تعلم القواعد Rule Learning

- تحدثنا في المحاضرة السابقة عن النظم الخبيرة، والتي تسمى أيضاً النظم الخبيرة المبنية على القواعد Rule Based Expert Systems وذلك لأن من العناصر الأساسية للنظام الخبير قاعدة المعرفة الخاصة به Knowledge Base، والتي بدورها تتألف من مجموعة من القواعد Rules (والحقائق facts والتي تعتبر حالة خاصة من الـ Rule)، ونظراً لأهمية النظم المبنية على القواعد، وللجهود البشرية الكبيرة المطلوبة لتحصيل م قواعد الأنظمة الخبيرة اليا.

مكتبة الشرق للخدمات الطلابية هـ: 2115173

- حيث تقوم طرق التعلم الاستنتاجية بخلق قواعد جديدة حول المجال، أي يقوم النظام باستخدام المعارف التي يملكها مسبقاً لاستنتاج قواعد وحقائق جديدة، سنتعلم الخوارزمية المستخدمة لذلك من خلال المثال التالي:

مثال من الدرس L6-TranspCh5 سلايد رقم 43:

- ليكن لدينا نظام بنكي من أجل الموافقة على القروض للزبائن، فعملية الموافقة على إعطاء قرض لأحد الزبائن تكون وفقاً لأسس وقواعد مسبقة ولا تكون بشكل عشوائي، أي يجب أن يكون الزبون محققاً لبعض الشروط حتى يتم الموافقة على إعطائه قرض، لتكن حقائق المسألة هي التالية:

يجب أن يتمّ الموافقة على القرض OK
تخمين الضمانة أكبر من مبلغ القرض APP
للعميل دفعات دورية منتظمة RATING
يتجاوز دخل العميل مصاريفه INC
للعميل صفحة متوازنة ممتازة BAL

ولتكن لدينا المشاهدات التالية في قاعدة بيانات البنك، أي أن الموظفين قد وافقوا أو رفضوا طلبات بعض الزبائن بحسب قيم الحقائق الأخرى وفقاً للجدول التالي:

	APP	RATING	INC	BAL	OK
1	1	0	0	1	0
2	0	0	1	0	0
3	1	1	0	1	1
4	0	1	1	1	1
5	0	1	1	0	0
6	1	1	1	0	1
7	1	1	1	1	1
8	1	0	1	0	0
9	1	1	0	0	0

- ومهمتنا في هذه المسألة هي التوصل إلى القواعد التي تحكم عملية الموافقة على إعطاء قرض لأحد الزبائن أو رفض ذلك.

خطوات الخوارزمية:

١ - نلاحظ أن الهدف هو OK في هذه المسألة، وبالتالي يجب كتابة القواعد التي تعطينا OK كنتيجة لاقتضاءاتها.

٢ - يجب اختيار الرموز التي تكون ضمن القاعدة لتحقيق OK، مبدئياً يجب حساب عدد الأمثلة الموجبة (أي التي تعطي OK=1)، ونلاحظ أنها 4 مرات، ثم نقوم بإيجاد المقدار $\frac{n^+}{n}$ حيث n^+ عدد المرات التي يكون فيها المتحول موجباً مع الـOK وn هو عدد المرات التي يكون فيها المتحول موجباً، وهذا المقدار من أجل المتحولات الأربعة هو:

$$\propto_A = \frac{3}{6} \quad , \quad \propto_N = \frac{3}{6} \quad , \quad \propto_R = \frac{4}{6} \quad , \quad \propto_B = \frac{3}{4}$$

أي أن App يكون موجباً في 6 أمثلة، ويتقاطع مع الـOK في ثلاث أمثلة موجبة، وكذلك حسب البقية. لنسمي هذه الخطوة بإسم خطوة *.

٣ - نلاحظ أن أكبر نسبة بين السابقين هي التي يحققها المتحول B وبالتالي يضاف على أنه حقيقة على قاعدة المعارف وبالتالي تصبح قاعدة المعارف تحوي حقيقتين هما:

$$B \Rightarrow OK$$

٤ - ولكن B لا يكفي أن تكون true حتى يتحقق ok، ذلك لأن النسبة التي حققتها في الخطوة * هي $\frac{3}{4}$ وليست 1 كامل، لذلك يجب أن نبحث عن المتحول الآخر الذي يجب تحققه مع الـBAL حتى يعطينا تقاطعهما الـOK، نقوم بتنفيذ الخطوة * هذه المرة ليس على المتحولات المفردة، وإنما نحسبها مع المتحولات تحقق أنها موجبة مع الـBAL بنفس الوقت، أي النسبة هي:

$$\propto_{AB} = \frac{\text{number of times where A, B and OK are true}}{number\ of\ times\ where\ both\ A\ and\ B\ are\ true}$$

وبحسابها على المتحولات الثلاثة المتبقية في المسألة:

$$\propto_{AB} = \frac{2}{3} \quad , \quad \propto_{RB} = \frac{3}{3} \quad , \quad \propto_{NB} = \frac{2}{2}$$

٥ - والآن نختار واحداً من المتحولات التي تعطي أكبر نسبة ليصبح جزءاً من القاعدة مع B لإعطاء OK، والذي يحقق ذلك هو R، فتصبح القاعدة هي:

$$R1 : R \wedge B \Rightarrow OK$$

لم نأخذ N بدلاً من R وذلك لأن 3 أكبر من 2 مع أن النسبة أيضاً هي 1

أي عندما يتحقق R مع B عندها يتحقق OK، ونقوم بتثبيت هذه القاعدة في قاعدة المعارف لأن النسبة التي تعطيها 1 ولا تحتاج إلى متحول ثالث ليضاف عليها ليصبح ok=true ويكفي تحقق R و B معاً ليتحقق OK، أي لا يوجد أحد الأسطر التي يكون لها الشكل التالي:

R	B	OK
1	1	0

٦ - بعدما نقوم بتثبيت أحد القواعد نقوم بحذف الأسطر التي تحققها من الجدول، أي نحذف الأسطر رقم 3 و 4 و 7 لأنها تحوي :

R	B	OK
1	1	1

www.ite-sy.net

فيصبح الجدول يحمل الشكل:

	APP	RATING	INC	BAL	OK
1	1	0	0	1	0
2	0	0	1	0	0
5	0	1	1	0	0
6	1	1	1	0	1
8	1	0	1	0	0
9	1	1	0	0	0

٧ - بعد الخطوة السابقة نلاحظ أن عدد الأمثلة الموجبة المتبقية هو واحد فقط، أي ok تحقق فقط في مرة واحدة، ويجب تكرار الخطوات من عند * إلى أن يتم حذف هذا السطر الموجب من الجدول وعندها فقط تصبح القواعد كاملة ونتوقف من هذه الحلقة.

٨ - نطبق الخطوة * على المتحولات المتبقية:

$$\propto_A = \frac{1}{4} \quad , \quad \propto_B = \frac{0}{1} \quad , \quad \propto_N = \frac{1}{4} \quad , \quad \propto_R = \frac{1}{3}$$

٩ - نلاحظ أن R يحقق أكبر نسبة فنضيفه على القاعدة الجديدة، ثم نكرر الخطوة * عندما يكون R موجباً مع المتحولات الأخرى:

$$\propto_{AR} = \frac{1}{2} \quad , \quad \propto_{NR} = \frac{1}{2} \quad , \quad \propto_{BR} = \frac{0}{0}$$

نلاحظ أيضاً أن ولا نسبة من النسب السابقة قيمتها 1، فنختار التي تحقق أكبر نسبة ولنأخذ AR ونكرر الخطوة * عندما يكون A و R موجبين ونأخذ معهم مرة N ومرة B ونرى إن كنا سنصل إلى 1 في أحد النسب:

$$\propto_{BAR} = \frac{0}{0} \quad , \quad \propto_{NAR} = \frac{1}{1}$$

وبالتالي نلاحظ أنه عندما يكون A و R وN كلهم موجبين معاً يتحقق OK دوماً محقق هذا الكلام لأن النسبة 1.

١٠ -نضيف القاعدة R2 على قاعدة المعارف التي تصبح:

$$R1 : R \wedge B \Rightarrow OK$$

$$R2 : N \wedge A \wedge R \Rightarrow OK$$

١١ -نحذف السطر الذي يتحقق فيه:

A	R	N	OK
1	1	1	1

وهو السطر السادس، وبالتالي يصبح الجدول هو:

	APP	RATING	INC	BAL	OK
1	1	0	0	1	0
2	0	0	1	0	0
5	0	1	1	0	0
8	1	0	1	0	0
9	1	1	0	0	0

١٢ -لم يتبقى لدينا ولا مثال موجب أي ولا سطر يحوي OK=1 وبالتالي نتوقف عن التكرار وننهي الخوارزمية.

- وبهذه الطريقة نكون قد تعلمنا القواعد باستخدام هذه الطريقة بالتعلم، وهذه الطريقة تسمى طريقة التصنيف classification.

www.ite-sy.net

- وفي هذا المثال كانت لدينا كل المتحولات تأخذ قيماً إما true وإما false أي يوجد حتمية في القرار، أما في أمثلة أخرى قد تكون المتحولات تحمل certainty factors وعندها نبني شجرة القرار decision tree كما قمنا في المحاضرة السابقة.
- يوجد مثال في السلايد 64 من نفس الدرس L6-TranspCh5 قمنا كذلك بالمرور عليه في المحاضرة ولن أقوم بكتابته ذلك لأنه مشروح بشكل جيد في السلايدات وليس بتلك الصعوبة لذلك يمكن قراءته من السلايد.

- **هنا يكون منهاج الذكاء من السلايدات قد انتهى وفي جميع المحاضرات القادمة سنقوم بحل مسائل على جميع التقنيات التي قمنا بدراستها وكذلك مسائل من دورات المادة السابقة، هذا لا يعني أن ما يأتي هو ليس مهماً، ذلك لأن أغلب أسئلة الفحص ستكون مسائل وتمارين ونسبة النظري متدنية (إن لم تكون معدومة ☺).**

- تمرين دورة: إذا كانت G زمرة، بحيث أن كل عنصر مقلوب نفسه يجب أن نبرهن أن G تبديلية من الحقائق التالية:

$$H1: \forall x \forall y \forall z\ [(x.y).z = x.(y.z)]$$
$$H2: \forall x\ \ x.e = e.x = x$$
$$H3: \forall x\ \ x.x = e$$

أي يجب إثبات $x.y = y.x$. المطلوب هو كتابة صياغة مناسبة للمسألة.
الحل:
يكفينا predicate واحدة هي p(x,y,z) والتي تعني أن x.y=z، وبالتالي نكتب الحقائق السابقة باستخدام هذه الـpredicate:

H2 : p(x,e,x)
p(e,x,x)
H3 : p(x,x,e)
H1 : p(x,y,u) ^ p(y,z,v) ^ p(u,z,w) ⇒ p(x,v,w)
p(x,y,u) ^ p(y,z,v) ^ p(x,v,w) ⇒ p(u,z,w)

بالنسبة لكيفية الحصول على كل واحدة منها فإن H2 واضحة ولا تحتاج لشرح، وH3 كذلك، أما بالنسبة لـH1:
العملية x.y قمنا بتخزين ناتجها في u، ومن ثم خزنا ناتج العملية y.z بالمتحول v، وفيما بعد حسبنا ناتج العملية $(x.y).z$ ووضعناه في المتحول w والذي يمثل خرج كل الطرف الأيسر، والطرف الأيمن يمثل ناتج العملية $x.(y.z)$ لأننا كما ذكرنا y.z=v، ويمكن التعبير عنها بالـprolog على الشكل التالي:

p(x,v,w):- p(x,y,u), p(y,z,v) , p(u,z,w).

ويجب التعبير عنها أيضاً بالاتجاه المعاكس، أي يجب أن ننطلق من الطرف اليميني من العلاقة أي $x.(y.z)$ ومن ثم نصل إلى الطرف الأيسر: (ذلك لأن عملية التساوي يجب أن تتحقق، ولتفسير ما قمنا به هنا يمكن أن نذكر مثال مشابه في البيان غير الموجه، فعندما يكون البيان غير موجه وتوجد وصلة بين العقدة A والعقدة B فيجب كتابة path(A,B) وكذلك path(B,A) حتى يفهم البرولوغ أن الطريق بالاتجاهين، وهذا ما نقوم به هنا في هذا المثال)، أي نكتب:

p(x,y,u) ^ p(y,z,v) ^ p(x,v,w) ⇒ p(u,z,w)

والتي أيضاً تكتب بالبرولوغ على الشكل:

p(u,z,w):- p(x,y,u), p(y,z,v), p(x,v,w).

ويكون الهدف المطلوب هو: p(x,y,z) ⇒ p(y,x,z) .

- تمرين دورة آخر عن الـunification:

هل يمكن توحيد p(f(y,f(y,A)),A) مع p(f(x,x),A)؟
الحل:
لنقوم بذلك يجب تعويض قيم العبارة الأولى في الثانية لنتأكد من إن كنا نستطيع القيام بعملية unification أم لا، بتعويض y في x تصبح العبارة الثانية هي:

p(f(y,x),A)

ومن ثم نعوض الـf(y,A) في x الثانية:

p(f(y,f(y,A)),A)

فنلاحظ أن ذلك غير ممكن لأن قيمة y هي ليست نفسها f(y,A) وبالتالي فإن عملية التوحيد غير ممكنة بينهما.

- تمرين prolog:

أوجد predicate ترد true إن كانت السلسلة التي اسمها M هي جزئية من السلسلة L، أي سيكون لها فقط وسيطين هما السلسلة الأولى والتي سنختبر كونها جزئية من الثانية.
مثال:

M=[b,c] is subset of L=[a,b,c,d]

الحل: هذا الحل هو الذي اقترحه الدكتور في البداية:

```
sub([],_).
sub([X|Q1],[X|Q2]):-sub(Q1,Q2).
sub([X|Q1],[Y|Q2]):-X=\=Y,sub([X|Q1],Q2).
```

ولكن ذكر بعدها أنه حل ناقص، وفيه مشكلة، ولم يذكر ما هي المشكلة وتركها وظيفة لنحلها بأنفسنا.
والحل الذي سأطرحه هو حل شخصي (يلي بيلاقي فيو مشكلة يخبرني☺) بعد أن كشفت أن المشكلة هي التالية:
البرنامج يعطي false في الحالات التالية:

```
sub([1,2],[15,4,2]).
sub([1,2],[1,4,5]).
```

وفي الحالات السابقة يجب أن يعطي false أي أنه لم يغلط.
ولكن الحالة التي يجب أن يعطي فيها false وهو لا يعطي فيها false هي:

```
sub([1,2],[1,4,2,5]).
```

أي أن الحل السابق ينقصه أمر مهم وهو مراعاة ترتيب ظهور عناصر السلسلة الأولى في السلسلة الثانية، أي في المثال السابق نلاحظ أن 1 ظهرت أول عنصر و2 ظهرت ثالث عنصر، مع العلم أنه يجب أن تكون في المكان التالي تماماً للعنصر 1. (اذا قمنا بتتبع تنفيذ البرنامج السابق على هذا المثال نلاحظ أن البرنامج يحمل هذه المشكلة بالفعل)
الحل المقترح للمشكلة:
يجب في كل مرة التأكد من أن العنصر التالي لرأس السلسلة الأولى هو نفسه العنصر التالي لرأس السلسلة الأخرى، أي على مثالنا يجب أن يكون 2 هو العنصر التالي للـ1 في كل من السلسلتين، وفي مثال آخر:

```
sub([1,2,3],[4,1,2,3]).
```

فيجب أن نضمن أن 2 هو التالي لـ1 في السلسلة الثانية الكبيرة، وكذلك 3 هو التالي لـ 2 في السلسلة الكبيرة.
أي أن فكرة الحل تقوم على التأكد فقط من أن العنصر التالي للعنصر الحالي من السلسلة الأولى هو نفسه العنصر التالي له في السلسلة الثانية، لذلك نحن بحاجة إلى predicate تتأكد من ذلك ولتكن بإسم

www.ite-sy.net

is_adjacent_in_list نعطيها سلسلة وعنصرين وتتأكد من أن هذين العنصرين هما متتاليين في السلسلة، ونستدعيها عند كل عنصر من عناصر السلسلة الأولى ونعطيها العنصر الحالي والعنصر التالي له، ولنوجد العنصر التالي له في السلسلة الأولى نحتاج إلى predicate أخرى تعطينا العنصر الثاني من السلسلة ولتكن بإسم second، وبالتالي البرنامج كاملاً هو:

```
is_adjacent_in_list(X,Y,[X,Y|_]).
is_adjacent_in_list(X,Y,[_|W]):-is_adjacent_in_list(X,Y,W).
second([],_):-!,fail.
second([_,Y|_],Y).
sub(_،[]).
sub([X],[X|_]).
sub([X|Q1],[X|Q2]):-second([X|Q1],S),is_adjacent_in_list(X,S,[X|Q2]),sub(Q1,Q2).
sub([X|Q1],[Y|Q2]):-X=\=Y,sub([X|Q1],Q2).
```

ويمكن التأكد من خرجه بتطبيقه على برنامج البرولوغ.
وكما ذكرت هو حل شخصي وليس بالضرورة أن يكون صحيحاً 100%، وأرجو إخباري في حال ورود أي خطأ فيه.

- تمرين دورة آخر:

$R1: H \Rightarrow B , cf_{R1}=0.3$
$R2 : E \Rightarrow B , cf_{R2}=0.6$
$R3 : H \Rightarrow S , cf_{R3}=0.6$
$R4 : E \Rightarrow S , cf_{R4}=0.9$

احسب كلاً من cf_B , cf_S؟ (حيث أن $cf_H=0.5$ و $cf_E=0.5$)

الحل:

نلاحظ أن المتحول B نحصل عليه من العبارتين R1 وR2 وبالتالي يوجد or بين العبارتين، ولهذا نحسب قيمة cf_B من العبارة الأولى ومن ثم من العبارة الثانية ومن ثم نأخذ الـmax لوجود or بين العبارتين:

$cf_B \text{ from } R1 = cf_H * cf_{R1} = 0.5*0.3 = 0.15$
$cf_B \text{ from } R2 = cf_E * cf_{R2} = 0.5*0.6 = 0.3$

وبالتالي $cf_B=0.3$ لأن $0.3>0.15$.

والمتحول S نحصل عليه من العبارتين R3 وR4 ونكرر نفس الخطوات عليه مثل المتحول B:

$cf_S \text{ from } R3 = cf_H * cf_{R3} = 0.5*0.6 = 0.3$
$cf_S \text{ from } R4 = cf_E * cf_{R4} = 0.5*0.9 = 0.45$

وبالتالي $cf_S=0.45$ وذلك لأن $0.45>0.3$

انتهت المحاضرة

www.ite-sy.net