

KHÁI QUÁT VỀ TRÍ TUỆ NHÂN TẠO VÀ HỆ CHUYÊN GIA

Hồ Tú Bảo

Viện Khoa học tính toán và Điều khiển

Trong thời gian gần đây, ta thường nghe nói về các thành ngữ 'Trí tuệ nhân tạo' (TTNT) và 'Hệ chuyên gia' (HCG). Là một phần của Tin học hiện đang được đầu tư rất lớn ở các nước phát triển, các kỹ thuật, nội dung và xu thế của ngành khoa học này rất phong phú và thay đổi rất nhanh. Do vậy trong bài viết ngắn này chúng tôi chỉ có thể trình bày sơ bộ về quá trình phát triển, những khái niệm cơ bản nhất và một số nét về tình hình nghiên cứu, ứng dụng của TTNT và HCG trên thế giới.

Thật khó có thể đưa ra một định nghĩa chặt chẽ về TTNT khi mà bản thân khái niệm 'trí tuệ' cũng khá trừu tượng. Tuy nhiên nói nôm na thì đây là ngành khoa học liên quan đến việc làm cho máy tính có thể thực hiện một số hành vi thông minh của con người, mà chủ yếu là khả năng suy luận hình thức và khai thác lượng trí thức loài người đã tích lũy được qua bao thế hệ. Nếu về bản chất, lập trình truyền thống là việc thực hiện các TÍNH TOÁN nhờ các THUẬT TOÁN trên DỮ LIỆU, thì các đặc trưng cơ bản nhất của TTNT là việc thực hiện trên máy tính quá trình SUY LUẬN theo LOGIC bằng TRI THỨC con người.

Sự thăng trầm và hồi sinh của trí tuệ nhân tạo

Có thể nói năm 1956 đã báo hiệu sự ra đời của trí tuệ nhân tạo từ lần gặp gỡ của mười nhà khoa học tại trường hè Dartmouth. Những tiên đoán của họ đã được nhiều nhóm nghiên cứu, nhiều phòng thí nghiệm lớn trên thế giới bèn bí theo đuổi trong hơn một phần tư thế kỷ, dù phải trải qua rất nhiều thất bại, để đến đầu thập kỷ 80 ngành khoa học này tạo được mối quan tâm phát triển ở tất cả các nước tiên tiến, được xem là yếu tố cơ bản sẽ đem đến sự đổi thay về chất của tin học trong tương lai. Tại Hội nghị quốc tế lần thứ bảy về TTNT năm 1981 ở Canada, chỉ còn năm người trong số các nhà khoa học này gặp lại được nhau để cùng nhìn lại chặng đường dài đã qua.

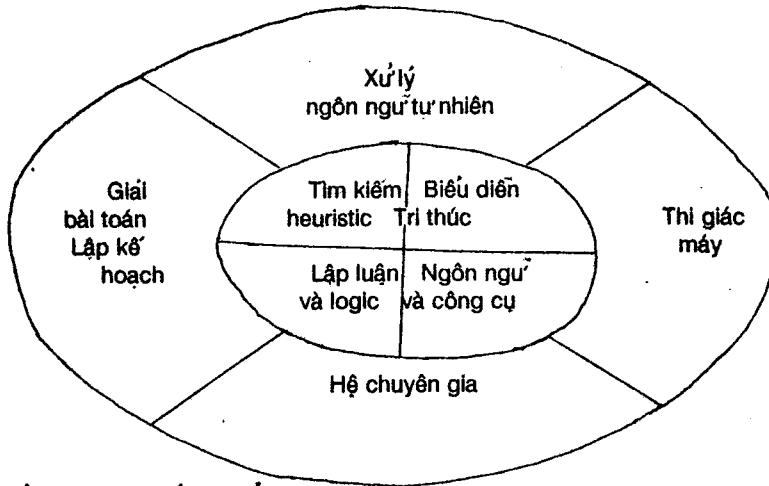
Vào năm 1956, ở buổi đầu của TTNT, người ta đã quá lạc quan và cho rằng những hành vi thông minh của máy có thể đạt được chỉ dựa vào các kỹ thuật lập luận và chẳng bao lâu con người sẽ tạo ra được các chương trình thông minh. Nhưng suốt 15 năm đầu các phòng thí nghiệm đã không đạt được những điều mong muốn tuy rất nhiều cố gắng đã được ghi nhận, chủ yếu trong việc làm các máy dịch tự động hoặc hiểu ngôn ngữ tự nhiên (ELIZA ở MIT, 1966), các chương trình đánh cờ (tiểu biểu là của Samuel ở IBM, 1963), các chương trình giải các bài toán tổng quát (GPS của Newell, 1960), ... Điều đáng kể là 15 năm này đã cho những bài học sâu sắc, rằng TTNT khó hơn nhiều so với sự tưởng tượng ban đầu, rằng cần phải có các tìm kiếm heuristic để chống lại sự bùng nổ tổ hợp, rằng chỉ riêng các kỹ thuật lập luận là không đủ, ... Và từ 1970, nhận thức về vai trò của việc sử dụng trí thức để làm các chương trình thông minh đã đưa vấn đề biểu diễn trí thức lên chiếm vị trí trung tâm trong nghiên cứu TTNT.

Thập kỷ 70 chứng kiến một loạt công trình về các kỹ thuật biểu diễn trí thức. Các kỹ thuật tìm kiếm heuristic cũng dần hoàn thiện. Nhiều chương trình ứng dụng đã mang lại kết quả khả quan trong xử lý ngôn ngữ, thị giác máy, hiểu tiếng nói, hệ chuyên gia. Chính trong giai đoạn này một số hệ chuyên gia đã khẳng định mình: DENDRAL, MYCIN, PROSPECTOR, ... Chúng đã góp một phần đáng kể tạo ra bước ngoặt trên con đường phát triển TTNT.

Và từ đầu thập kỷ 80, giai đoạn thứ ba của sự phát triển này, dù đồng ý hay chưa đồng ý, bỗng nhiên mỗi người đều nói về TTNT và HCG. Tất cả các nước phát triển đều ý thức vai trò quan trọng và đầu tư những khoản tiền khổng lồ cho TTNT, các kết quả nghiên cứu đã bước từ các phòng thí nghiệm ra thị trường, đề án máy tính thế hệ 5 do nước Nhật công bố trong đó TTNT là một nội dung then chốt, các kỹ thuật TTNT được hệ thống hoá, bộ Handbook về TTNT ra đời, các Hội nghị quốc tế, các tạp chí mới xuất hiện, các hệ chuyên gia được xây dựng thử nghiệm khắp nơi, ...

Những nội dung cơ bản của trí tuệ nhân tạo

Việc phân chia này đương nhiên phụ thuộc rất nhiều vào quan niệm của từng người. Tuy nhiên mô hình sau đây của Nilsson(1982), một người tiên phong trong lĩnh vực TTNT, theo chúng tôi là khá hợp lý và không có nhiều thay đổi so với sự phát triển ngày nay. Bốn khối ở vòng tròn trong là bốn trục nghiên cứu cơ bản, còn bốn khối ở vòng ngoài là bốn hướng ứng dụng chính của TTNT.



Các hướng nghiên cứu cơ bản

1. Tìm kiếm heuristic: Những nghiên cứu đầu tiên trong lĩnh vực TTNT tập trung vào việc tạo ra các chương trình tìm nghiệm các bài toán quyết định. Nhớ rằng mỗi khi định đưa ra một quyết định ta thường phải chọn lựa trong rất nhiều tình huống khác nhau. Và sự thay đổi các điều kiện lại có thể mở ra rất nhiều tình huống mới. Thí dụ điển hình cho việc chọn lựa quyết định này là các chương trình đánh cờ tự động: máy tính phải tìm nước đi trong rất nhiều trường hợp loại trừ lẫn nhau, và mỗi khi chọn một quyết định sẽ xuất hiện rất nhiều tình huống mới phải xem xét. Không

gian tìm kiếm thường được biểu diễn bởi một cấu trúc cây phân cấp. Quá trình tìm kiếm trên cây chính là việc đi từ trạng thái ban đầu (nút gốc) dọc theo các nhánh của cây đến trạng thái đích (nút lá). Nghĩ rằng có thể nhờ vào sức mạnh của máy tính, ban đầu người ta đã phát triển nhiều phương pháp tìm kiếm mò mẫm (blind search) mà việc chọn đường đi được thực hiện theo một thứ tự xác định nào đấy cho đến khi tìm ra câu trả lời. Hai kỹ thuật cơ bản nhất là tìm kiếm trước hết theo chiều sâu (depth-first search) và tìm kiếm trước hết theo chiều rộng (breadth-first search). Các phương pháp tìm kiếm mò mẫm không sử dụng một chút tri thức nào để dẫn dắt quá trình này. Và người ta nhanh chóng nhận thấy với các bài toán phức tạp (chơi cờ chẳng hạn) việc tìm kiếm lời giải như vậy thường thất bại. Nguyên nhân chính là sự bùng nổ tổ hợp (combinatorial explosion) của số các con đường phải xem xét mà dù có nhanh như máy tính cũng không thể xét kịp. Có thể dễ dàng kiểm tra điều này: Nếu trung bình ở mỗi nút của không gian tìm kiếm có n nhánh rẽ, và ta đang ở độ sâu d trên cây thì số nút cần phải duyệt từ gốc là n^d .

Các phương pháp heuristic nhằm thu hẹp không gian tìm kiếm nhờ việc sử dụng thông tin về bản chất và đặc thù của vấn đề đang xem xét. Heuristic ở đây là các luật suy diễn, các hiểu biết, kinh nghiệm có thể dùng được để hướng dẫn việc tìm kiếm. Ý chủ đạo của phương pháp này là sinh ra và kiểm tra các giả thiết (generating and testing) theo những con đường có nhiều triển vọng được lựa chọn. Tìm kiếm heuristic là một trong những ý tưởng quan trọng trong nghiên cứu TTNT. Đến nay, các nghiên cứu về hướng này đã khá ổn định. Gần đây, nghiên cứu về tìm kiếm heuristic song song vẫn đang được thực hiện tại MIT và tại một số nhóm ở Canada.

2. Biểu diễn tri thức Mục đích của công việc này là biểu diễn tri thức con người dưới những dạng thích hợp để máy có thể sử dụng được chúng. Ở đây, cái đem biểu diễn (tri thức) được giả thiết là đã có, việc phải làm là tìm cách diễn đạt và mã hoá các tri thức này như thế nào? Ta đi qua một số kỹ thuật biểu diễn tri thức cơ bản nhất:

* Logic Tri thức con người thường được biểu diễn bằng các câu nói, chẳng hạn như "mọi con chim đều biết bay". Ta có thể biểu diễn tri thức này như sau:

$$\forall x, \text{Chim}(x) \rightarrow \text{biết-bay}(x)$$

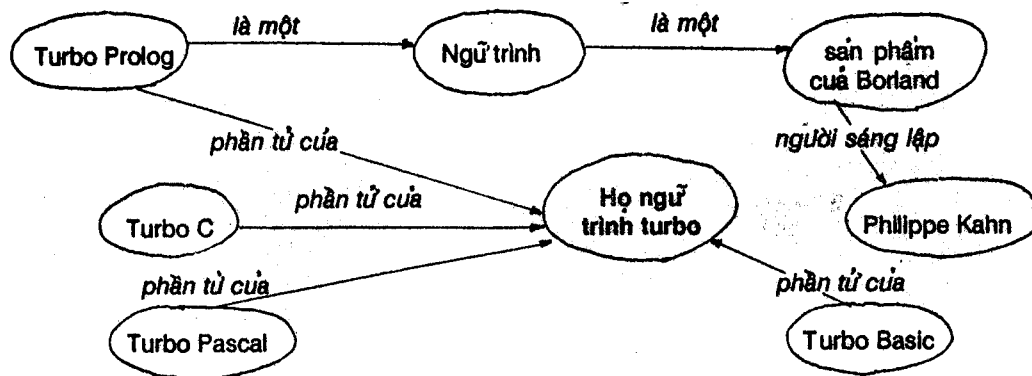
Đây là một luật suy diễn trong logic. Một phương pháp cơ bản để biểu diễn tri thức là dựa vào logic tân từ bậc một. Khi này cơ sở tri thức có thể xem như một tập các công thức logic cho ta mô tả từng phần của thế giới bên ngoài. Các luật dẫn (production rules) là dạng đang được sử dụng phổ biến nhất trong các hệ TTNT và có rất nhiều ưu điểm. Các luật dẫn có dạng tổng quát là:

NEU điều kiện THÌ hành động

trong đó phần điều kiện là tập hợp các ràng buộc, các điều kiện biểu diễn bằng các tân từ, các mệnh đề, và phần hành động thể hiện các kết luận. Các hành động cần thực hiện nếu tất cả các luật đều thỏa mãn. Thí dụ

NEU Đau bụng
và uống nhân sâm
THÌ Chết

* **Mạng ngữ nghĩa** Là cách tiếp cận mô tả các tính chất và quan hệ giữa các đối tượng, sự kiện, khái niệm, tình huống hành động bằng một đồ thị định hướng gồm các nút và các cung. Thí dụ



* **Dạng khung (Frame)** Một phần lớn các hoạt động hàng ngày của con người trong từng lĩnh vực được lặp lại giống nhau, mang cùng một cấu trúc, như các cách mô tả bệnh tật, nấu ăn, đi chợ, sửa máy... Minski (1975) đã đề nghị một cấu trúc dữ liệu phức tạp để biểu diễn chúng gọi là "frame". Thí dụ

Frame: Bệnh ung thư hạch bạch huyết

Triệu chứng:

- Sốt kéo dài hơn ba tuần #LCT,
- Hạch to #LCT,
- Hạch nổi tại một vùng #LPT,
- Lách to #LPT,
- Bạch cầu tăng #CCT,
- Máu lắng tăng không nhiều #CCT,
- Sinh thiết hạch có tế bào đảo cấu trúc #CDT,

Trong đó #LCT, #LPT chỉ các triệu chứng lâm sàng chính và phụ, #CCT, #CPT chỉ các triệu chứng cận lâm sàng chính và phụ, #CDT chỉ triệu chứng cận lâm sàng đặc thù của bệnh này.

* **Đối tượng** Biểu diễn tri thức dưới dạng các đối tượng là một phương pháp hiệu quả khi có thể tìm ra những cấu trúc của tri thức. Trong cách biểu diễn này người ta phải quan niệm và thể hiện tri thức dưới dạng các đối tượng và xây dựng một cấu trúc phân cấp trên tập các đối tượng ban đầu. Mỗi lớp đối tượng là tập các đối tượng có chung nhau một số thuộc tính nào đó. Các tính chất của mỗi đối tượng đều được truyền lại (kế thừa) cho mọi con cháu. Ngoài phần mô tả kể trên, các thủ tục cũng được gắn với các đối tượng để thực hiện các tính toán cần thiết.

Ngoài một số kỹ thuật chủ yếu kể trên, hiện nay đang bắt đầu xuất hiện xu hướng xây dựng các hệ cho phép biểu diễn tri thức đồng thời dưới nhiều dạng khác nhau (hệ lai), mà cách biểu diễn đồng thời bằng luật dẫn và đối tượng đang được nhiều chuyên gia quan tâm và đánh giá cao.

3. Lập luận và Logic Logic là phương pháp hình thức của lập luận. Lập luận bằng máy tính được thực hiện dựa trên nền tảng của logic toán học, như những kết quả đặc sắc và tính chứng minh được của các hệ hình thức của Godel, Church, Tarski,... và các hệ lập luận, trong chừng mực nào đấy chính là sự sự cài đặt của các hệ hình thức lên máy tính trong đó logic mệnh đề và logic tân từ là hai hệ hình thức được dùng rộng rãi nhất. Suy diễn logic là quá trình sử dụng logic để đi đến kết luận, và thường được gọi là quá trình chứng minh tự động các định lý. Nổi tiếng nhất trong các phương pháp chứng minh tự động này là nguyên lý giải (resolution principle) của Robinson (1966) cho phép kiểm tra xem một kết luận có suy ra được hay không từ một tập các giả thiết cho trước.

Lập luận của con người trong thực tế rất đa dạng, trong đó lập luận theo nghĩa chung (common sense reasoning), là cách lập luận thông dụng hàng ngày theo kinh nghiệm, dù là đơn giản, "hiển nhiên" với con người nhưng lại vô cùng khó khăn khi đưa lên máy tính. Sự phát triển của TTNT đưa một phần quan trọng vào các kết quả của logic toán học truyền thống, nhưng sự phát triển này cũng chính là động lực của rất nhiều hướng phát triển mới của logic toán học. Lập luận xấp xỉ, lập luận không đơn điệu, lập luận theo ngầm định, lập luận theo logic mờ, theo logic tình huống,... đang là những vấn đề nóng hổi của nghiên cứu TTNT, và chúng được một số người gọi chung là logic "không chuẩn" (non-standard logic). Ta nêu qua ở đây một vài ý về lập luận xấp xỉ và lập luận không đơn điệu.

Nguyên tắc suy diễn cơ bản nhất của các hệ lập luận là modus ponens: "Nếu $A \rightarrow B$ là một luật, A được khẳng định đúng thì suy ra B là một khẳng định đúng". Trong rất nhiều tình huống, nhất là trong vấn đề chẩn đoán, ta thường chỉ biết về A với một độ tin cậy (A) nào đấy ($0 = \mu(A) \leq 1$) và cũng chỉ được biết độ tin cậy

^(A→B) của luật A→B. Khi này ta có thể kết luận ra sao về độ tin cậy của B? Độ tin cậy của một kết luận sẽ ra sao nếu ta xác định nó sau khi áp dụng liên tiếp một dãy các luật xấp xỉ?

Lập luận không đón điệu hàm nghĩa khi thêm vào những khẳng định mới sẽ làm thay đổi những khẳng định đã có trong trí thức của ta. Ở trên ta đã có khẳng định "mỗi con chim đều biết bay", nhưng nếu bây giờ ta thêm vào trong hiểu biết của khẳng định "đà diều là một loài chim" thì chính trí thức về khả năng bay được của các loài chim của ta phải thay đổi lại.

Phải xây dựng những cơ sở toán học thế nào để có thể thực hiện được trên máy cái logic rất "thông thường" vừa kể trên ?

4. Ngôn ngữ và công cụ của trí tuệ nhân tạo. Trên phương diện lý thuyết, LISP và PROLOG được xem là các ngôn ngữ của TTNT. LISP (LISt Processing) ra đời năm 1958 tại MIT do John McCarthy phát triển và hiện đang được dùng phổ biến nhất ở Mỹ trong nghiên cứu TTNT. Đặc điểm chủ yếu của LISP là khả năng xử lý các danh sách và các hàm đệ quy để mô tả các quá trình và các bài toán. Hiện nay LISP là ngôn ngữ đang có rất nhiều phiên bản :LeLISP, LOGLISP, MACLISP,...

Ra đời năm 1973 tại đại học Marseille ở Pháp do A. Colmerauer, PROLOG đến đầu những năm 1980 trở nên nổi tiếng do sự lựa chọn của nước Nhật làm ngôn ngữ chính thức cho các máy tính thế hệ 5. PROLOG đang được sử dụng rộng rãi ở châu Âu và Nhật bản. PROLOG được xây dựng dựa trên logic tân từ bậc một và bản thân PROLOG chính là một hệ chứng minh định lý.

Trên thực tế, rất nhiều hệ TTNT được viết bằng các ngôn ngữ lập trình quen thuộc : pascal, cobol, fortran, C,... Cuối năm 1986 ở Mỹ có một cuộc tranh luận sôi nổi về việc chọn ngôn ngữ nào cho các hệ TTNT. Rất nhiều hãng lớn đã chuyển các hệ đã viết bằng prolog sang viết lại bằng C và chọn C làm ngôn ngữ xây dựng các hệ TTNT.

Dù sao ngôn ngữ cũng chỉ là công cụ để làm ra các sản phẩm. It nhiều phụ thuộc vào thói quen của người sử dụng. Đòi hỏi của thị trường và ứng dụng vẫn là yêu cầu cao nhất cho các công cụ và ngôn ngữ của TTNT.

Các ứng dụng của ttnt

Những ứng dụng của TTNT về thị giác máy nhằm góp phần làm ra các máy có khả năng hiểu được các hình ảnh, cảnh vật, dựa vào quá trình sử dụng trí thức để giải thích các dữ liệu của sensor. Những ứng dụng này liên quan rất nhiều đến nhận dạng và xử lý ảnh.

Nói đại thể, xử lý ngôn ngữ tự nhiên là làm cho máy hiểu được nghĩa các câu nói của con người bằng ngôn ngữ thông thường. Một khía cạnh quan trọng khác của xử lý ngôn ngữ tự nhiên là việc dịch tự động các văn bản từ tiếng một nước này sang tiếng một nước khác. Đương nhiên đây là các lĩnh vực cực kì khó khăn nhưng cũng có ý nghĩa vô cùng quan trọng trong sự phát triển của tin học trong tương lai.

Có rất nhiều vấn đề mặc dù không có chuyên gia giải quyết nhưng lại cần những chương trình để tìm ra giải pháp cho chúng, và một số trong chúng liên quan đến các kỹ thuật giải hơn là liên quan đến trí thức. Các kỹ thuật giải vấn đề (problem solving) liên quan đến việc xây dựng các thủ tục dùng các chiến lược điều khiển, phân ra vấn đề thành các vấn đề nhỏ hơn để giải quyết nhằm đạt được một đích cuối cùng nào đấy- được áp dụng rộng rãi trong rất nhiều hệ chứng minh.

Do khuôn khổ hạn chế của bài này, ta sẽ chỉ tập trung nói nhiều hơn trong phần còn lại về các hệ chuyên gia, là những ứng dụng đang được quan tâm nhiều nhất.

Các hệ chuyên gia

Khi ta hiểu Tin học là khoa học xử lý tự động thông tin bằng máy tính thì một dạng thông tin cực kì quan trọng và quý báu đến nay vẫn chưa được xử lý tốt trên máy là trí thức con người, đặc biệt là trí thức được các chuyên gia của từng lĩnh vực chất lọc, tích lũy qua nhiều tháng năm làm việc. Mỗi chuyên gia là một người có rất nhiều hiểu biết, kinh nghiệm và khả năng suy luận, vận dụng trí thức của mình để giải quyết các tình huống cụ thể với chất lượng cao mà người khác không làm được. Có thể nói nôm na các HCG là các hệ chứng minh máy tính thông minh nhằm bắt chước hai khả năng trên của người chuyên gia.

Một hệ chuyên gia như vậy cũng chủ yếu gồm hai thành phần chính: một cơ sở trí thức (Knowledge base) và một Mô-tơ suy diễn (Inference engine). Có thể nói việc xây dựng một HCG cần đến hầu hết các kết quả nghiên cứu cơ bản của TTNT mà ta vừa kể qua ở trên từ kỹ thuật biểu diễn trí thức đến kỹ thuật tìm kiếm heuristic, từ các phương pháp lập luận đến việc sử dụng một ngữ trình nào đấy của TTNT, kể cả việc giao tiếp với người sử dụng bằng một ngôn ngữ tự nhiên hay gần tự nhiên, khả năng nhận biết và phân biệt các hình ảnh,...

Tùy theo lĩnh vực ứng dụng, tùy theo kinh nghiệm và tùy theo khả năng của mô-tơ suy diễn có trong tay, trí thức sẽ được biểu diễn trong cơ sở dưới một dạng nào đấy. Cho đến hiện nay người ta mới chỉ xây dựng các hệ chuyên gia riêng biệt cho từng lĩnh vực cụ thể. Có lẽ chẳng điều ấy cũng phù hợp với con người vì ngày nay thật khó mà có được một người hiểu biết thấu đáo nhiều lĩnh vực, cả y học lẫn địa chất, xây dựng, luật pháp,...

Để đón giã, ta coi trí thức được biểu diễn bằng các luật dẫn. Cơ sở trí thức mô tả các đối tượng và quan hệ giữa chúng, mô tả các trường hợp đặc biệt và ngoại lệ, những cách nhìn khác nhau của người chuyên gia về

cùng một vấn đề, nhưng chiến lược giải quyết khác nhau và điều kiện áp dụng chúng ... Đặc biệt cơ sở tri thức có các luật heuristic là những trực cảm nghề nghiệp được người chuyên gia tích lũy qua tháng năm lao động và sáng tạo mà thông thường người ta không thể tìm thấy trong sách vở.

Ngoài những tri thức tương đối ổn định chứa trong cơ sở tri thức do người chuyên gia cung cấp trước cho hệ thống, trong quá trình làm việc hệ còn nhận được thông tin do người sử dụng cung cấp về từng đối tượng cụ thể cần xử lý, xem như dữ liệu của chương trình và được chứa trong một vùng bộ nhớ làm việc gọi là cơ sở sự kiện. Cơ sở sự kiện có tính chất tạm thời trong mỗi lần làm việc và chỉ tồn tại trong quá trình thực hiện chương trình. Nó còn chứa những thông tin, những kết luận hệ rút ra được trong lúc suy luận. Để làm đơn giản vấn đề, một số tác giả xem cơ sở sự kiện là một phần của cơ sở tri thức.

Cơ sở tri thức có thể dễ dàng được sửa đổi, thêm bớt, đổi thứ tự mà không hề ảnh hưởng gì đến chất lượng làm việc của hệ chuyên gia.

Mô-tơ suy diễn là chương trình tạo nên lập luận từ cơ sở tri thức. Trước hết nó phải có khả năng "hiểu" được tri thức chuyên gia và biến thành tri thức của mình. Trước một tình huống cụ thể, mà thông tin về tình huống này được chứa trong cơ sở sự kiện, mô-tơ thực hiện việc đối sánh một cách thông minh các thông tin này với cơ sở tri thức để tìm ra các tri thức liên quan, vận dụng các tri thức này, móc nối chúng lại, đặt ra các câu hỏi mới để lấy thêm thông tin, lập luận trên chúng để xây dựng nên các giải pháp. Cách đối sánh cụ thể của mỗi mô-tơ suy diễn được gọi là chiến lược điều khiển.

Mỗi mô-tơ suy diễn đều được lập tương đối với cơ sở tri thức theo nghĩa chính, nó có thể được nối với nhiều cơ sở tri thức để tạo nên những hệ chuyên gia khác nhau. Thật ra không thể có một mô-tơ vạn năng có thể làm việc tốt cho mọi kiểu ứng dụng. Vì như ta đã nói ở trên, có rất nhiều cách đối sánh cơ sở tri thức với cơ sở sự kiện xác định bằng thứ tự phát triển cây tìm kiếm, cũng như về đại thể có hai kiểu lập luận chính: lập luận với tri thức chính xác hoặc lập luận với tri thức và thông tin không chính xác, xấp xỉ... Như vậy chính xác hơn là mỗi mô-tơ suy diễn chỉ có thể hoạt động tốt cho một lớp các hệ chuyên gia nào đấy, thậm chí là trong từng ứng dụng cụ thể còn cần phải sửa đổi cho mô-tơ thích ứng hơn, chẳng hạn như để có "ngôn ngữ chuyên môn" của ngành nào đấy...

Hai nguyên tắc cơ bản của logic được các mô-tơ suy diễn sử dụng để tạo ra các kết luận là "modus ponens" và "modus tollens". Theo modus ponens, nếu trong cơ sở tri thức có $A \rightarrow B$ là một luật và trong cơ sở sự kiện có các khẳng định đúng về A, khi này kết luận B cũng đúng và được thêm vào trong cơ sở sự kiện. Theo modus tollens, nếu trong cơ sở tri thức có $A \rightarrow B$ là một luật và trong cơ sở sự kiện có khẳng định B sai, khi này A cũng sai và sự kiện này được bổ sung vào cơ sở sự kiện.

Nếu người sử dụng có một sự kiện nào đấy cần kiểm tra, mô-tơ suy diễn sẽ xét tập các luật chứa sự kiện này ở phần kết luận và gắng tìm cách chứng minh. Trong quá trình này một số sự kiện trở thành các đích phụ trợ cần phải kiểm tra. Cách làm này của mô-tơ được gọi là suy diễn lùi (backward chaining). Ngược lại là kiểu suy diễn tiến (forward chaining), tức là khi hệ không phải thực hiện một kiểm chứng đặc biệt nào và chỉ phải gắng tìm ra tất cả các kết luận có thể có.

Những hệ chuyên gia thực thụ thường phải được trang bị thêm một giao diện thân thiện với người sử dụng để họ có được cảm giác đang đối thoại với các chuyên gia "thật" một cách thông minh, có thể được thỏa mãn các câu hỏi "tại sao" và "thế nào". Mô-tơ suy diễn cùng với một giao diện thân thiện tạo nên một công cụ, thường gọi là cái "vỏ" của HCG (shell) hay một số người ở các nước đông Âu gọi là hệ chuyên gia rỗng, cho phép dễ dàng xây dựng nên các hệ chuyên gia bằng cách "nạp" tri thức cho nó.

Có thể kể ra một số ưu điểm cơ bản của hệ chuyên gia:

- * Trước hết các HCG có khả năng nhận nguồn tri thức và cách vận dụng tri thức của những người chuyên gia tài năng ra thành nhiều bản, cho nhiều người được hưởng năng lực này một cách dễ dàng và thuận lợi, và có thể dùng được ở bất cứ nơi xa xôi nào.

- * Chúng có thể làm việc thường xuyên, ngày đêm không biết mỏi mệt và nhầm lẫn, không bị các cảm xúc chi phối.

- * Chúng giúp cho các chuyên gia tránh được một số công việc nhàm chán lặp lại hàng ngày, để dành tâm trí cho những hoạt động sáng tạo khác.

- * Chúng có thể là công cụ dạy học rất tốt.

- * Để làm các hệ tri thức, người chuyên gia cần phải suy nghĩ, hệ thống hóa, tổ chức và sắp xếp lại các hiểu biết và kinh nghiệm của mình. Những tri thức này rất có ý nghĩa và người ta đã bắt đầu nói đến công nghệ tri thức, đến cách khai thác nguồn thông tin quý báu nhất này của con người.

Trong số những hệ chuyên gia đầu tiên người ta thường nói đến các hệ rất nổi tiếng sau đây:

Hệ MYCIN được E.H Shortliffe phát triển ở đại học Staford (1977) dùng để trợ giúp chẩn đoán và điều trị bệnh nhiễm khuẩn. Hệ MYCIN được xây dựng từ shell nổi tiếng EMYCIN.

Hệ DENDRAL được B.G. Buchanan và E.A Felgenbaum xây dựng năm 1978 (bằng LISP và FORTRAN) giúp xác định cấu trúc các hợp chất hoá học, những nguyên tử tham gia vào hợp chất và chúng được liên kết với nhau thế nào để tạo nên phân tử... xuất phát từ dữ liệu của các thiết bị đo phổ.

Hệ PROSPECTOR do P.E. Hart và R.O. Duda xây dựng ở Viện nghiên cứu Stanford có khả năng phát hiện các mỏ quặng mà ngay một số nhà chuyên môn cũng không tìm ra.

Ngày nay các hệ chuyên gia được xây dựng và thử nghiệm rất nhiều ở các nước có tin học phát triển. Những người nghiên cứu, thiết kế và xây dựng HCG, những người quan tâm sử dụng và thương mại HCG hàng năm gặp nhau tại Hội nghị quốc tế về "Hệ chuyên gia và Ứng dụng" tại Avignon nước Pháp. Tại đây người ta giới thiệu các kết quả nghiên cứu mới và các ứng dụng phong phú trên khắp thế giới về HCG...

Có thể điểm qua một vài khuynh hướng mới cho các HCG:

* Các HCG sẽ chỉ thực sự phát huy được hết sức mạnh nếu chúng được ghép vào các hệ tin học truyền thống. Chúng không chỉ đưa ra các lời khuyên, gợi ý các quyết định, các giải pháp, mà còn thực hiện các thủ tục tính toán sau quá trình suy luận. Như vậy chúng phải có khả năng nối với các Cơ sở dữ liệu, với các hệ phân tích dữ liệu, các hệ thống điều khiển kỹ thuật ...

* Chúng phải được xây dựng trong một môi trường tin học thích ứng được với tiến bộ mới nhất, như có các giao diện thân thiện người-máy, như các chế độ của số, con chuột(click and point), các chỉ thức biểu thị bằng hình ảnh, các hệ chuyên gia dựa trên cấu trúc tính toán song song,... Một hệ chuyên gia tốt không phải chỉ có khả năng suy luận mà còn phải có khả năng trao đổi thông tin dễ dàng, sinh động với người dùng. Đây cũng là một nguyên nhân để nhiều nhà chế tạo chuyển sang xây dựng hệ của mình bằng C là ngôn ngữ cho phép dễ dàng bổ sung một số hạn chế của Prolog và Lisp về các phương diện này.

* Các hệ lai (hybrid shell) cho phép đồng thời biểu diễn nhiều kiểu tri thức có bản chất đa dạng. Cần phải có các hệ giúp thu nhận tri thức (ở mức đơn giản là các chương trình soạn thảo tri thức) và các hệ giúp tổ chức, đánh giá chất lượng tri thức (tính phi mâu thuẫn của một hệ tri thức chẳng hạn). Ở mức cao hơn là các hệ học giúp tự động tạo sinh tri thức cho các HCG.

* Các mô-tơ suy diễn thực hiện được nhiều chiến lược lập luận khác nhau, mà một phần rất quan trọng dựa trên các kết quả lý thuyết về logic không chuẩn, nhất là về lập luận xấp xỉ. Khả năng quản lý những cơ sở tri thức rất lớn và hoạt động được trong thời gian thực cũng là một yêu cầu hết sức quan trọng.

Lúc đặc đã có tuyên bố về các HCG thế hệ thứ hai, nhất là các công cụ để phát triển các HCG. Chủ yếu đó là các hệ được thiết kế theo các khuynh hướng vừa kể trên, định hướng cho người sử dụng, hoàn thiện hơn, vượt ra ngoài khuôn khổ các phòng thí nghiệm và được bán trên thị trường TTNT. Nói chung đây là các hệ xây dựng trên máy tính lớn, có tính phổ dụng cao, chấp nhận các cơ sở tri thức lớn và không cố định trước cách thức biểu diễn tri thức. Những hệ nổi tiếng nhất theo xu hướng này là ART(Automated Reasoning Tool) của Inference Corp., S1 của Teknowledge Inc.,CRL của Carnegie Group Inc.,NEXPERT của Digital Equipment Corp., và KEE (Knowledge Engineering Environment) của Intellicorp trong đó hai hệ sau cùng, cho đến lúc này đang bán được nhiều nhất trên thị trường và được sử dụng rộng rãi nhất. Dù sao cũng hãy còn sớm để nói về một thế hệ mới các HCG khi thế hệ đầu đang trong giai đoạn phát triển và tự khẳng định mình.

Chất lượng của các HCG sẽ phụ thuộc chủ yếu vào chất lượng của cơ sở tri thức. Và vấn đề làm sao có được tri thức đem sử dụng đang ngày càng trở nên hết sức quan trọng, liên quan chặt chẽ với các kỹ thuật thu nhận tri thức (knowledge acquisition) và học tự động (learning). Nhớ lại mô hình của Nilson, nếu như ngày nay các kỹ thuật tìm kiếm heuristic đã tương đối hoàn thiện, thì có thể thay phần này của mô hình bằng vấn đề học và các mô hình quá trình nhận thức của con người (perception). Xuất phát từ các thí dụ để tạo nên tự động các cơ sở tri thức đang là bài toán quan trọng và thú vị của TTNT hôm nay và tương lai.

TÀI LIỆU THAM KHẢO

Nhận ngày 10 - 10 - 1988

1. W.B. GEVARTER, Artificial Intelligence, Expert System, Computer Vision and Natural Language Processing, Noyes Publication, 1984.
2. P. JACSON, Introduction to Expert System, Addison-Wesley Publishing Company, 1986.
3. A. Barr. E.A. Feigenbaum, Handbook of Artificial Intelligence, Los, Calif., 1981.
4. Proceedings of International Conferences on Expert Systems and their Applications, Avignon 1985, 1986, 1987, 1988.

ABSTRACT

AN OVERVIEW ON ARTIFICIAL INTELLIGENCE AND EXPERT SYSTEMS

This paper presents essential concepts of Artificial Intelligence and Expert systems as well as its development history, actual situation and perspective.