

HỆ TRỢ GIÚP KÊ VÀ KIỂM TRA ĐƠN THUỐC CHỮA BỆNH TĂNG HUYẾT ÁP ES - TENSION

NGUYỄN THANH THÚY

Abstract. In this paper, we investigate some features of the ES-Tension system for supporting prescription and verification of therapy recommendation for the hypertensive disease. The system is implemented in Borland C, version 3.1 with the Knowledge base of 234 rules. These rules are divided into 4 subclasses. In addition to the reasoning capacity, ES-Tension can explain and judge why and how questions of non-professional users.

MỞ ĐẦU

Hệ chuyên gia đầu tiên DENDRAL xuất hiện vào giữa những năm 70, nhưng phải đến khi xuất hiện và thử nghiệm hệ chuyên gia MYCIN trong lĩnh vực chuẩn đoán và điều trị bệnh nhiễm trùng máu, kỹ thuật hệ chuyên gia mới trở thành mối quan tâm đối với các chuyên gia tin học. Về thực chất có thể “giải phẫu” các hệ chuyên gia ứng dụng theo hai thành phần chức năng cơ bản như sau:

Hệ chuyên gia = Cơ sở tri thức + Mô tơ suy diễn

tuy rằng trong đó còn có những môđun khác như giao diện người máy và môđun giải thích.

Các hệ chuyên gia chia làm hai loại: Hệ chuyên gia cỡ nhỏ với cơ sở tri thức (CSTT) khoảng vài trăm luật và hệ chuyên gia cỡ lớn với CSTT khoảng vài nghìn luật. Với các hệ chuyên gia cỡ lớn nhằm giải quyết các bài toán phức tạp, điều quan trọng là xác định mẫu hình bài toán, trên cơ sở đó có thể chọn lựa công cụ tạo lập hệ chuyên gia (CCTLHCG) phù hợp. Do yêu cầu thực tế và hạn chế kinh phí, trong hoàn cảnh hiện tại ở Việt Nam có được CCTLHCG ở dạng sản phẩm thương mại không phải là điều dễ dàng. Vấn đề đặt ra là xây dựng CCTLHCG cỡ nhỏ và các hệ chuyên gia “vỏ” có thể thích hợp với các hệ thống ứng dụng khác.

Ngay từ cuối những năm 80, ở Viện Công nghệ thông tin đã có những thử nghiệm bước đầu xây dựng các hệ trợ giúp khám chữa bệnh nội khoa, châm cứu và chẩn trị đông y. Một đề tài nghiên cứu có qui mô và đã được tiến hành nhằm tạo ra một bộ sinh hay các hệ chuyên gia “vỏ” OREST, từ đó xây dựng các hệ chuyên gia khác nhau, tùy thuộc vào CSTT được tích hợp. Hệ ES-Tension là

một thử nghiệm nhằm áp dụng kỹ thuật hệ chuyên gia trong trợ giúp kê và kiểm tra đơn thuốc chữa bệnh tăng huyết áp.

Nhiệm vụ của hệ ES - Tension là: Khi người dùng đưa vào một vài thông tin về bệnh nhân và

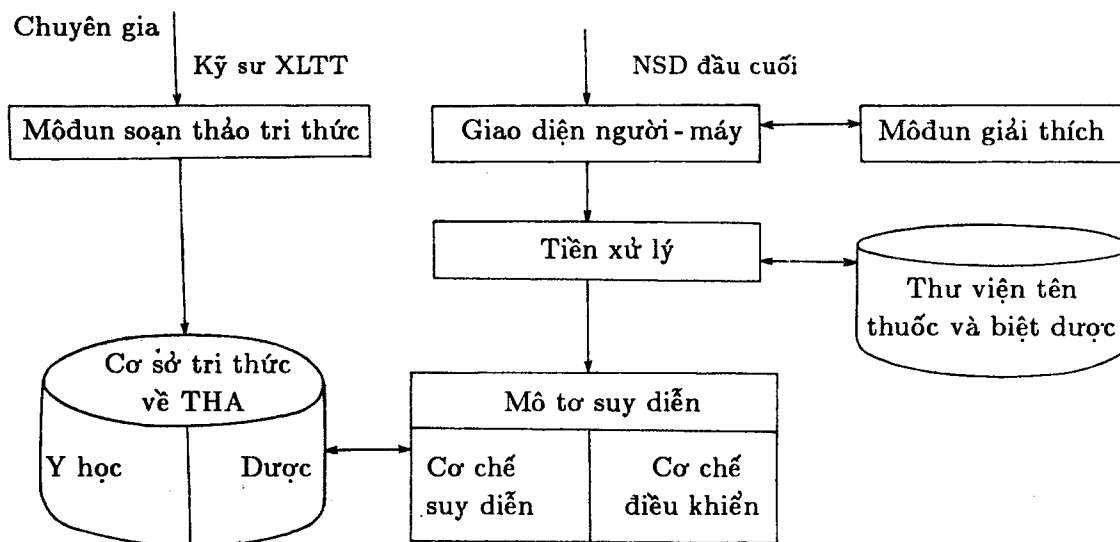
- Nếu có kèm theo đơn thuốc do bác sĩ kê, hệ sẽ kiểm tra đơn thuốc xem có hợp lệ không. Nếu không hợp lệ thì giải thích cho người dùng lý do tại sao và đưa ra đơn thuốc phù hợp.

- Nếu người dùng không đưa đơn thuốc vào thì hệ sẽ tự kê đơn.

- Tại bất kỳ thời điểm nào trong quá trình trợ giúp, hệ luôn có thể giải đáp các thắc mắc của người dùng dưới dạng các câu hỏi Why? How?

2. CẤU TRÚC CỦA HỆ ES-TENSION

Cũng giống như cấu trúc một hệ chuyên gia bất kỳ, Hệ ES - Tension bao gồm các mô đun cơ bản sau:



Mô đun giao diện người-máy có nhiệm vụ lấy thông tin từ người dùng và đưa ra những trợ giúp hay giải thích trong trường hợp cần thiết. Để tiện lợi cho người sử dụng, các thông báo được thể hiện bằng ngôn ngữ tự nhiên và đối với mỗi sự kiện hệ gắn kèm với một câu hỏi để yêu cầu người sử dụng đưa thông tin vào.

Mô tơ suy diễn chịu trách nhiệm chức năng “suy nghĩ” trong hệ thống. Hệ sử dụng cơ chế suy diễn tiến nhằm đưa ra kết luận từ những thông tin về các thông

số / sự kiện ban đầu hoặc suy diễn lùi nhằm thử khẳng định một kết luận nào đó trên cơ sở áp dụng một cách quay lui các phương án / luật suy diễn cho đến khi mọi tiên đề đều thuộc vào tập giả thiết đã cho. Môđun giải thích đảm nhiệm chức năng lý giải các câu hỏi của người sử dụng:

- 1 - Tại sao phải cần đưa vào thông tin về một sự kiện, thuộc tính nào đó?
- 2 - Tại sao có được luật suy diễn f ?
- 3 - Tại sao đưa ra kết luận về luật suy diễn f ?
- 4 - Tại sao không suy diễn ra được kết luận về sự kiện f ?

Các giải thích có thể theo hai phương án:

- hoặc trả lời và giải thích theo kiểu “chọn gói”,
- hoặc trả lời theo từng bước.

Cần lưu ý là các câu hỏi 1, 2, 3, 4 có thể trộn lẫn vào nhau.

Môđun soạn thảo tri thức cho phép chuyên gia hoặc kỹ sư xử lý tri thức soạn thảo các luật suy diễn. Ngoài chức năng kiểm tra cú pháp của luật đầu vào, môđun này còn chịu trách nhiệm biến đổi các luật từ biểu diễn ngoài vào biểu diễn trong.

Môđun tiền xử lý có nhiệm vụ thuần nhất hóa dữ liệu đầu vào trong đơn thuộc, trên cơ sở tra cứu thư viện các tên thuốc đã biết được.

Cơ sở tri thức lưu trữ các tri thức chuyên gia trong lĩnh vực điều trị bệnh tăng huyết áp, cả về y học và dược học. Thư viện tên thuốc và biệt dược được phục vụ cho môđun tiền xử lý, trong đó chứa các tên chuyên môn chung và các biệt dược của thuốc.

3. CƠ SỞ TRI THỨC VÀ TỪ ĐIỂN BIỆT DƯỢC TRONG HỆ ES-TENSION

3.1. Cơ sở tri thức

Hệ ES-Tension sử dụng phương pháp biểu diễn tri thức chuyên gia dưới dạng luật sản xuất

$$rr : p_1 \wedge \dots \wedge p_n \Rightarrow q$$

Với ngữ nghĩa là

Nếu	tồn tại/có/biết	p_1
Và	tồn tại/có/biết	p_2
.....
Và	tồn tại/có/biết	p_n
Thì	tồn tại/có/biết	q

ở đây mỗi p_i và q là các sự kiện, có dạng:

- hoặc tên một biến mệnh đề logic,
- hoặc một biểu thức quan hệ giữa các đại lượng, biểu thức số, string..

Ví dụ: 1. Nếu THA kịch phát : là đúng
 Và HATĐ \geq 200 : num
 Và HATĐ \leq 260 : num
 Và HATT \geq 120 : num
 Và HATT \leq 140 : num
 Thì Thuốc lợi tiểu - an thần : là đúng

2. Nếu giới tính = "Nữ" : string
 Và có thai : là đúng
 Thì dùng thuốc UCCD Angiotensin : là sai

Các luật được chia thành 4 lớp con:

a. Lớp các luật xác định loại thuốc phải dùng và không được dùng do đặc điểm của bệnh quy định. Lớp luật này xác định nhóm thuốc hoặc có thể tên thuốc cụ thể tùy theo các thông số về bệnh nhân theo giới tính, độ tuổi... và điều kiện thuốc men hiện có.

Luật 12 Nếu bệnh nhân bị THA thường xuyên : là đúng
 Và chống chỉ định chẹn giao cảm β : là đúng
 Và chưa được điều trị lần nào : là đúng
 Thì dùng thuốc lợi niệu : là đúng

b. Lớp các luật xác định thuốc cần phải dùng trong nhóm thuốc đã chọn.

Lớp các luật này bao gồm các luật xác định các tên thuốc cụ thể trong nhóm thuốc: đường dùng: uống (viên, nước...), tiêm (ven, bắp...), liều dùng, thời hạn....

Luật 77 Nếu dùng thuốc chẹn giao cảm β : là đúng
 Và có Acebutolol : là đúng
 Và có Propranolol : là đúng
 Thì dùng Acebutolol : là đúng

Trong nhóm này cũng có các luật, xác định nhóm thuốc đặc trưng trong khi phải phối hợp các nhóm thuốc khác nhau:

Luật 94 Nếu dùng phối hợp lợi tiểu mạnh và an thần : là đúng
 Và có thuốc lợi tiểu mạnh : là đúng
 Và có thuốc an thần : là đúng
 Thì dùng thuốc lợi tiểu mạnh : là đúng

c. Lớp các luật xác định đường dùng là liều dùng của thuốc

Luật 183 Nếu uống hydrochlorothiazide : là đúng
 và có Catapressan : là đúng
 Thì dùng Catapressan : là sai

3.2. Từ điển tên thuốc và biệt dược

Hệ ES - Tension lưu một từ điển tương đối đầy đủ các tên thuốc và biệt dược trong phạm vi điều trị bệnh tăng huyết áp. Các thuật ngữ được trích từ "Từ điển bách khoa bệnh học", NXB y học, 1994 và các sổ tay tra cứu thuốc chuyên khoa.

4. MÔ TƠ SUY DIỄN

So với suy diễn lùi, cài đặt suy diễn tiến đơn giản hơn. Song quá trình suy diễn tiến phải huy động mọi luật có thể áp dụng được, mà không lưu ý đến liệu luật có liên quan đến kết luận mong muốn hay không. Do vậy, đối với các CSTT lớn với số luật ngày càng tăng, thuật giải suy diễn này dẫn tới bùng nổ tổ hợp. Ngược lại, suy diễn lùi phức tạp hơn, nhưng có ưu điểm là chỉ chọn những luật hướng tới đích đặt ra. Về thực chất cơ chế suy diễn lùi được cài đặt ở đây tương ứng với tìm kiếm sâu trên đồ thị Và/Hoặc biểu diễn tập luật [6].

Ví dụ: Xét tập luật $r_1 : a \wedge b \Rightarrow c$ $r_2 : a \wedge c \Rightarrow d$
 $r_3 : d \wedge k \Rightarrow e$ $r_4 : a \wedge d \Rightarrow e$

Tập sự kiện giả thiết $GT = \{a, b\}$, tập sự kiện kết luận $KL = \{e\}$.
 Cần phải chứng minh $GT \Rightarrow^* KL$.

4.1. Thuật giải suy diễn tiến

Trong thuật giải sau $R = \{r_1, \dots, r_m\}$, r_i là các luật sản xuất

TGian là các tập sự kiện đúng

Vet là tập các luật sản xuất đã sử dụng

Lọc (F, Rule) là thủ tục cho tập các luật $r \in \text{Rule}$, $r : \text{left} \rightarrow q$ sao cho $\text{left} \subseteq F$.

Phương pháp $\{_1$ TGian = GT , Vet = 0, Thoa = Lọc (TGian, R);
 While Thoa \neq 0 and $KL \not\subset$ TGian do $\{_2$
 $r \leftarrow$ get (Thoa) /* $r : \text{left} \rightarrow q^*$ /*
 Vet = vet \cup $\{r\}$; $R = R \setminus \{r\}$;
 TGian = TGian \cup $\{q\}$; Thoa = Lọc (TGian, R) $\}_2$
 if $KL \subset$ TGian then exit ("Thành công")
 eles exit ("Không thành công") $\}_1$

Ví dụ: Với tập luật trên, ta có kết quả Vet = (r_1, r_2, r_3) .

Nhận xét: Tùy thuộc vào cơ chế vào, ra của tập Thoa trong thuật giải dưới dạng Queue (vào trước, ra trước) hay Stack (vào sau, ra trước) ta sẽ có kết quả tập Vet khác nhau.

4.2. Thuật giải suy diễn lùi

Trong thuật toán này:

- Goal là tập sự kiện cần phải chứng minh (tổ chức dưới dạng STACK)

- Vet là tập luật đã sử dụng (tổ chức dưới dạng STACK)
- Tìm luật (f, j, Rule, i) là thủ tục xác định $r_j, m \geq j \geq i + 1$ sao cho r_j có dạng $r_j : \text{left} \Rightarrow f$. Nếu không tìm thấy thì $j = m + 1, m$ là số luật trong Rule.

```

Phương pháp {1 if  $KL \subseteq GT$  then exit ("Thành công")
              else {2 Goal =  $KT \setminus GT$ ; Vet = 0; back = false;
                    do f ← get (goal);
                    repeat {3 if  $f \notin GT$  then {4 timluat ( $f, j, r, o$ ) /* Tìm luật thứ  $j$ 
                               sao cho  $r_j : \text{left}_j \Rightarrow f^*$ */;
                               if  $j \leq m$  then {5 Vet =  $\text{Vet} \cup \{(f, j)\}$ 
                                       goal =  $\text{goal} \cup \text{left}_j \setminus GT$  }5
                               else /*, else quay lui theo các luật */ {6 back = true;
                                       while  $f \notin KL$  and back do {7
                                               repeat {8 ( $g, k$ ) ← get (vet); goal =  $\text{goal} \setminus \text{left}_k$  }8
                                               until  $f \in \text{left}_k$ ;
                                               Timluat ( $g, l, r, k$ ) /* tìm phương án khác đối với  $g^*$  */
                                               if  $1 \leq m$  then {9 goal =  $\text{goal} \setminus \text{left}_k$ ;
                                                       goal =  $\text{goal} \cup \text{left}_l \setminus GT$ ;
                                                       Vet =  $\text{Vet} \cup \{(g, l)\}$ ;
                                                       back = false }9
                                               else  $f = g$  }7 }6 }4
                               if goal = 0 then BREAK else f ← get (goal) }3
                    until  $f \in KL$ ;
                    if  $f \in KL$  and back then exit ("không thành công")
                    else exit (Vet) }2 }1

```

5. CÀI ĐẶT HỆ ES-TENSION

Hệ ES-Tension được cài đặt bằng ngôn ngữ lập trình Borland C, version 3.1. Các modul lớn trong hệ là:

5.1. Nạp dữ liệu

Phần này có nhiệm vụ tải tri thức (bộ luật), tập sự kiện giả thiết, kết luận từ bộ nhớ ngoài vào bộ nhớ trong.

i) Bảng sự kiện:

```

struct FACT { char name [MAXLEN], position, type, value - pregiven;
              union UNIVERSAL value;
              struct RULE      *op, *head, *tail;
              struct FACTt     *next }

```

ii) Bảng luật

```
struct RULE { char name [MAXLEN], role, sens, sign;
              struct FACT *fact, *prec, *nect;
              struct RULE *same }
```

Chức năng nạp CSTT được đảm nhận bởi hàm

```
int GetRuleBase (char *filenameR, struct RULE **head - rule, **head - op,
                **head - fact)
```

iii) Cấu trúc một phần tử trong bảng sự kiện giả thiết và kết luận được cho bởi

```
struct PARA - PC { char name [MALEN], type;
                  union UNIVERSAL vule;
                  int error }
union UNIVERSAL { char bool, string [20];
                  float num }
```

Nạp giả thiết và kết luận được cho bởi:

```
int GetFactBase (char *filename PC, struct PREC - CONC
                **head - prec, **head - conc)
```

5.2. Mô tả suy diễn lùi

Trong mô tả suy diễn lùi ta sử dụng hai cấu trúc STACK là Vet và Goal:

```
struct Vet { struct RULE *clause;
             int order;
             struct vet *next }
struct Goal { struct PREC-CONC *conc;
              struct vet *tro - Vet
              struct goal *next }
```

Sau đây là cấu trúc dùng để lưu luật tìm được trong thủ tục Tìm-luật

```
struct FOUND - RULE { int order;
                      struct RULE *conc1, *conc2
                      struct PREC - CONC *head }
```

Hàm suy diễn chính được cho bởi:

```
int Backward (struct FACT **fact - table, struct RULE **rule - table,
              op - table, struct PREC - CONC *conc - var),
```

Hàm này thực hiện suy diễn lùi và cho kết quả:

```
1 nếu thành công
-1 nếu ngược lại
```

5.3. Cài đặt cơ chế hỏi đáp và giải thích trong hệ ES - Tension

Đối với các biến/sự kiện, thay vì đưa vào các giá trị cụ thể, người sử dụng có thể dùng các lựa chọn “không biết” và “tại sao”. Ngoài ra, hệ còn cho phép giải đáp các câu hỏi khác:

- Tại sao đưa ra được luật r
- Tại sao đưa ra được sự kiện f

Hơn nữa các câu hỏi dạng trên có thể lồng lại với nhau. Tương ứng với 3 chức năng này là các hàm.

- int Askforfact (struct FACT *fact - table, struct GOAL goal - conc);
- int Explain for Goal (struct FACT *fact - table struct GOAL goal);
- int Explain for Vet (struct Vet **heat - Vet).

6. THỬ NGHIỆM HỆ ES - TENSION VÀ HƯỚNG PHÁT TRIỂN

Hệ ES - Tension đã được thử nghiệm với:

- CSTT đầy đủ gồm 234 luật, tạm chia thành 4 lớp, Các luật được soạn thảo có sử dụng font chữ Việt nhằm tăng tính thân thiện với sử dụng không chuyên. Các điều kiện kết luận tham gia trong các luật có thể:

- Sự kiện logic
- Biểu thức quan hệ < biến> <phép toán quan hệ> <giá trị>
hoặc <biến> <phép toán quan hệ> <biến> ,

các đại lượng có thể là số hoặc string.

- Cơ chế suy diễn tiến và lùi, đã chạy thử nghiệm cho kết quả tốt.
- Phần giải thích đáp đúng được nhu cầu của người dùng.

Các hướng phát triển tiếp của hệ:

- Hoàn thiện CSTT, kiểm tra và phát hiện mâu thuẫn.
- Thiết kế và xây dựng các hệ chuyên gia khác trợ giúp và kiểm tra đơn thuốc dùng cho các bệnh có liên quan với nhau.
- Tích hợp các hệ HCGG trong một hệ cỡ lớn, dùng cho phạm vi tương đối rộng các loại bệnh.

TÀI LIỆU THAM KHẢO

1. D. A. waterman, *A guide to expert systems*, Addison - Wesley Publishing company, 1989.
2. B. Buchman, E. Shortliffe, *Rule - based expert systems: The MYCIN experiments of the stanford heuristic programming project*, Addison - Wesley Pub. company, 1984.

3. R. Levine, D. Drang, B. Edelson, *A comprehensive guide to AI and ES*, McGraw-Hill Book company, 1986.
4. M. Gondran, *Introduction aux systemes experts*, Eyrolles, Paris, 1986.
5. C. Kulicowski, S. Weiss, *A practical guide to designing expert system*, Rowman and Allanheld Publishers, 1984.
6. Nguyễn Thanh Thủy, *Trí tuệ nhân tạo: Các phương pháp giải quyết vấn đề và xử lý tri thức*, NXB Giáo dục, 1995.
7. Từ điển Bách khoa bệnh học. NXB Y học, 1994.

*Khoa Công nghệ thông tin
Trường Đại học Bách khoa Hà Nội.*

Nhận bài ngày 12-4-1995