THUẬT TOÁN ILA

1 Các dữ kiện ban đầu

a. Các mẫu chính là các dòng trong bảng tương ứng. Mỗi cột của bảng chính là các thuộc tính của mẫu.

b. Một tập huấn luyện m mẫu, mỗi mẫu gồm k thuộc tính và thuộc tính lớp với n quyết định có thể có.

c. Một tập hợp luật R với khởi đầu R bằng rỗng.

d. Khởi đầu tất cả các dòng trong bảng đều là không khóa (unmarked).

2 Thuật toán:

Bước 1:

Chia bảng có chứa m mẫu thành n bảng con. Một bảng ứng với một giá trị có thể có của thuộc tính lớp. (Từ bước 2 đến bước 8 sẽ được lặp lại cho mỗi bảng)

Bước 2:

Khởi tạo số lượng thuộc tính kết hợp j với j = 1.

Bước 3:

Với mỗi bảng con đang xét, phân chia các thuộc tính của nó thành một danh sách các thuộc tính kết hợp, mỗi thành phần của danh sách có j thuộc tính phân biệt.

Bước 4:

Với mỗi kết hợp các thuộc tính trong danh sách trên, đếm số lần xuất hiện các giá trị cho các thuộc tính trong kết hợp đó ở các dòng chưa bị khóa của bảng đang xét nhưng nó không được xuất hiện cùng giá trị ở những bảng con khác. Chọn ra một kết hợp trong danh sách sao cho nó có giá trị tương ứng xuất hiện nhiều nhất và được gọi là Max_combination.

Bước 5:

If max_combination = 0 thì j = j + 1 quay lại bước 3.

Bước 6:

Khóa các dòng ở bảng con đang xét mà tại đó nó có giá trị bằng với giá trị tạo ra max_combination.

Bước 7:

Thêm vào R luật mới với giả thiết là max_combination các thuộc tính và giá trị tương ứng phân biệt và kết nối các bộ này bằng AND, kết luận của luật là giá trị của thuộc tính quyết định tương ứng với bảng con này.

Bước 8:

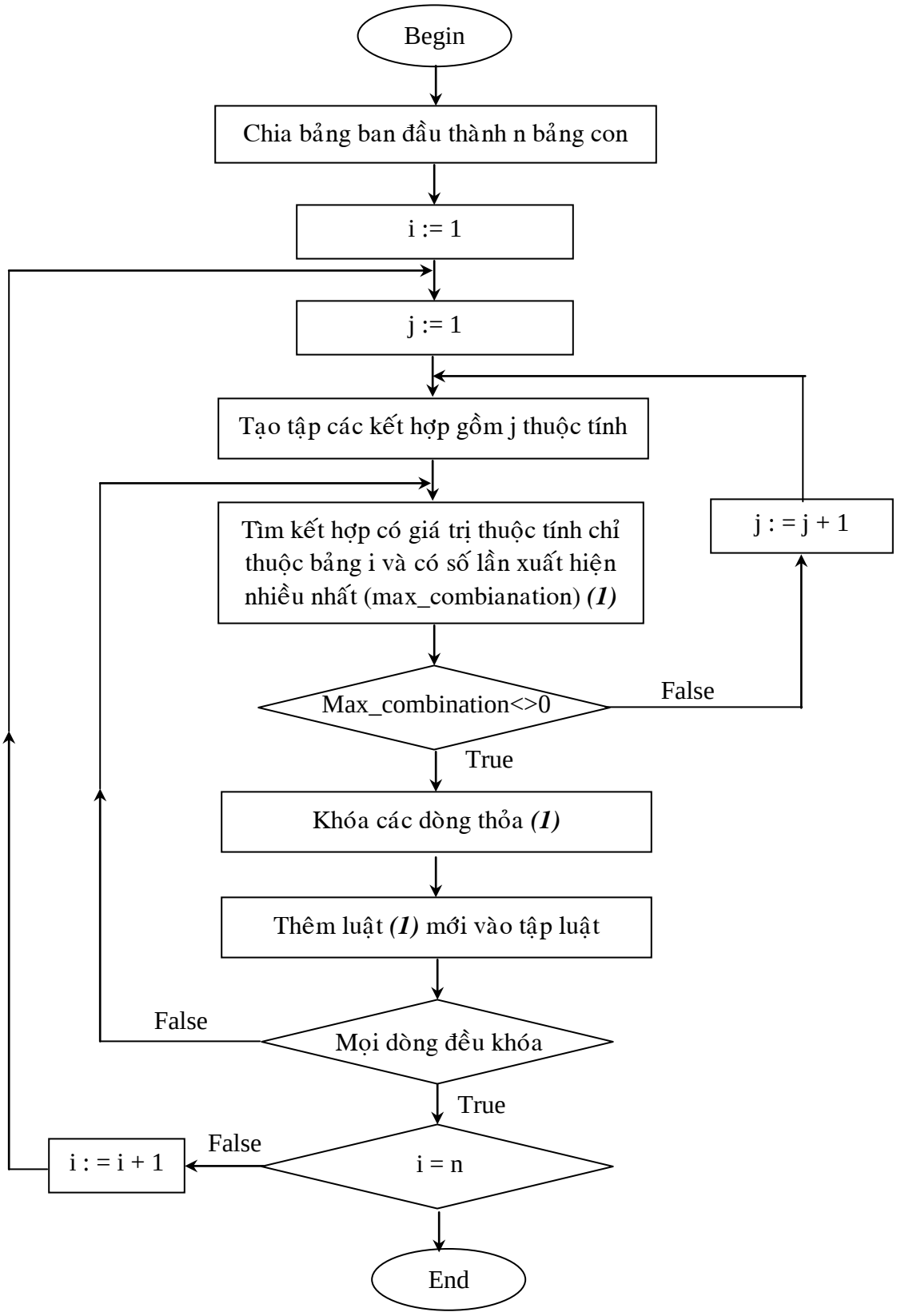
Nếu tất cả các dòng đều khóa

Nếu còn bảng con thì qua bảng con tiếp theo và quay lại bước 2.

Ngược lại chấm dứt thuật toán

Ngược lại (nghĩa là vẫn còn dòng chưa khóa trong bảng con đang xét) thì quay lại bước 4.

3 Lưu đồ


Begin
Chia bảng ban đầu thành n bảng con
i := 1
j := 1
Tạo tập các kết hợp gồm j thuộc tính
Tìm kết hợp có giá trị thuộc tính chỉ thuộc bảng i và có số lần xuất hiện nhiều nhất (max_combianation) (1)
j : = j + 1
Max_combination<>0
False
True
Khóa các dòng thỏa (1)
Thêm luật (1) mới vào tập luật
False
Mọi dòng đều khóa
True
False
i : = i + 1
i = n
End

4. Ví dụ minh họa

ILA là một thuật toán đơn giản để rút ra các luật từ một tập mẫu. Một mẫu bao gồm các phần với các thuộc tính cố định, với các giá trị có thể của nó. Để mô tả ILA chúng ta sẽ sử dụng 3 tập mẫu khác nhau bao gồm đối tượng, thời tiết, và mùa.

Trước tiên ta xem xét với mẫu đối tượng như trong bảng 1.

Bảng này bao gồm:

+ 7 mẫu (m = 7)

+ 3 thuộc tính (k = 3) : [size], [color], [shape]

+ Thuộc tính quyết định ([decision]) có 2 giá trị có thể là {“Yes”, “No”}(n = 2).

Example No.	Size	Color	Shape	Decision
1	Vừa	Xanh dương	Hộp	Yes
2	Nhỏ	Đỏ	Nón	No
3	Nhỏ	Đỏ	Cầu	Yes
4	Lớn	Đỏ	Nón	No
5	Lớn	Xanh lá cây	Trụ	Yes
6	Lớn	Đỏ	Trụ	No
7	Lớn	Xanh lá cây	Cầu	Yes

Bảng **1: Tập huấn luyện các đối tượng phân lớp**

Trong đó:

- Thuộc tính [size] có các giá trị {“nhỏ”, “vừa”, “lớn”}

- Thuộc tính [color] có các giá trị {“đỏ”, “xanh lá cây”, “xanh dương”}

- Thuộc tính [shape] sẽ có các giá trị tương ứng có thể là {“hộp”, “nón”, “cầu”, “trụ”}.

Bước 1

Với n = 2 đầu tiên thuật toán sẽ tạo thành 2 bảng con như trong Bảng 2.

Example No		Size	Color	Shape	Decision
Old	New				
Bảng_con_1					
1	1	Vừa	Xanh dương	Hộp	Yes
3	2	Nhỏ	Đỏ	Cầu	Yes
5	3	Lớn	Xanh lá cây	Trụ	Yes
7	4	Lớn	Xanh lá cây	Cầu	Yes
Bảng_con_2					
2	1	Nhỏ	Đỏ	Nón	No

4	2	Lớn	Đỏ	Nón	No
6	3	Lớn	Đỏ	Trụ	No

Bảng 2: Các bảng con của tập huấn luyện được chia thông qua lớp quyết định

Bước 2

Với bảng con đầu tiên (Bảng_con_1) trong Bảng 2, với j = 1.

Bước 3

Danh sách các thuộc tính kết hợp có thể có bao gồm {[size]}, {[color]}, {[shape]}.

Bước 4

- Với thuộc tính [size] giá trị thuộc tính "vừa" xuất hiện trong Bảng_con_1 nhưng không xuất hiện trong Bảng_con_2 , do đó giá trị max_combination sẽ là "vừa".

Ngoài ra giá trị các giá trị "nhỏ" và "lớn" xuất hiện trong cả 2 Bảng_con_1 và Bảng_con_2 nên không được xét ở bước này.

- Tiếp theo với thuộc tính [color] giá trị "xanh lá cây" sẽ được chọn thay vì là "xanh dương" do "xanh lá cây" xuất hiện 2 lần trong khi "xanh dương" xuất hiện 1 lần như vậy max_combination sẽ là "xanh lá cây".

- Tiếp theo với thuộc tính [shape] ta sẽ chọn giá trị "cầu" vì xuất hiện 2 lần trong Bảng_con_1, như vậy cả 2 "xanh lá cây" và "cầu" đều có cùng số lần xuất hiện. Theo thuật toán thì sẽ chọn thuộc tính đầu tiên có nghĩa là "xanh lá cây" cho max_combination.

Bước 5

Do max_combination<>0 nên làm tiếp bước 6

Bước 6

Dòng 3 và 4 trong Bảng_con_1 sẽ được khóa với giá trị max_combination là "xanh lá cây".

Bước 7

Khi đó luật được tạo ra là:

Luật 1: **if color là "xanh lá cây" then quyết định là YES.**

Bước 8

Bảng_con_1 chưa được khóa hết nên quay lại bước 4

ILA sẽ lặp lại từ bước 4 đến bước 8 ở các mẫu còn lại của Bảng_con_1 (nghĩa là dòng 1 và dòng 2). Bằng cách áp dụng lại các bước này, chúng ta sẽ có giá trị của thuộc tính [size] là "vừa" và "xanh dương" của [color] và giá trị "hộp" và

“cầu” của [shape] là max_combination, áp dụng thuật toán ta sẽ lấy giá trị đầu tiên (nghĩa là “vừa” của [size]) do đó luật 2 sẽ được thêm vào tập luật như sau:

Luật 2: **if size là “vừa” then quyết định là YES.**

Như vậy dòng đầu tiên trong Bảng_con_1 sẽ bị khóa và từ bước 4 đến bước 8 lại được áp dụng cho dòng còn lại trong bảng (nghĩa là dòng 2). Ở đây giá trị “cầu” của thuộc tính [shape] sẽ được chọn khi đó ta có luật:

Luật 3: **if shape là “cầu” then quyết định là YES.**

Khóa dòng thứ 2 còn lại của Bảng_con_1 do đó tất cả các dòng đều được khóa.

Chúng ta sẽ tiếp tục thực hiện trên Bảng_con_2. Giá trị “nón” của thuộc tính [shape] xuất hiện 2 lần trong dòng đầu tiên và dòng thứ 2 của Bảng_con_2, do đó 2 dòng này sẽ được khóa khi đó luật 4 được đưa vào tập luật với luật 4 như sau:

Luật 4: **if shape là “nón” then quyết định là NO.**

Ở dòng còn lại của Bảng_con_2 (dòng 3) chúng ta có thuộc tính [size] có giá trị là “lớn” nhưng nó cũng xuất hiện trong Bảng_con_1, do đó theo thuật toán nó sẽ không được xem xét, tương tự cho “đỏ” của {color} và “cầu” của {shape}. Trong trường hợp bày ILA sẽ tăng giá trị của j lên 1 và sẽ tạo ra 2 thuộc tính trong 1 kết hợp khi đó ta có {[size] và [color]}, {[size] và [shape]} và {[shape] và [color]}. Kết hợp đầu và kết hợp thứ 3 thỏa điều kiện xuất hiện trong Bảng_con_2 nhưng không xuất hiện trong Bảng_con_1 trong khi giá trị “lớn, cầu” của kết hợp {[size] và [shape]} thì không thỏa. Do đó chúng ta chỉ có thể chọn kết hợp đầu hay thứ 3, ở đây kết hợp đầu sẽ được chọn. Khi đó luật 5 sẽ được tạo ra và dòng 3 sẽ được khóa.

Luật 5: **if size là “lớn” AND color là “đỏ” then quyết định là NO.**

Bây giờ tất cả các dòng của Bảng_con_2 đã được khóa và không còn bảng con nào khác nên thuật toán kết thúc.