ngon thì có thể xào riêng rẽ”, CHEF chỉnh sửa cách chế biến thành “xào xúp lơ và thịt bò riêng rẽ”. Hình 3 là cách chế biến “xúp lơ xào thịt bò” do CHEF tạo ra.

Rõ ràng, CHEF không chỉ đơn thuần là hệ thống tìm kiếm/thay thế mà còn có năng lực suy diễn dựa trên các luật.

2.3. Biểu diễn tri thức và suy diễn

Trong các hệ thống CBR, tri thức được lưu song song bằng hai dạng: ca suy luận và luật suy diễn. Ca suy luận là các tri thức dạng kinh nghiệm và luật là các tri thức đã được hình thức hóa cao cho suy diễn. Ví dụ, CHEF chứa ca suy luận là các cách chế biến một số món ăn điển hình và các luật như “nếu xào chung không ngon thì xào riêng rẽ”. CHEF vận dụng luật này để đưa ra cách giải quyết là xào xúp lơ và thịt bò riêng rẽ. Bằng việc sử dụng các luật, CBR hiệu chỉnh lời giải của ca suy luận trong quá khứ và đưa ra lời giải mới phù hợp với bài toán. Nghĩa là, CBR không chỉ đơn thuần là hệ tìm kiếm mà còn là một hệ suy diễn và thông qua việc suy diễn CBR tìm cách đảm bảo tính thỏa đáng của lời giải. Tuy nhiên việc phối hợp giữa ca suy luận và luật suy diễn thường rất phức tạp và khó xây dựng được một cơ chế tổng quát.

Giả sử, nếu CBR có chứa các luật đủ để có thể tự suy diễn ra lời giải cho các ca suy luận mẫu thì hiển nhiên nó cũng sẽ suy diễn ra lời giải cho bài toán mới mà không cần tham khảo tới ca suy luận mẫu. Tức là trong trường hợp đặc biệt này CBR đảm bảo được tính đúng đắn về mặt logic của lời giải. Vậy trong trường hợp này ý nghĩa của ca suy luận nằm ở chỗ nào? Bản chất của ca suy luận mẫu trong hệ thống là làm tăng hiệu suất tìm lời giải. Trong thực tế, con người vẫn thường tìm lời giải bằng dựa vào kinh nghiệm giải các bài toán tương tự trong quá khứ. Việc xuất phát từ lời giải của tình huống tương tự sẽ rút ngắn thời gian để đưa ra lời giải cần thiết. Giống như vậy, các ca suy luận mẫu chính là các kinh nghiệm giúp hệ thống thu hẹp không gian tìm kiếm lời giải.

A half pound of beef
Two tablespoons of soy sauce
One teaspoon of rice wine
A half tablespoon of corn starch
One teaspoon of sugar
A half pound of broccoli
One teaspoon of salt
One chunk of garlic

Chop the garlic into pieces the size of matchheads
Shred the beef
Marinate the beef in the garlic, sugar, corn starch, rice wine and soy sauce
Chop the broccoli into pieces the size of chunks
Stir fry the broccoli for three minutes
Stir fry the spices, rice wine and beef for one minute
Add the broccoli to the spices, rice wine and beef
Stir fry the spices, rice wine, broccoli and beef for a half minutes
Add the salt to the spices, rice wine, green bean and beef

Hình 3. Cách chế biến “xúp lơ xào thịt bò”.

Năng lực của các hệ suy diễn được quyết định bởi lượng kiến thức và phương pháp suy diễn. Thông thường, với các phương pháp suy diễn càng hiệu quả thì năng lực biểu diễn càng thấp. Do đó, không phải lúc nào chúng ta cũng dễ dàng hình thức hóa được các tri thức, nhất là tri thức kinh nghiệm dưới dạng các luật suy diễn. Ở điểm này, các ca suy luận ngoài việc làm tăng hiệu suất tìm lời giải còn là một phương pháp hỗ trợ đắc lực cho chúng ta trong việc biểu diễn tri thức.

Mặt khác, bằng việc sử dụng các ca suy diễn, việc bổ sung (học) tri thức được tiến hành một cách tự nhiên. Nếu người sử dụng chấp nhận lời giải thì có thể cập nhật nó vào CSDL ca suy luận. Như vậy, với các bài toán tương tự chúng ta hoàn toàn có thể hy vọng hệ thống sẽ đưa ra lời giải với tốc độ nhanh hơn.

Tóm lại, CBR là phương pháp tìm lời giải kết hợp giữa suy diễn và vận dụng kinh nghiệm chứ không loại bỏ suy diễn. Memory-based reasoning (MBR) là phương pháp phát triển dựa trên CBR theo hướng chỉ vận dụng kinh nghiệm và loại bỏ hẳn suy diễn [6]. MBR thông thường được cài đặt trên các máy tính song song và tận dụng năng lực tính toán cao này để tìm ca suy luận tương tự từ các CSDL lớn [7]. Trong các phương pháp này, vấn đề đánh giá độ tương tự để tìm ca suy luận mẫu thỏa đáng nhất là đặc biệt quan trọng. Việc hình thức hóa độ tương tự cũng khó khăn như việc hình thức hóa các tri thức dạng kinh nghiệm. Suy luận tương tự (analogical reasoning - analogy) là một cách tiếp cận sử dụng sự tương tự vào việc tìm lời giải [8]. Khác với CBR tính độ tương tự trong cùng một miền bài toán, analogy tính độ tương tự giữa cấu trúc của các miền bài toán và tìm cách ánh xạ tri thức từ miền này sang miền kia.

3. Cài đặt CBR bằng logic vị từ

Trong mục này chúng tôi đề xuất một phương thức cài đặt CBR bằng logic vị từ cấp một. Lời giải cho một bài toán mới được suy diễn bằng cách ưu tiên áp dụng các luật đã được sử dụng trong ca suy luận mẫu. Phương thức này cho phép đảm bảo tính đúng đắn về mặt logic của lời giải do nó được suy diễn từ các luật chứa trong hệ thống. Các ca suy luận mẫu lưu trữ trong hệ thống được xem như là các kinh nghiệm suy luận và sẽ làm tăng tính hiệu quả trong việc tìm lời giải. Chúng tôi giới hạn bài toán là các bài toán dạng Yes/No và sử dụng phương pháp phản bác (SLD) để chứng minh.

3.1. Biểu diễn ca suy luận

1) Đặc tả ca suy luận và tính độ tương tự

Chúng tôi định nghĩa đặc tả bài toán là tập các vị từ xuất hiện trong câu hỏi và trong các điều kiện (sự thực) của bài toán. Ví dụ, với bài toán

Bài toán 1

Sự thực:

skilful(John)
mechanic(John)
truck(T)

Câu hỏi:

repair(John, T)

thì đặc tả tương ứng sẽ là tập hợp {skilful, mechanic, truck, repair}.

Có nhiều phương pháp để so sánh độ giống nhau của đặc tả, ở đây chúng tôi tính độ tương tự $sim(c_1, c_2)$ giữa hai đặc tả c_1 và c_2 như sau:

$$sim(c_1, c_2) = \frac{2s}{n_1 + n_2} \quad (1)$$

trong đó s là số các vị từ giống nhau giữa hai đặc tả; n_1, n_2 lần lượt là số vị từ trong từng đặc tả. Như vậy độ tương tự sẽ nhận giá trị từ 0 đến 1.

2) Lời giải của ca suy luận

Lời giải của một ca suy luận được định nghĩa là tập các luật giải được dùng trong cây chứng minh của ca đó. Giả sử hệ thống sử dụng các luật sau để chứng minh bài toán 1:

skilful(x) maintain(x, y) -> repair(x, y)
mechanic(x) vehicle(y) -> maintain(x, y)
truck(x) -> vehicle(x)

Khi đó lời giải của ca suy luận tương ứng sẽ chính là tập chứa các luật này. Hình 4 minh họa một ca suy luận hoàn chỉnh.

Description	skilful, mechanic, truck, repair
Solution	skilful(x) maintain(x, y) -> repair(x, y), mechanic(x) vehicle(y) -> maintain(x, y), truck (x) -> vehicle(x)

Hình 4. Ví dụ về ca suy luận

3.2. Quá trình tìm lời giải

Việc tìm lời giải cho một bài toán mới được tiến hành như sau:

- tạo đặc tả cho bài toán
- sử dụng công thức (1) tìm ca suy luận mẫu phù hợp nhất với bài toán
- chứng minh bài toán bằng cách ưu tiên áp dụng các luật của ca sử dụng mẫu

Điểm khác biệt so với các hệ CBR khác là ở phương pháp tái sử dụng/hiệu chỉnh lời giải. Lời giải mới được đưa ra bằng phương pháp SLD chứ không phải được tạo ra trực tiếp

từ lời giải cũ. Tuy nhiên, thay vì áp dụng các luật một cách tuần tự thì chúng ta ưu tiên áp dụng các luật trong lời giải của ca suy luận tương tự.

Xét bài toán sau:

Bài toán 2

Sự thực:

skilful(Tom)

mechanic(Tom)

motorbike(M)

Câu hỏi:

repair(Tom, M)

Đặc tả của bài toán này sẽ là {skilful, mechanic, motorbike, repair}. Độ tương tự giữa đặc tả này và đặc tả của ca suy luận trong hình 4 theo công thức (1) là 0,75. Giả sử đây là ca suy luận gần giống nhất với bài toán 2. Chúng ta cũng giả sử rằng hệ thống chứa một tập các luật suy diễn mô tả trong hình 5. Khi đó hệ thống sẽ ưu tiên áp dụng các luật của ca suy luận nói trên và tạo ra được cây chứng minh và người sử dụng sẽ nhận được câu trả lời "Yes". Các luật được áp dụng trong cây chứng minh sẽ là r200, r201, r203.

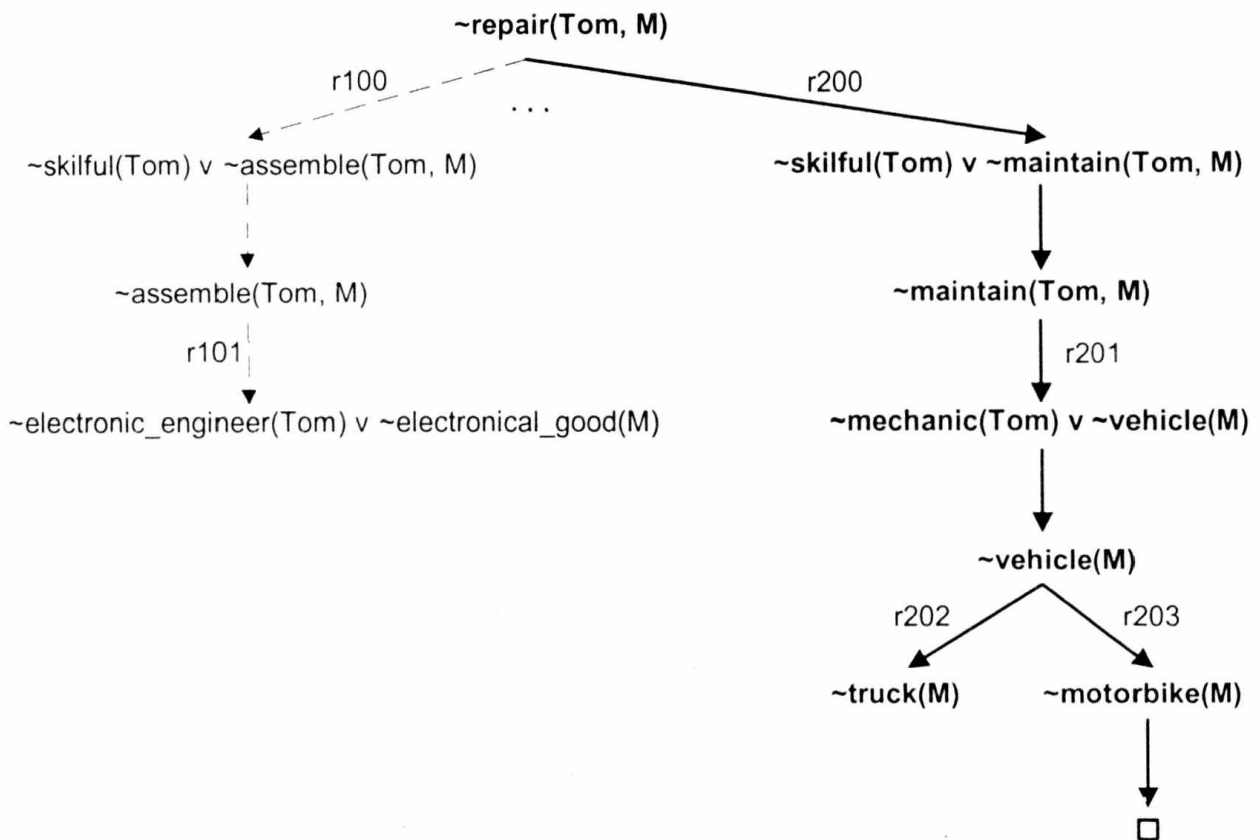
3.3. Khảo sát

Chúng ta có thể thấy lời giải (tập các luật được sử dụng) của ca suy luận và lời giải cho bài toán là rất giống nhau. Tuy chúng ta suy diễn từ các luật của hệ thống nhưng đồng thời, trên khía cạnh một hệ CBR, cũng có thể nói rằng chúng ta đã thành công trong việc hiệu chỉnh lời giải mẫu thành lời giải cho bài toán mới.

Lời giải thu được hiển nhiên là lời giải đúng đắn do nó được suy diễn từ các sự thực và luật trong hệ thống. Ý nghĩa của ca suy luận mẫu là ở chỗ nó góp phần tăng tốc độ tìm kiếm lời giải. Rõ ràng, việc ưu tiên áp dụng các luật của ca mẫu sẽ hạn chế được sự bùng nổ của không gian tìm kiếm trong quá trình xây dựng cây chứng minh. Nếu không ưu tiên sử dụng các luật trong ca suy luận mẫu, việc tìm kiếm có thể được tiến hành với nhiều nhánh không liên quan gì đến lời giải cuối cùng. Hình 6 mô tả không gian tìm kiếm truyền thống, tức là không ưu tiên các luật của ca suy luận mẫu. Các luật như "skilful(x) assemble(x, y) -> repair(x, y)" và "electronic_engineer(x) electrical_good(y) -> assemble(x, y)" được sử dụng trước và tạo ra một nhánh tìm kiếm không cần thiết (nửa cây bên trái). Với các CSDL luật lớn, không gian tìm kiếm có thể bị bùng nổ dẫn đến không tìm được lời giải trong một thời gian cho trước. Với phương thức đề xuất, chúng ta ưu tiên áp dụng các luật của ca suy luận mẫu và khi đó chúng ta chỉ cần tìm kiếm trong một không gian hẹp. Ví dụ như với bài toán nêu trên là nửa bên phải, phần in đậm trong hình 6.

r100: skilful(x) assemble(x, y) \rightarrow repair(x, y)
 r101: electronic_engineer(x) electrical_good(y) \rightarrow assemble(x, y)
 r102: television(x) \rightarrow electrical_good(x)
 r103: radio(x) \rightarrow electrical_good(x)
 ...
 r200: skilful(x) maintain(x, y) \rightarrow repair(x, y)
 r201: mechanic(x) vehicle(y) \rightarrow maintain(x, y)
 r202: truck(x) \rightarrow vehicle(x)
 r203: motorbike(x) \rightarrow vehicle(x)

Hình 5. Các luật suy diễn của hệ thống.



Hình 6. Không gian tìm kiếm truyền thống.

Tóm lại, với phương thức đề xuất chúng ta tận dụng được ưu điểm của CBR là sử dụng lại kinh nghiệm và ưu điểm của logic vị từ là năng lực suy diễn và tính đúng đắn. Thêm nữa, phương thức này dễ dàng cài đặt và vận dụng cho lớp các bài toán thông dụng vẫn được cài bằng logic vị từ. Một điểm cần lưu ý ở đây là các ca suy luận mẫu nêu trên mới chỉ áp dụng cho trường hợp lời giải khẳng định, tức là tồn tại một cây chứng minh. Với trường hợp phủ định, chúng ta sẽ phải thử mọi nhánh tìm kiếm có thể và thời gian để đưa ra câu trả lời phủ định tương đương với thời gian suy luận trong các hệ thống suy luận truyền thống.

4. Kết luận

Trong bài báo này, chúng tôi đã khảo sát CBR trên góc độ là một hệ suy diễn và chỉ rõ bản chất của mô hình này. Tri thức trong CBR được biểu diễn bằng một mô hình lai giữa các kinh nghiệm suy luận và luật. Lời giải được đưa ra dựa trên sự vận dụng đồng thời hai loại tri thức này. Có thể nói CBR rất gần với mô hình suy luận của con người.

Chúng tôi đề xuất một phương thức cài đặt CBR bằng logic vị từ cấp một. Trong phương thức này, lời giải của ca suy luận mẫu được lưu bằng tập các luật giải trong cây suy luận. Lời giải của bài toán mới sẽ được tìm kiếm hiệu quả bằng việc suy luận từ tập các luật giải của ca suy luận mẫu. Điểm mới của phương thức này là đảm bảo được tính đúng đắn về mặt logic của lời giải. Phương thức đề xuất có ý nghĩa quan trọng trong việc đưa ra một cách tiếp cận hình thức cho CBR. Hướng nghiên cứu tiếp theo của chúng tôi là hình thức hóa chặt chẽ và khảo sát tính hiệu quả của phương thức này.

TÀI LIỆU THAM KHẢO

1. Kolodner J., *Case-based reasoning*, Morgan Kaufmann, 1993, 668 tr
2. Aamodt A. and Plaza E., Case-based reasoning: Foundational issues, methodological variations, and system approaches, *AI Communications* Vol. 7 No. 1 (1994), 39-59.
3. Watson I., *Applying Case-based reasoning: Techniques for Enterprise Systems*, Morgan Kaufmann, 1997, 290tr.
4. Hammond K.J., Case-based planning: a framework for planning from experience, *Cognitive Science* Vol. 14 (1990), 385-443.
5. Aha D.W. and Breslow L.A., Conversational case-based reasoning, *Applied Intelligence* Vol. 14 (2001) , 9-32.
6. Stanfill C. and Waltz D., The memory based reasoning paradigm, in *Case based reasoning workshop* (1988), 414-424.
7. Kitano H., Challenges for massive parallelism, *IJCAI-93*, 813-834.
8. Hall R.P., Computational approaches to analogical reasoning: A comparative analysis, *Artificial Intelligence* Vol. 39, No. 1 (1989), 39-120.

CASE-BASED REASONING AND AN IMPLEMENTING APPROACH BY PREDICATE LOGIC

Nguyen Viet Ha, Pham Ngoc Hung, Ho Si Dam

College of Technology, Vietnam National University, Hanoi

Tran Vu Viet Anh

The National University of Singapore

Case-based reasoning (CBR) is a recent approach to problem solving and learning. It derives a solution for a given problem by searching previous cases and adapting the solution of the most similar case. This method has received a lot of attention over the last few years and many systems have been implemented. However, the solution revising of CBR is based on problem domain, which makes it complicated. In addition, the soundness of obtained solutions cannot be guaranteed in many cases because of the shortness of inferring process. This paper discusses CBR on the view of reasoning and proposes a framework to implement CBR by predicate logic. In this framework, previous cases are used to reduce the inference cost and the obtained solution is guaranteed to be sound.