

TRÌNH ĐỘ CAO

KỸ THUẬT KIẾN THỨC

Giảng viên:

KHUẤT THANH TÙNG

Đại học Công nghệ Sydney

Đề cương môn học

Chủ đề 1: Tổng quan về Kỹ thuật tri thức Chủ đề 2: Tổng quan
về Hệ thống dựa trên tri thức Chủ đề 3: Thu thập tri thức **Chủ đề**
4: Biểu diễn và lý luận tri thức

Đánh giá giữa kỳ Chủ đề 5:

Ontology Chủ đề 6: Đồ

thị tri thức Chủ đề 7: Hệ thống

chuyên gia Chủ đề 8: Lý luận

không chắc chắn Chủ đề 9: Hệ thống dựa

trên tri thức lai Chủ đề 10: Lập kế hoạch AI tự động

Các dự án nhóm cho các chủ đề nâng cao

BIỂU DIỄN KIẾN THỨC VÀ LÝ LUẬN

Mục tiêu của chủ đề này

Đến cuối chủ đề này, bạn sẽ có thể:

giải thích cách thức kiến thức có thể được biểu diễn dưới dạng khai báo
chương trình

mô tả và phân tích quá trình suy luận

giải thích các nguyên tắc của chuỗi ngữ ý và chuỗi tiến

phân tích loại chuỗi được sử dụng bởi một chuyên gia cụ thể
hệ thống

giải thích cách mạng ngữ nghĩa biểu diễn dữ liệu

xác định ưu điểm và nhược điểm của ngữ nghĩa
mạng lưới

giải thích cách sử dụng khung để biểu diễn kiến thức

Chương trình nghị sự

Giới thiệu về Biểu diễn tri thức và Lý luận

Lập trình thủ tục so với lập trình khai

báo Phương pháp biểu diễn tri thức

Logic bậc nhất

Lý luận

Phát triển hệ thống dựa trên quy tắc

Mạng ngữ nghĩa

Khung

Giới thiệu về Kiến thức Biểu diễn và lý luận

Biểu diễn tri thức và lý luận là lĩnh vực nghiên cứu liên quan đến cách sử dụng hệ thống ký hiệu để biểu diễn một phạm vi kiến thức với các chức năng cho phép suy luận (lý luận chính thức) về các đối tượng trong phạm vi đó.

Chúng ta định nghĩa trước kiến thức là một niềm tin được biện minh làm tăng khả năng hành động hiệu quả của một thực thể.

Đề xuất

Biểu tượng chính thức

Lý luận

Lập trình thủ tục so với lập trình khai báo

Lập trình thủ tục

Một chương trình được viết bằng ngôn ngữ thủ tục (ví dụ: C++ hoặc Java) bao gồm một tập hợp các thủ tục phải được thực hiện theo một trình tự nghiêm ngặt để hoàn thành một mục đích

Nguy ý phản ứng tự động với các kích thích - ít hoặc không suy nghĩ về phản ứng liên quan

Lập trình khai báo

Một chương trình bao gồm một tập hợp các quy tắc và sự kiện có thể được sử dụng bởi một công cụ suy luận để đạt được các kết quả thực khác kết luận

Lập trình thủ tục so với lập trình khai báo

Lập trình thủ tục

```
Nếu (x.equals("tuyết"))  
    system.out.print("Nó có màu trắng.");  
  
Nếu không thì nếu (x.equals("grass"))  
    system.out.print("Nó có màu xanh lá cây.");  
  
Nếu không thì nếu (x.equals("sky"))  
    system.out.print("Nó có màu vàng.");  
  
Khác  
    system.out.print("Tôi không chịu nổi.");
```

Lập trình khai báo

```
printColor(X) :- color(X,Y), !, write("Nó  
    là "), viết(Y), viết(".").  
  
printColor(X) :- write("Tôi không chịu nổi.").  
  
màu sắc (tuyết, trắng).  
  
màu sắc (bầu trời, vàng).  
  
màu sắc(X,Y) :- đư ợc tạo thành từ(X,Z), màu sắc(Z,Y).  
  
đư ợc làm từ (cỏ, thảm thực vật).  
  
màu sắc (thực vật, xanh lá cây).
```


Tại sao biểu diễn tri thức và lý luận

Tại sao phải biểu diễn kiến thức? Chúng

ta có thể thêm các nhiệm vụ mới và dễ dàng khiến chúng phụ thuộc vào kiến thức trước đó

Chúng ta có thể mở rộng hành vi hiện có bằng cách thêm vào những niềm tin mới. Chúng ta có thể gỡ lỗi hành vi sai bằng cách xác định vị trí sai niềm tin của hệ thống

Chúng ta có thể giải thích và biện minh một cách ngắn gọn về hành vi của hệ

thống. Tại sao phải

lý luận? Chúng tôi muốn hành động phụ thuộc vào hệ thống

tin tư ởng về thế giới, trái ngược với hệ thống đư ợc thể hiện rõ ràng.

Yêu cầu về kiến thức Cơ sở đại diện

Nó phải có khả năng biểu diễn kiến thức đã cho ở mức độ sâu vừa đủ.

Nó phải bảo tồn các đặc điểm cơ bản của kiến thức, chẳng hạn như tính đầy đủ, khả năng tiếp cận, tính minh bạch, tính tự nhiên, v.v.

Nó phải có khả năng suy ra kiến thức mới.

Nó phải có khả năng đưa ra lý luận và giải thích.

Nó phải đủ khả năng thích ứng để lưu trữ các bản cập nhật và hỗ trợ phát triển gia tăng.

Biểu diễn kiến thức chung

Phương pháp

Logic

Quy tắc bậc nhất

Quy tắc

Quy tắc sản xuất

Khung

Mạng ngữ nghĩa

Kiến thức thực tế

Hằng số

Biến

Chức năng

Vị ngữ (Mối quan hệ, vị từ)

Các hàm đặc biệt chỉ trả về các giá trị Boolean (đúng hoặc sai)

Công thức (Well Formed) (biểu thức)

Chuỗi ký hiệu được tạo ra bởi ngôn ngữ chính thức

Giới thiệu về Logic bậc nhất

Một logic hình thức được tạo ra bằng cách kết hợp các vị từ logic và logic mệnh đề.

Logic mệnh đề được sử dụng để khẳng định các mệnh đề, đó là các câu lệnh đúng hoặc sai. Nó chỉ xử lý giá trị chân lý của các câu lệnh hoàn chỉnh và không xem xét các mối quan hệ hoặc sự phụ thuộc giữa các đối tượng.

Logic vị từ là một phần mở rộng và tổng quát của logic mệnh đề. Các công thức của nó chứa các biến có thể được lượng hóa. Hai lượng từ phổ biến là tồn tại.

và các quantifiers phổ quát. Các biến có thể là các yếu tố trong vũ trụ, hoặc có thể là các mối quan hệ hoặc chức năng trên vũ trụ.

Cú pháp Logic bậc nhất

Ký hiệu

Ký hiệu biến: x, y, z, A, B, C .

Ký hiệu hàm: $f(), g(), h(), \text{bestFriend}(), \dots$

Ký hiệu vị ngữ: $P(), Q(), R(), \text{OlderThan}(), \dots$

Ký hiệu logic: $\neg, \wedge, \vee, \rightarrow, \leftrightarrow, \models, \vdash$

Ký hiệu dấu câu: $(,), \text{và}, \text{.}$

Cú pháp Logic bậc nhất

Thuật ngữ (hạng từ)

Một thuật ngữ đư ợc sử dụng để chỉ một cái gì đó trong thế giới

Biến là các thuật ngữ và $f(T)$ là một thuật ngữ, trong đó f là một hàm và T là một chuỗi n điều khoản.

Công thức (biểu thức) Công thức đư ợc

sử dụng để diễn đạt một mệnh đề (mệnh đề) Công thức nguyên tử - $P(T)$

là một công thức nguyên tử, trong đó P là một vị ngữ và T là một chuỗi của n số hạng.

Nghĩa đen (thể hiện) - công thức nguyên tử (ví dụ: $P(x)$) và công thức nguyên tử phủ định (ví dụ: $\neg P(x)$) Công thức đư ợc hình thành tốt (wffs, biểu thức có cấu trúc chuẩn) - nghĩa đen là wffs và wffs đư ợc kết nối hoặc lư ợng hóa cũng là wffs.

Câu (câu mệnh đề)

Một câu là bất kỳ công thức nào trong đó tất cả các biến đều nằm trong phạm vi của các lư ợng từ tư ơ ng ứng.

Mệnh đề hợp (Mệnh đề hợp) Một wff bao

gồm một nghĩa đen hoặc một sự phân tách các nghĩa đen (các nghĩa đen đư ợc kết nối bằng OR).

Các vị từ trong Logic bậc nhất

Các vị ngữ thể hiện các thuộc tính, mối quan hệ hoặc điều kiện có giá trị giữa các đối tượng

Chúng mô tả trạng thái của thế giới hoặc khẳng định sự thật về các thực thể trong phạm vi

Các hàm đặc biệt chỉ trả về các giá trị Boolean (đúng hoặc sai) Ví dụ:

`IsHuman(x)`

`IsParent(x, y)`

Các lượng tử trong Logic bậc nhất

Các lượng tử cho phép chỉ rõ các phát biểu về toàn bộ hoặc sự tồn tại của các đối tượng trong phạm vi.

Lượng tử phổ quát (\forall) (lượng tử phổ quát)

Biểu thị rằng một câu lệnh có giá trị đối với tất cả các đối tượng trong miền

Ví dụ: $\forall x P(x)$ có nghĩa là "với mọi x , $P(x)$ là đúng", chỉ ra rằng tính chất P đúng với mọi đối tượng x trong miền.

Lượng tử hiện sinh (\exists) (lượng tử tồn tại)

Biểu thị rằng một câu lệnh có giá trị đối với ít nhất một đối tượng trong lãnh địa

Ví dụ: $\exists x P(x)$ có nghĩa là "tồn tại một x sao cho $P(x)$ là đúng", chỉ ra rằng có ít nhất một đối tượng x trong miền mà thuộc tính P giữ nguyên

Các phép kết nối trong Logic bậc nhất

Liên từ (\wedge) (phép toán hội)

Biểu diễn phép "AND" logic giữa hai mệnh đề.

sự kết hợp của hai mệnh đề chỉ đúng nếu cả hai mệnh đề là đúng

Ví dụ: Nếu $P(x)$ biểu diễn "x màu đỏ" và $Q(x)$ biểu diễn "x tròn", thì $P(x) \wedge Q(x)$ biểu diễn "x màu đỏ và tròn".

Phân đoạn (\vee) (phép toán tuyển)

Biểu diễn phép "OR" logic giữa hai mệnh đề. Sự phân tách của hai mệnh đề là đúng nếu ít nhất một trong các mệnh đề là

ĐÚNG VẬY.

Ví dụ: Nếu $P(x)$ biểu diễn "x là một con mèo" và $Q(x)$ biểu diễn "x là một con chó", thì $P(x) \vee Q(x)$ biểu diễn "x là một con mèo hoặc một con chó".

Các phép kết nối trong Logic bậc nhất

Hàm ý (\rightarrow) (phép kéo theo)

Biểu thị mối quan hệ logic “nếu-thì” giữa hai mệnh đề.

Suy luận $P \rightarrow Q$ là sai khi và chỉ khi P đúng, nhưng Q sai.

Nếu không thì nó luôn luôn đúng.

Ví dụ: Nếu $P(x)$ biểu thị “ x là động vật có vú” và $Q(x)$ biểu thị “ x sản xuất sữa”, thì $P(x) \rightarrow Q(x)$ biểu thị “nếu x là động vật có vú, thì x sản xuất sữa”.

Phủ định (\neg) (phép toán phủ định)

Biểu thị “KHÔNG” logic hoặc phủ định của một mệnh đề. Nó đảo ngược giá trị thực của mệnh đề.

Ví dụ: Nếu $P(x)$ biểu diễn “ x thông minh”, thì $\neg P(x)$ biểu thị “ x không thông minh”

Ví dụ về Logic bậc nhất

Ví dụ C

= Bob uống cà phê D = Bob

ăn bánh C D: Bob

uống cà phê VÀ Bob ăn bánh C D: Bob uống cà phê

HOẶC Bob ăn bánh $\neg C$: Bob KHÔNG uống cà phê C

D: Nếu Bob uống cà phê, thì Bob ăn

bánh CD: Nếu Bob uống cà phê, thì Bob ăn bánh và ngược

lại \leftrightarrow

Các câu lệnh có thể được nối với nhau bằng các ký hiệu ở trên để cung

cấp các câu lệnh logic phức tạp hơn Thích (Bob,

cà rốt) Thích (Bob, bắp cải) Các câu lệnh

ngữ nghĩa có thể được mở rộng để bao gồm các ý tưởng tổng quát hơn

X (Thích(Bob, X) Ăn (Bob, X))

ngụ ý rằng Bob ăn mọi thứ mà anh ấy thích

Đại diện cho thủ tục/quan hệ Kiến thức

Quy tắc sản xuất

Nếu <tiền đề>, thì <kết luận>

Nếu <điều kiện>, thì <hành

động> Các quy tắc cho phép tạo ra kiến thức mới dưới dạng các sự kiện ban đầu không có sẵn như ng có thể suy ra từ các phần kiến thức khác. Các sự kiện này được tạo ra khi các kết luận của các quy tắc được áp dụng.

Mạng ngữ nghĩa

Mô tả đồ họa về kiến thức bao gồm các nút và liên kết mang thông tin ngữ nghĩa về mối quan hệ giữa các nút.

Khung

Tổ chức kiến thức theo nguyên nhân và kết quả mối quan hệ. Các ô của khung chứa các mục như quy tắc, sự kiện, tài liệu tham khảo, v.v.

Lý luận: Các loại Logic

Diễn giải

Quá trình lý luận trong đó kết luận nhất thiết phải theo sau từ những tiền đề đã nêu; lý luận từ tổng quát đến cụ thể.

Nếu X đúng và nếu X đúng thì Y cũng đúng, khi đó Y đúng.

Quy nạp (Induction)

Quá trình lý luận trong đó đưa ra kết luận về tất cả các thành viên của một lớp học chỉ kiểm tra một số ít thành viên của lớp; lý luận từ cái riêng đến cái chung.

Đối với tập hợp các đối tượng, $X=\{a,b,c\}$, nếu tính chất P đúng với a, b và c, thì P đúng với mọi X.

Bắt cóc (lý luận suy đoán)

Một dạng logic diễn dịch chỉ cung cấp “suy luận hợp lý”.

Sử dụng thống kê và lý thuyết xác suất, phép suy luận quy nạp có thể đưa ra suy luận có khả năng xảy ra cao nhất trong số nhiều suy luận có thể có.

Nếu Y đúng và X suy ra Y thì X đúng.

Lý luận: Chuỗi tiến

Để chứng minh X , trong đó X có dạng $A \rightarrow C$, hãy tìm một tiên đề hoặc định lý dạng $A \rightarrow B$ và biến đổi bài toán thành bài toán chứng minh $B \rightarrow C$.

Bắt đầu với một số sự kiện và áp dụng các quy tắc để tìm ra tất cả các kết luận có thể (dựa trên dữ liệu)

Dữ liệu thường được nhập trước khi hệ thống bắt đầu suy luận quá trình.

Các quy tắc thường được kiểm tra riêng lẻ.

Các quy tắc có liên quan được nhóm lại với nhau để giúp hệ thống dễ viết và xác thực hơn

Quy tắc chỉ được kích hoạt khi tất cả thông tin liên quan đến quy tắc đó là có sẵn.

Công cụ suy luận không được lập trình để đặt câu hỏi và thu thập thông tin mới trong khi chương trình đang chạy.

Có thể đạt được nhiều kết luận

Lý luận: Chuỗi tiến

Các bước thực hiện:

1. Xem xét các sự kiện ban đầu và lưu trữ chúng trong bộ nhớ làm việc của cơ sở kiến thức.
2. Kiểm tra phần trước của các quy tắc.
3. Nếu tất cả các điều kiện đều khớp, hãy hủy bỏ quy tắc.
4. Nếu chỉ có một quy tắc, hãy thực hiện như sau:
 - a) Thực hiện các hành động cần thiết
 - b) Sửa đổi bộ nhớ làm việc và cập nhật thông tin
 - c) Kiểm tra các điều kiện mới
5. Nếu có nhiều hơn một quy tắc được chọn, hãy sử dụng giải quyết xung đột chiến lược để chọn quy tắc phù hợp nhất và chuyển sang Bước 4
6. Tiếp tục cho đến khi tìm thấy và thực hiện được quy tắc phù hợp
7. Nêu giải pháp, hoặc nếu không có giải pháp thì nêu rằng cơ sở quy tắc không đủ

Lý luận: Chuỗi tiến

Ví dụ: Xét một hệ thống có ba quy tắc

1. Nếu ai đó là sinh viên năm thứ ba, thì họ cần một công việc
2. Nếu ai đó là sinh viên năm thứ ba thì họ sống trong khuôn viên trường.
3. Nếu ai đó cần việc làm, họ sẽ xem quảng cáo việc làm.

Giả sử chúng ta đưa dữ liệu sau vào bộ nhớ:

John là sinh viên năm thứ ba.

Chuyện gì sẽ xảy ra?

Lưu ý: Trong chuỗi chuyển tiếp, vì hệ thống liên tục cảnh báo dữ liệu mới, hệ thống sẽ tìm kiếm tất cả các quy tắc cho bất kỳ điều kiện nào không đúng trước đây nhưng hiện tại là đúng. Sau đó, nó thêm kết luận của chúng vào bộ nhớ.

Lý luận: Chuỗi tiến

Ví dụ: Giải pháp

Quy tắc 1 và 2 có các điều kiện phù hợp với thực tế mới này (John là sinh viên năm thứ ba.)

Vậy là hệ thống sẽ ngay lập tức tạo và thêm vào 2 dữ kiện:

John cần việc làm.

John sống trong khuôn viên trường.

Những sự kiện này lần lượt có thể kích hoạt các quy tắc. Khi mỗi sự kiện xuất hiện, hệ thống sẽ tìm kiếm thêm nhiều quy tắc được chứng minh là đúng.

Trong trường hợp này, thực tế là John cần một công việc sẽ kích hoạt Quy tắc 3, dẫn đến

việc thêm một sự kiện khác vào bộ nhớ:

John sẽ xem quảng cáo việc làm.

Việc John sống trong khuôn viên trường sẽ không gây ra bất cứ điều gì khác.

Lý luận: Chuỗi ngữ ợc

Để chứng minh X , trong đó X có dạng $A \rightarrow C$, hãy tìm một tiên đề hoặc định lý dạng $B \rightarrow C$ và biến đổi bài toán thành bài toán chứng minh $A \rightarrow B$.

Bắt đầu bằng kết luận mong muốn và tiến hành ngữ ợc lại để tìm ra những sự kiện hỗ trợ (hướng đến mục tiêu).

hệ thống đư ợc điều khiển từ mục tiêu trở lại dữ liệu

Trong chuỗi ngữ ợc, hệ thống không hoạt động cho đến khi đư ợc yêu cầu, tức là mục tiêu đư ợc chỉ định.

Lý luận: Chuỗi ngữ ợc

Các bước thực hiện:

1. Bắt đầu bằng một giả thuyết có thể, H .
2. Lưu trữ giả thuyết H trong bộ nhớ làm việc cùng với các dữ kiện có sẵn.
3. Nếu H nằm trong các sự kiện ban đầu, giả thuyết đợc chứng minh. Chuyển đến Bước 7.
4. Nếu H không có trong các sự kiện ban đầu, hãy tìm một quy tắc R có phần con (hành động) đề cập đến giả thuyết.
5. Lưu trữ R trong bộ nhớ làm việc.
6. Kiểm tra các điều kiện của R và đối chiếu với các dữ kiện hiện có.
7. Nếu khớp, hãy kích hoạt quy tắc R và dừng lại. Nếu không, hãy tiếp tục Bước 4.
8. Cung cấp một kết quả—đó là mục tiêu có thể hoặc không thể đạt đợc đã đạt đợc.

Lý luận: Chuỗi ngữ ợc

Ví dụ: Trong cùng cơ sở kiến thức ba quy tắc mà chúng ta đã sử dụng trong hoạt động trước,

1. Nếu ai đó là sinh viên năm thứ ba, thì họ cần một công việc .
2. Nếu ai đó là sinh viên năm thứ ba, thì họ sống trong khuôn viên trường.
3. Nếu ai đó cần việc làm, họ sẽ xem quảng cáo việc làm.

chúng ta thêm dữ liệu (một sự

thật): John là sinh viên năm thứ ba.

Hệ thống sẽ thực hiện những gì ngay lập tức?

Hệ thống sẽ làm gì khi chúng ta đặt câu hỏi sau: Có ai sẽ xem quảng cáo việc làm không?

Lý luận: Chuỗi ngữ ợc

Ví dụ: Giải pháp:

Trong chuỗi ngữ ợc, vì hệ thống đợc điều khiển theo mục tiêu, hệ thống sẽ không làm gì cả cho đến khi đợc hỏi một câu hỏi, tức là đợc cung cấp một mục tiêu để tìm kiếm hoặc một giả thuyết để kiểm tra.

Khi đợc hỏi câu hỏi

Có ai muốn xem quảng cáo việc làm không?

hệ thống sẽ cố gắng trả lời.

Bước đầu tiên là tìm kiếm một sự kiện đợc đưa ra câu trả lời trực tiếp hoặc một quy tắc mà câu trả lời có thể đợc suy ra

nó tìm kiếm toàn bộ cơ sở kiến thức để tìm các quy tắc có kết luận, nếu đúng, sẽ trả lời câu hỏi

Trong ví dụ này, KHÔNG có dữ kiện nào trực tiếp đưa ra câu trả lời.

Lý luận: Chuỗi ngữ ợc

Ví dụ: Giải pháp:

có một quy tắc mà kết luận của nó, nếu đúng, sẽ cung cấp một trả lời: Quy tắc 3

Tiếp theo hệ thống sẽ kiểm tra các điều kiện của

Quy tắc 3 Có ai cần việc làm không?

Hệ thống sẽ tìm kiếm một sự kiện trả lời trực tiếp hoặc để có một quy tắc.

Không có sự kiện nào

Quy tắc 1 có liên

quan Sau đó, hệ thống sẽ kiểm tra các điều kiện của Quy tắc 1.

Có sinh viên năm thứ ba không? Có

một sự kiện trả lời cho câu hỏi này: John là sinh viên năm thứ ba

chúng ta đã chứng minh Quy tắc 1 và bằng cách đó cũng chứng minh đợc Quy tắc 3 và trả lời câu hỏi ban đầu.

Lý luận: Chuỗi ngữ ợc

Ví dụ: Giải pháp:

Sẽ thế nào nếu chúng ta không biết John là sinh viên năm thứ ba

Nếu không có quy tắc nào đưa ra điều này như một kết luận và điều này hiện không phải là
được biết, sau đó hệ thống nói ngữ ợc sẽ yêu cầu người dùng cung cấp
trả lời

Các hệ thống nối tiếp ngữ ợc sẽ tham gia vào cuộc đối thoại với

người dùng.

Chuỗi ngữ ợc chỉ giải quyết được mục tiêu cụ thể

nó không xác định được John sống trong khuôn viên trường vì điều này không phải
liên quan.

Chuỗi chuyển tiếp sẽ tìm ra mọi kết luận có thể

Lý luận: Chuỗi ngữ ợc

Sử dụng chuỗi ngữ ợc

Reason for backward chaining	Examples
There is a clear set of statements, which must be confirmed or denied.	Is machine one causing the quality control problem?
A large number of questions could be asked of the user, but typically only a few are necessary to resolve a situation.	When processing of a motor claim for vandalism; it is not necessary to know about personal injuries.
It is desirable to have interactive dialogue with the user.	Asking machine operator detailed questions about suspect machinery.
Rule execution depends on data gathering which may be expensive or difficult.	Real-time observations by the user.

So sánh giữa Forward và Chuỗi ngược

Các yếu tố hữu ích có thể được sử dụng để xem xét sự lựa chọn giữa chuỗi tiến hoặc lùi trong Hệ thống chuyên gia

Factor	Reason
The logical reasoning process.	Some processes naturally use forward chaining logic, e.g. using errors in computer systems to determine the cause of the error.
What are the inputs and where do they come from?	Where there are few inputs but many outputs, then forward chaining will be more efficient.
What are the outputs and where to they go?	Where there are few outputs, then backward chaining is more appropriate.
Hypothesis driven.	Backward chaining is relatively efficient where hypotheses are involved.

So sánh giữa Forward và Chuỗi ngược

Ví dụ về chuỗi tiến và chuỗi lùi

Use forward chaining	Use backward chaining
Sensor indicates machine failure; need to find out what happens next.	Defect observed in product; need to locate faulty machine.
User types erroneous input for insurance claim; need to alert user.	Suspect an overpayment on an insurance claim; need to check form for erroneous input.
Stock value suddenly drops; need to predict market responses.	FTSE industrials drop; need to know if a particular stock will be affected.

FTSE: Chỉ số chứng khoán Financial Times

Phát triển hệ thống dựa trên quy tắc

Những vấn đề chính trong việc xây dựng hệ thống dựa trên tri thức
(KBS) thiếu phương tiện giải
thích tính dễ vỡ (tính dễ vỡ của hệ thống)
không thể giải quyết vấn đề nằm ngoài phạm vi của
cung cấp kiến thức

Giải thích Tiện nghi

Giải thích cho người sử dụng hệ thống cách thức hệ thống đưa ra quyết định

tại sao các quy tắc cụ thể đã được áp dụng

Đầu ra từ các hệ thống chuyên gia (ES) phải có khả năng cung cấp mức độ giải thích tương tự như các chuyên gia con người

Người dùng muốn hiểu lý do tại sao họ được đưa ra lời khuyên

Người dùng muốn biết lý do tại sao một số hành động nhất định đã được thực hiện khuyến khích

Người dùng muốn thấy các vấn đề liên quan đến các hành động thay thế khác

Giải thích do ES cung cấp có thể không chi tiết bằng giải thích do chuyên gia cung cấp

ES chỉ có kiến thức trong một lĩnh vực chủ đề rất cụ thể

Câu trả lời không thể liên quan đến bất kỳ bối cảnh rộng hơn nào

Cơ sở kiến thức có thể không cung cấp đủ thông tin chi tiết trong lĩnh vực chủ đề cụ thể

Giải thích Tiện nghi

Theo dõi quy tắc

Một dấu vết 'cách' cho phép người dùng tìm ra cách ES đã đến một kết luận

Dấu vết 'tại sao' giúp người dùng hiểu lý do tại sao một câu hỏi cụ thể được hỏi

Cung cấp cho người dùng một số phản hồi về cách ES hoạt động

Chỉ đơn giản là cung cấp một chuỗi lý luận

liên kết vấn đề và giải pháp

ES có thể giải thích tại sao một kết luận cụ thể nào đó được đưa ra.

không nhất thiết phải biết lý do tại sao các quy tắc thích hợp lại có trong cơ sở kiến thức

Hệ thống có thể nêu **NẾU** điều gì đó **THÌ** điều gì đó khác xảy ra, nhưng không nêu lý do tại sao những sự kiện đó được liên kết

Giải thích Tiện nghi

Xây dựng Văn bản Giải thích thành Hệ thống Chuyên gia (ES)

Có thể tích hợp văn bản giải thích vào ES để giúp người dùng hiểu các kết quả đầu ra được cung cấp

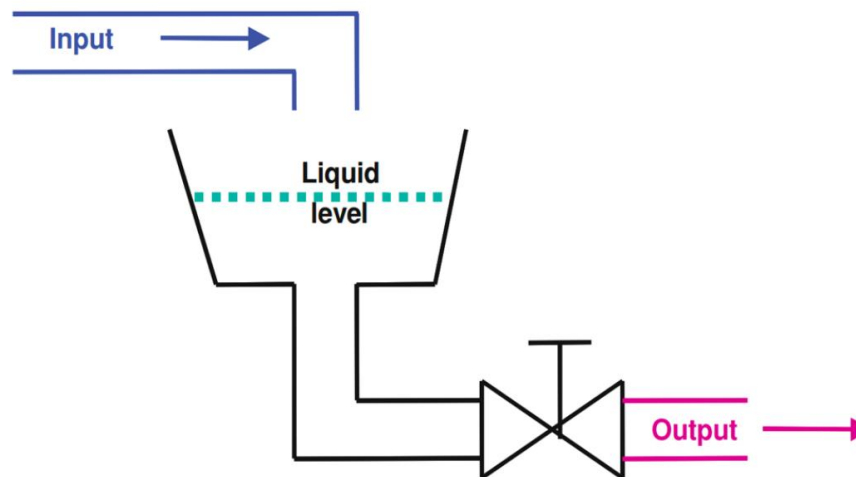
cấp Ví dụ:

Hỏi: Điều gì sẽ xảy ra với mực nước khi van mở thêm một chút?

A: Nó sẽ đi xuống.

H: Tại sao?

A: Bởi vì lưu lượng đầu ra sẽ lớn hơn lưu lượng đầu vào



Giải thích Tiện nghi

Xây dựng văn bản giải thích thành ES Ví dụ: Hỏi:

Tại sao?

A: Bởi vì lưu lượng nước sẽ thoát ra qua đường ống đầu ra nhiều hơn n.

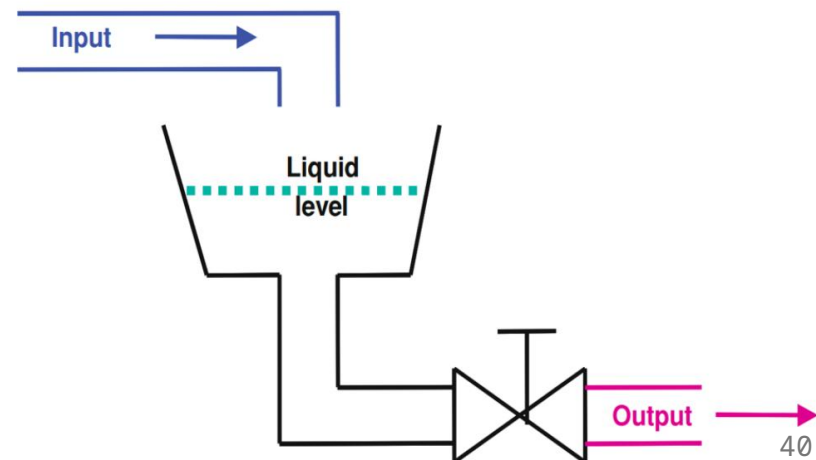
H: Tại sao lưu lượng nước thoát ra qua đường ống đầu ra lại nhiều hơn n?

A: Bởi vì khi mở van sẽ làm lỗ mở rộng hơn n, cho nhiều nước chảy qua hơn n.

H: Tại sao điều này xảy ra. . .?

Sau khi đưa tất cả các giải thích trên vào hệ thống, các câu hỏi có thể được hỏi như sau:

Bạn đã đi đến câu trả lời đó như thế nào? Sau đó, ES có thể cung cấp các giải thích về quy tắc để cho thấy câu trả lời được rút ra như thế nào



Giải thích Tiệm nghi

Xây dựng Văn bản Giải thích thành ES Ví

dụ: Ngư ời

dùng: Điều gì xảy ra với mực nư ớc khi van mở?

Hệ thống: Mực nư ớc sẽ giảm xuống.

Ngư ời dùng: Làm sao bạn biết điều này?

Hệ thống: Bởi vì tôi áp dụng Quy tắc 1, quy định khi van mở, nhiều nư ớc hơ n sẽ đư ợc giải phóng. Sau đó, tôi áp dụng Quy tắc 2, quy định khi nhiều nư ớc hơ n đư ợc giải phóng, mực nư ớc sẽ giảm.

Giải thích Tiện nghi

Nguy cơ của những lời giải thích thiếu thông minh

không có cơ chế nào đảm bảo rằng khuyến nghị này là

phù hợp với hoàn cảnh cá nhân của bạn.

Khi các hệ thống phát triển về mặt trí thông minh, chúng được trao nhiều

trách nhiệm hơn

không nên quá tin tưởng vào hệ thống

Người dùng vẫn có trách nhiệm kiểm tra tính hợp lý của bất kỳ câu trả lời nào

Việc thêm các tiện ích giải thích kém chất lượng hoặc đơn giản có thể tạo ra

sự tin tưởng không đáng có vào một hệ thống không đảm bảo điều đó

Điều này có thể khiến người dùng đưa ra quyết định sai lầm

Trí thông minh biểu kiến có thể vượt xa mức độ hiểu biết thực sự

ES chỉ tốt khi có cơ sở quy tắc tốt, và nếu điều này không đúng thì các

giải pháp từ hệ thống cũng sẽ sai

Cơ sở giải thích kém chất lượng có thể khuyến khích người dùng chấp nhận các khuyến nghị

sai lầm.

Độ giòn

Mức độ thông minh biểu kiến vượt quá mức độ thông minh thực sự.

Khi vấn đề mới đòi hỏi kiến thức không có trong hệ thống thì hệ thống sẽ không thể giải quyết được vấn đề.

Hệ thống đề xuất một giải pháp lỗi

Việc đưa vào một cơ sở giải thích không thông minh sẽ

khuyến khích người dùng chấp nhận kết quả đầu ra bị lỗi.

Công trình gần đây về việc xác định các thuật ngữ học đang giúp khắc phục vấn đề giòn.

xác định một bản thể học về giới hạn của kiến thức chứa đựng
trong một hệ thống

một ES cũng có thể nhận ra những hạn chế về kiến thức của mình
căn cứ

Đánh giá dựa trên kiến thức Hệ thống

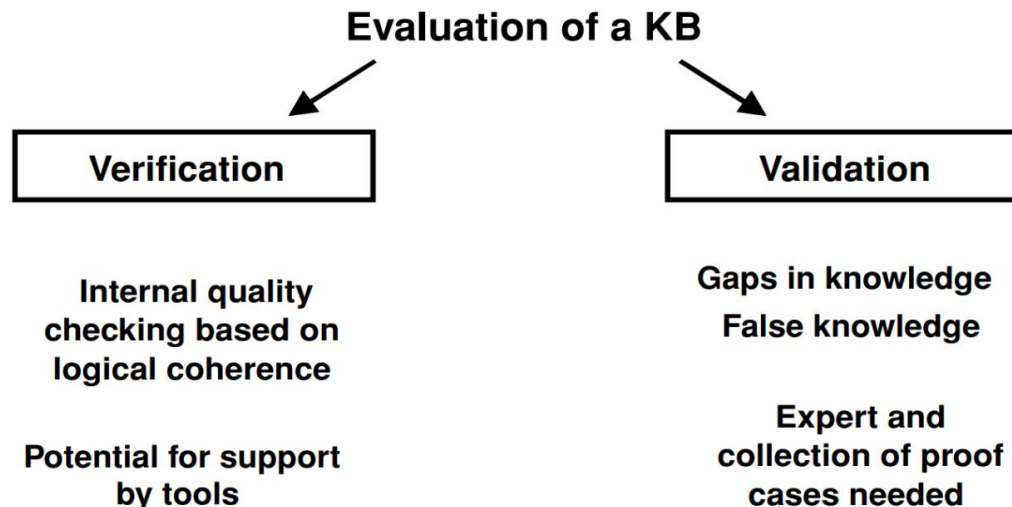
Một nỗ lực để đánh giá giá trị tổng thể của KBS

Kiểm tra không chỉ xem KBS có mức hiệu suất chấp nhận được hay không mà còn xem hệ thống có thể sử dụng được, hiệu quả và tiết kiệm chi phí hay không.

Bao gồm hai thuật ngữ nữa: xác thực và kiểm định

Xác thực (xác thực/nhận) đo lường hiệu suất của KBS

Xác minh (xác minh) kiểm tra xem KBS đã được xây dựng đúng chưa



Đánh giá dựa trên kiến thức Hệ thống

Xác minh: bao gồm việc kiểm tra các nội dung sau

Tính nhất quán cú pháp: kiểm tra xem tất cả các đối tượng trong KB có đúng không
được xác định liên quan đến công cụ suy luận

Sự mạch lạc về mặt logic: phát hiện ra những mâu thuẫn về mặt logic

Sự nhất quán theo ngữ cảnh: kiểm tra xem KB có nhất quán với
mô hình của vấn đề.

Loại lỗi có thể được phát hiện bằng cách xác minh

Quy tắc gộp: hai quy tắc có cùng kết luận nhưng là một quy tắc
có thêm điều kiện:

- Quy tắc 1. NẾU A VÀ B VÀ C THÌ X
- Quy tắc 2. NẾU A VÀ B THÌ X

Điều kiện IF không cần thiết: kết luận của hai quy tắc là

và các điều kiện của các quy tắc đều giống nhau ngoại trừ một điều.

- Nguyên tắc 1. NẾU bệnh nhân có các đốm hồng VÀ sốt THÌ bị sởi.
- Quy tắc 2. NẾU bệnh nhân có các đốm hồng VÀ không sốt THÌ
bệnh sởi.

Đánh giá dựa trên kiến thức Hệ thống

Xác thực: đảm bảo rằng phạm vi công việc đư ợc liên kết chính xác và đư ợc phản ánh trong phạm vi kiến thức

- xác định phạm vi công việc

- xác định các trư ờng hợp chứng minh cần sử dụng

- quyết định sử dụng bao nhiêu trư ờng hợp chứng minh

Các trư ờng hợp chứng minh kiểm tra KBS bằng cách đảm bảo rằng kết quả từ KBS phù hợp với kết quả mà chuyên gia đã dự đoán.

KBS sẽ đư ợc xác thực khi các trư ờng hợp chứng minh trùng khớp với trư ờng hợp của chuyên gia.

Số lư ợng trư ờng hợp chứng minh cần thiết phụ thuộc vào các biến số như số lư ợng quy tắc trong KBS và độ chính xác yêu cầu từ đầu ra.

Đánh giá dựa trên kiến thức Hệ thống

Xác thực: một số biện pháp được xác định trước để kiểm tra đầu ra của hệ thống:

Độ chính xác: hệ thống phản ánh thực tế tốt như thế nào

Tính đầy đủ: có bao nhiêu kiến thức cần thiết được đưa vào cơ sở kiến thức.

Tính thực tế: liệu KBS có cung cấp các giải pháp thực tế hay không

Độ nhạy: cách thức thay đổi trong cơ sở kiến thức ảnh hưởng đến chất lượng của đầu ra.

Tính hữu ích: mức độ hữu ích của kết quả đầu ra trong việc giải quyết vấn đề.

Tính hợp lệ (validity): liệu các đầu ra có thể được sử dụng để tạo ra dự đoán chính xác.

Mạng ngữ nghĩa

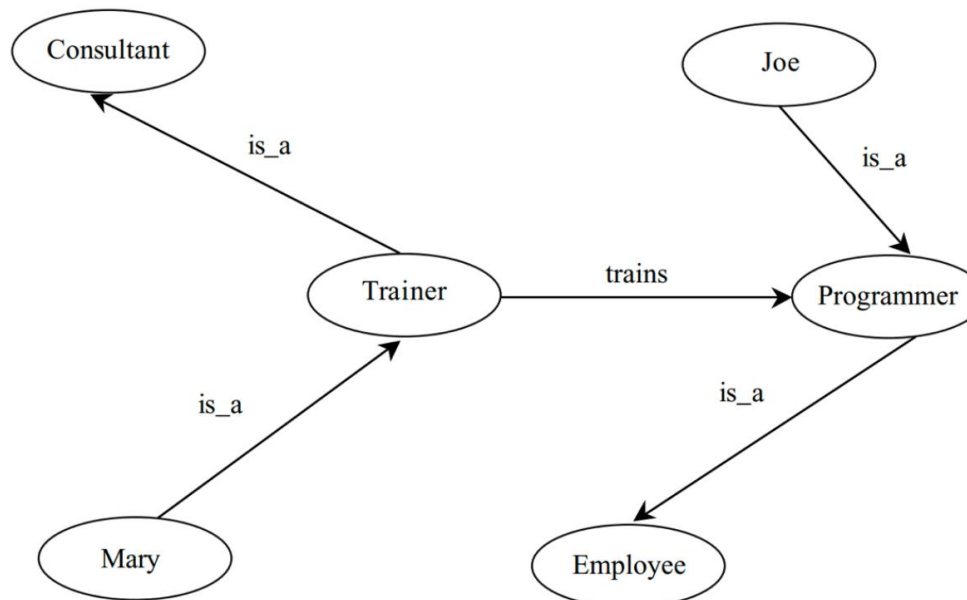
Một trong những sơ đồ biểu diễn kiến thức lâu đời nhất và dễ hiểu nhất

một biểu diễn đồ họa của kiến thức cho thấy các đối tượng và mối quan hệ của chúng

Các đối tượng được hiển thị bằng các nút

Các liên kết giữa các nút mô tả mối quan hệ giữa hai

các đối tượng



Mạng ngữ nghĩa

Di sản

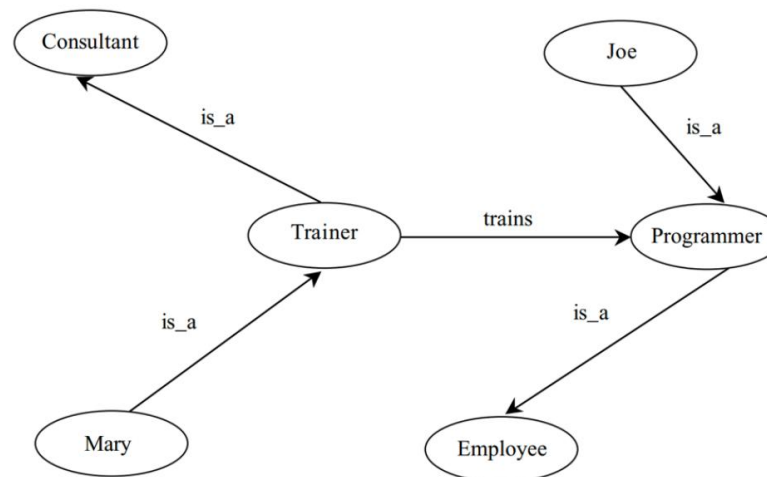
cách một đối tượng thừa hưởng các thuộc tính của một đối tượng khác.

Bản chất phân cấp của sơ đồ giúp giải thích các yếu tố của mạng

Ví dụ:

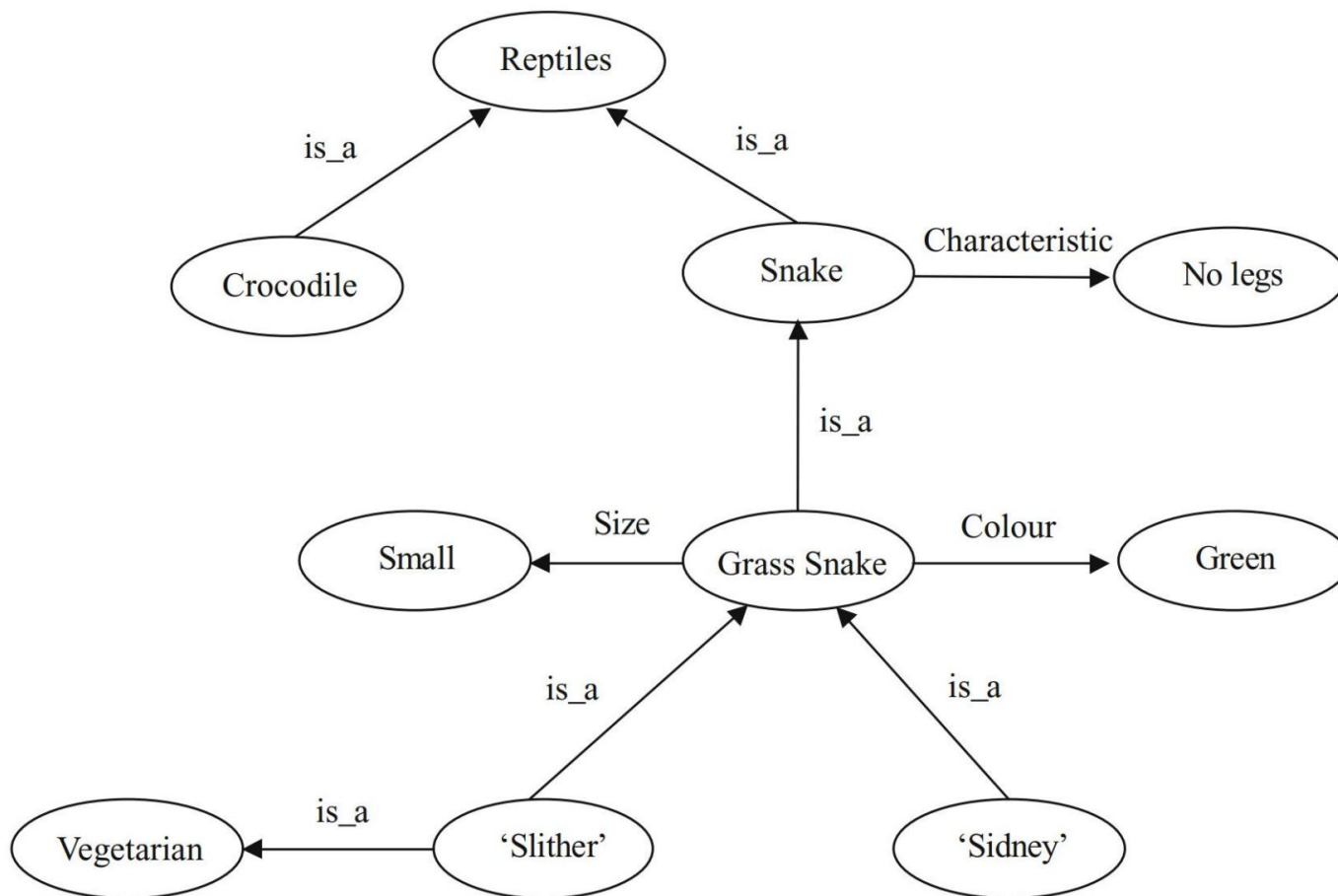
Mary, với tư cách là một người đào tạo, thừa hưởng các đặc tính của nhà tư vấn lớp học

Joe, với tư cách là một lập trình viên, thừa hưởng các thuộc tính của nhân viên lớp học



Mạng ngữ nghĩa

Di sản



Mạng ngữ nghĩa

Lý luận với Mạng ngữ nghĩa

Rất khó để lý luận từ một mạng ngữ nghĩa

Bởi vì một công cụ suy luận phải hiểu loại liên kết nào
tồn tại giữa các nút

không có hạn chế nào đối với các liên kết được phép

Nếu một công cụ suy luận lý luận với mạng ngữ nghĩa,

khi đó nó phải có một số hiểu biết về các từ biểu diễn mối quan hệ
giữa các nút (ví dụ: `is_a`, `eats`, `colour`,..)

Phần lớn lý luận với mạng ngữ nghĩa ở dạng

suy ra các liên kết gián tiếp giữa các khái niệm dựa trên các liên kết trực
tiếp được nêu rõ ràng

khó để một công cụ suy luận đạt được trong thực tế

Mạng ngữ nghĩa thường chỉ được sử dụng như một công cụ giao tiếp
giữa kỹ sư kiến thức và chuyên gia trong lĩnh vực

Mạng ngữ nghĩa

Ưu điểm

Chúng có xu hướng là một phương pháp mạnh mẽ và thích ứng để biểu diễn kiến thức

nhiều loại đối tượng khác nhau có thể được đưa vào mạng

Mạng được biểu diễn dưới dạng đồ họa nên tư duy đối dễ hiểu.

Có thể được sử dụng như một công cụ giao tiếp chung giữa

kỹ sư kiến thức và chuyên gia ngôn ngữ trong giai đoạn thu thập kiến thức khi thiết kế hệ thống chuyên gia

Mạng ngữ nghĩa

Nhược điểm

Khó có thể thể hiện tất cả các tình huống suy luận khác nhau bằng cách sử dụng mạng

Ít tin cậy hơn các kỹ thuật biểu diễn kiến thức khác

Bởi vì suy luận trở thành một quá trình tìm kiếm trên toàn bộ
sơ đồ

Biểu đồ có thể trở nên rất phức tạp

Phạm vi rộng lớn của các loại liên kết có thể có và cách chúng có thể
kết hợp để tạo thành các liên kết gián tiếp

làm cho mạng dễ bị bùng nổ kết hợp

Gặp khó khăn trong việc liên kết kiến thức thủ tục với các sự kiện được thể
hiện bởi mạng lưu trữ

họ thiếu bất kỳ phương tiện nào để gom các sự kiện liên quan vào các sự kiện liên quan
các cụm

yêu cầu các hoạt động tìm kiếm mở rộng để đi đến kết luận

Mạng ngữ nghĩa

Hạn chế

Trong các biểu diễn mạng ngữ nghĩa, không có ngữ nghĩa chính thức, không có khái niệm thống nhất về ý nghĩa của một cấu trúc biểu diễn nhất định, chẳng hạn như trong logic.

Các mạng ngữ nghĩa có xu hướng dựa vào các thủ tục thao túng chúng

Hệ thống bị giới hạn bởi sự hiểu biết của người dùng về ý nghĩa của các liên kết trong mạng ngữ nghĩa

Các liên kết giữa các nút không giống nhau về chức năng hoặc hình thức.

cần phân biệt giữa các liên kết khẳng định một số mối quan hệ và các liên kết có bản chất là cấu trúc

Khung

Khung là phiên bản đơn giản hóa của mạng ngữ nghĩa trong đó chỉ mỗi quan hệ 'là' được áp

dụng. Cung cấp phương pháp lưu trữ kiến thức, thu thập thông tin cụ thể về một đối tượng trong ES.

Coffee mug FRAME	
IS_A	Mug
COLOUR	
CAN_HOLD_LIQUID	True
NUMBER_OF_HANDLES	Default = 1
SIZE	Range: Small, Medium, Large
PURPOSE	Value : drinking coffee
COST	Demon (£ needed)
MATERIAL	Default = pottery

Khung

Trong cấu trúc khung, các khe (tức là các hàng) có thể:

lưu trữ thông tin chi tiết của từng đối

tượng dữ liệu cung cấp liên kết

đến các khung khác chứa mã thủ tục, liên kết đến các ứng dụng khác để lấy dữ liệu

chỉ ra liệu có cần một số thuộc tính nhất định của từng đối tượng trong khung đó hay không

Coffee mug FRAME	
IS_A	Mug
COLOUR	
CAN_HOLD_LIQUID	True
NUMBER_OF-HANDLES	Default = 1
SIZE	Range: Small, Medium, Large
PURPOSE	Value : drinking coffee
COST	Demon (£ needed)
MATERIAL	Default = pottery

Khung

Các mức trong một khung:

Mức cao nhất trong một khung theo nghĩa đen là FRAME: lưu trữ tên của khung cụ thể

Bên dưới KHUNG là các KHE, với mỗi khe cung cấp thông tin về một trong các thuộc tính của khung đó

Trong khe, FACET cung cấp thông tin chi tiết về từng thuộc tính bao gồm giá trị, phạm vi, giá trị mặc định, giá trị tính toán

DỮ LIỆU cung cấp thông tin cụ thể về từng thuộc tính

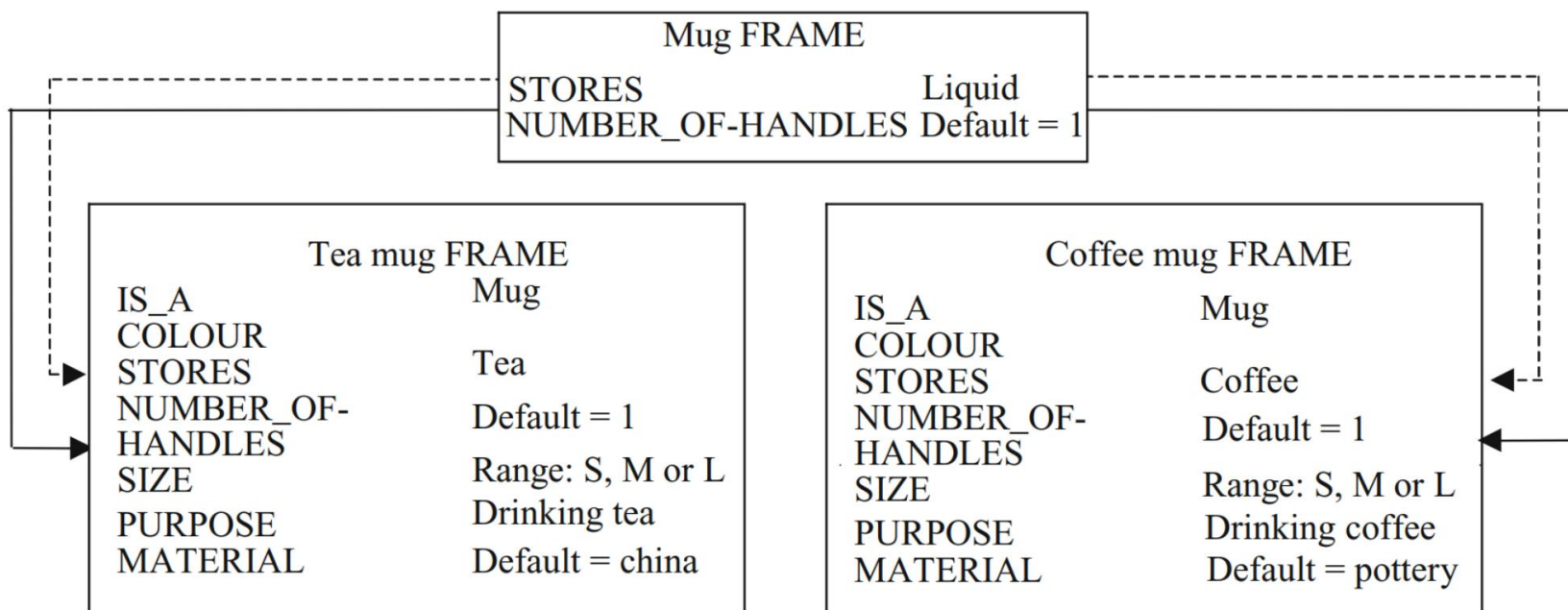
SỐ_BÁNH_TRĂNG là 4 trong một khung mô tả một phương tiện cơ giới

Khung

Di truyền:

Các khung có thể kế thừa các thuộc tính của các khung khác, theo thứ bậc kết cấu

các đối tượng thấp hơn trong hệ thống phân cấp tự động kế thừa nội dung của các khung ứng, trừ khi dữ liệu này bị ghi đè



Khung

Ưu điểm:

được thể hiện dưới dạng bảng, giúp thông tin dễ dàng đọc

lưu trữ các giá trị mặc định

sử dụng các giá trị mặc định trong quá trình suy luận

Nếu sau đó, giá trị mặc định được phát hiện là không chính xác, thì hệ thống có thể ghi đè giá trị mặc định và sau đó chạy lại lý luận của nó

được cấu trúc theo thứ bậc và do đó cho phép phân loại dễ dàng kiến thức

giảm độ phức tạp bằng cách cho phép xây dựng hệ thống phân cấp các khung ghi lại thông tin rõ ràng bằng các định dạng và cú pháp chung kết hợp kiến thức thủ tục và khai báo hạn chế các

giá trị được phép hoặc cho phép các giá trị được nhập trong một phạm vi cụ thể

Khung

Như ợc điểm:

có thể không hiệu quả khi chạy

vì chúng không cung cấp phư ơ ng pháp lư u trữ hiệu quả nhất
dữ liệu cho máy tính

có thể dẫn đến 'sốt thủ thuật'

yêu cầu rõ ràng là tập trung vào việc thực hiện phù hợp

các thủ tục thay vì kiểm tra cấu trúc và nội dung tổng thể của khung

cần đư ợc chú ý trong giai đoạn thiết kế để đảm bảo phân loại phù hợp