جامعة حماه

الكلية التطبيقية

قسم تقنيات حاسوب

السنة الثالثة

مبادئ الذكاء الصنعي

مدرس المقرر : م. رغد العمادي

العام الدراسي 2019-2020

مفهوم الذكاء الاصطناعي Intelligent Artificia(AI)

هناك مئات التعاريف للذكاء الصنعي، يختلف كل تعريف عن الآخر وفقا لرؤية كل شخص،

وله بعدين (التفكير thinking ، الأفعال Acting .)

حيث ال **Thinking** عبارة عن نمذجة لعملية التفكير والاستنتاج والتعلم، وعندما يتحول إلى سلوك وتصرفات يصبح **Acting** (كالروبوت الآلي مثلا).

تعريف: الذكاء الاصطناعي (AI)

هو علم وهندسة في آن واحد،

- علم: حيث يقوم بفهم الأشياء الذكية.
- هندسة: اعتماداً على هذا العلم نقوم بنمذجة الأشياء.

هناك أربعة وجهات نظر للذكاء:

1) التفكير البشري AI mean thinking humanly

والتي تعتمد بشكل كبير على العلوم الإدراكية Cognitive science ، حيث تنمذج طريقة تفكير الإنسان ومحاولة معرفة كيف يعمل الدماغ.

مثال: عملية التذكر وعملية التعرف على الأصوات

2) التصرف البشري humanlyAI mean acting

مبنية على أساس اختبار تورينغ Turing test ، حيث في عام 1950 وضع عالم الرياضيات الشهير Alan Turing أول حدث مسجل كبداية لعلم الذكاء وهو اختبار تورينغ حيث وضع بحث عنوانه

"computing machine and intelligent "، فطرح السؤالين التاليين:

1) Can machine think? هل يمكن للآلة أن تفكر

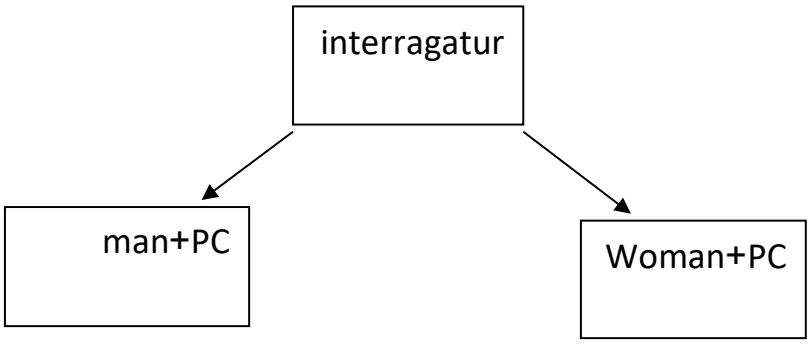
2) Can machine behave intelligently? هل يمكنها التصرف بذكاء

وللإجابة على هذين السؤالين وضع هذا العالم اختبار سمي باسمه،

وبشكل مجرد Abstract يوصف هذا الاختبار بأنه محقق للآلة والإنسان.

الاختبار عبارة عن لعبة محاكاة TM (في اللغات الصورية والأوتومات):

هناك ثلاث غرف، يوجد في الغرفة الأولى حاسب ورجل، في الغرفة الثانية يوجد حاسب وامرأة، أما الغرفة الأخيرة فتحوي حاسب ومستجوب (له الحق في طرح أي سؤال)، علما أن هذا المستجوب لا يعلم في أي غرفة يوجد الرجل وفي أي غرفة توجد المرأة، فتأتي مهمته بأنه يجب أن يعرف في أي غرفة الرجل وفي أي غرفة المرأة...



في الطور الأول:

المرأة تحاول إقناعه بأنها امرأة.

والرجل يحاول خداعه ووهمه بأنه هو الامرأة

في الطور الثاني:

نترك في غرفة الرجل فقط الحاسب، إذا استطاع الحاسب خداع المستجوب كما فعل الرجل سابقاً

نقول أن الحاسب استطاع اجتياز اختبار السلوك الذكي (وذلك لمدة خمس دقائق).

السلوك الذكي: هو الاستفادة من التجارب وإضافة المعارف الجديدة لها لتستخدمها في أمور أخرى

متطلبات الحاسب حتى يظهر على شكل إنسان:

1 . معالجة اللغات الطبيعية NLP: Nature languages processing

2 . التمثيل المعرفي Know representation

3 . استنتاج مؤتمت Automated reasoning

4 . تعلم الآلة Machine learning

3) AI means Thinking Rationally التفكير بشكل عقلاني

تستخدم علوم المنطق الرياضي.

4) Al means Acting Rationally

تعمل بشكل عقلاني، كالوكلاء الأذكياء Intelligent Agents .

بمعنى اخر المفهوم الرابع يهتم بالمفاهيم الثلاثة الأولى لذلك سنهتم بدراسة الرؤية الرابعة

تعريف الوكيل Agent

An Agent is anything that can be viewed as perceiving its environment through sensors and Action upon that .environment through actuators

هو أي شيء يستطيع إدراك (فهم) بيئة من خلال الحساسات ...ويتصرف على تلك البيئة من خلال محركات.

تعريف البيئة Environment :

هو العالم المحيط بالوكيل...

إذا اعتبرنا أن الوكيل عبارة عن برنامج فإن البيئة المحيطة للوكيل في هذه الحالة هي الحسابات المبيتة

في هذا الوكيل (الموجودة في هذا الوكيل) "الانسان يعتبر وكيل"

تعريف الوكيل الذكي IA (Intelligent Agent) :

IA is a system or program that perceives its environment and takes Action which maximizes its chances of success.

هو نظام أو برنامج يفهم (يدرك) بيئة.. ويتخذ القرارات (الأفعال) التي تعظم من فرص نجاحه.

بمعنى آخر: قادر على التعلم (الاستجابة) من خلال المدخلات والمخرجات وخوارزمية التعلم الموجودة داخله...

أي وكيل نضيف له بعض الذكاء يتحول إلى وكيل ذكي.. فإذا تم ملء؟ في الشكل السابق بالذكاء.. لأصبح الوكيل فيه ذكيا

حصلنا على مفهوم الذكاء الصنعي بناء على مفهوم IA

أحد أهم أنواع الوكلاء الأذكياء:

الوكيل المنطقي Logical Agent :

وسوف نتعرف على ما يسمى ب

" الوكيل المبني على المعرفة: Knowledge – Based Agent

تعريف: الوكيل المنطقي.. Logical Agent :

In which we design Agent Than can from representation of a complex world, use a process of inference (Reasoning) to drive new representation about the world. And use the new representation to deduce what to do…

Deduction → induction → abduction

فيه نستطيع تصميم وكلاء التي نستطيع منه تمثيل معقد للعالم (البيئة المحيطة) مستخدما عملية الاستدلال (الاستنتاج) لاشتقاق تمثيلات جديدة حول العالم المحيط ويستخدم أيضا هذه التمثيلات الجديدة لاستنتاج ماذا بإمكانه أن يقوم بعمله...

NoTe $:y=x^2$

$y'=2x$

لا يمكن برهان السابق باستخدام الاستقراء الرياضي .. حيث الاستقراء يطبق فقط على الاعداد الطبيعية .... ولكن هدا التابع مستمر

الاستنتاج المنطقي.. Logical Reasoning :

مثال: <هالكون>

All men are mortal

Socrates is a man

There for: Socrates is mortal

بفرض لدينا مجموعة من المقدمات.. بناءً عليها نستطيع التوصل لنتيجة وتخزينها لاستخدامها في مواقف أخرى...

الوكيل المبني على المعرفة... Based Agent -Knowledge

NoTe : الفرق بين المعلومات والمعرفة: أن المعلومات Free متاحة للجميع, بينما المعرفة man Free قيّمة أكثر, يمكن أن تكون متاحة لأيّ شخص.(بيانات ← معلومات ← معرفة)

المكوّن الأساسي للوكيل المبني على المعرفة هو Based Agent -Knowledge قاعدة ( ) KB

المعرفة...

قاعدة المعرفة: (KB) وهي مجموعة من الجمل.. ومجموعة هذه الجمل يعبّر عنها عن طريق ما يسمى:

لغات تمثيل المعرفة Knowledge Representation Languages

وكل لغة تمثل معرفة تكون مزودة بآلية للحصول على عبارات جديدة من عبارات قديمة تدعى آلية الاستدلال...

أشهر لغات التمثيل:

المنطق الرياضي Mathematic logic :

Logic : هو الدراسة العلمية لعملية التفكير (كيف تتم عملية التفكير) ...

Formal logic : المنطق الشكلي "الصوري", احيانا يدعى المنطق المرمّز Symbolic logic

لأنه يتعامل مع صورة الأشياء، دون أن يدخل في المضمون...

وله نوعين:

1. **منطق الفرضيات Logic Proportional :**

منطق المرتبة الصفرية (PL)Zero –order logic

2. **Predicate Logic :**

منطق المرتبة الأولى (PL)First –order logic

منطق الفرضيات **Logic Proportional**:

مبني على الفرضيات، الفرضية عبارة أو جملة تحتمل لصح أو الخطأ:

$ex{:}it's\ raining\{TrueFalse\}$

سوف ندرس منطق الفرضيات كلغة من لغات تمثيل المعرفة:

أي لغة تمثيل معرفة، تمتلك المكونات التالية:

- ► تُعنى بشكل العبارات Syntax:
- ► تُعنى بإعطاء المعاني (شكل عبارة معنى) Semantic:
- ► (قواعد الاستدلال) نظرية اثبات Rules of inference (Proof Theory)

Syntax في PL اي عبارة عن فرضيات (تمثل مفردات اللغة) بالتالي تحتمل هذه الفرضيات

الصّح أو الخطأ...

$\Sigma\{P1, P2, \ldots\ldots, Pn,$ $\sim, \wedge, \vee, \rightarrow, \leftrightarrow,$ $(,)$ $True, false\}$

لتحديد أولويات العمليات الحسابية الروابط المنطقية فرضيات

حيث أيّ عبارة منطقية تصاغ باستخدام هذه المكونات...

تعريف: التفسير Interpresentation :

تخصيص قيمة $True, false$ لكل فرضية من فرضيات العبارة المنطقية..

مثال: اذا كان لدينا العبارة المنطقية المكونة من الفرضيات الثلاثة A, B, C :

$$W=(A\cup B)\cap C$$

تفسير 1 $\begin{cases} A=F \\ B=T \\ C=F \end{cases}$ $(F\cup T)\cap F=F$ هذا التمثيل خصص لهذه العبارة المنطقية القيمة False

تفسير 2 $\begin{cases} A=T \\ B=T \\ C=T \end{cases}$ $(T\cup T)\cap T=T$

يوجد 2^3 تفسير للعبارة المنطقية السابقة...

بالتعميم: اذا كان لدينا عبارة منطقية ب n من الفرضيات... يكون مجموع التفاسير 2^n

تعريف: Model :

هو تفسير يعطي قيمة حقيقة $True$ للعبارة المنطقية (التفسير 2 السابق هو Model .)

مثال: ليكن:

$Model \leftarrow A \cap B$ ... هده العبارة هو - T,T

$Model \leftarrow A \cup B$ هده العبارة هو احداهما او كلاهما $true$ نحصل على $Model$

- العبارة المنطقية التي تكون صحيحة في كل ال $Model$ تدعى توتولوجي $totology$
  - $A \cup \sim A$ ≪ في كل الحالات $true$ ≫
- العبارة المنطقية التي تكون خاطئة في كل ال $Model$ تدعى تناقض $contradictio$
  - $A \cap \sim A$ ≪ في كل الحالات $false$ ≫
- العبارة المنطقية التي لها $Model$ واحد على الأقل تدعى مُرضية $Satisfiable$ ...

≪ آخر فرض في خاطئة و ما فرض في صحيحة ≫

أدوات الربط:

1_ and logic أو ما يعرف برابط "وَ" المنطقي:

و هو رابط يربط بين عبارتين أو أكثر و هذه العبارات قد تكون بسيطة أو مركبة و يكون ناتج العبارة الجديدة يحمل القيمة True إذا وفقط إذا كانت القيمة الحقيقة لكل من العبارتين تحمل القيمة true .

رياضياً: نعبر عن الرابط logic and بالشكل (p ∧ q)

2_ logic or أو ما يعرف برابط "أو" المنطقي:

و هو أداة ربط تربط بين عبارتين أو اكثر من النوع البسيط أو المركب و يكون ناتج العبارة الجديدة يحمل القيمة false إذا و فقط إذا كانت قيمة الحقيقة لكل من العبارتين المشكلتين تحمل

القيمة false .

رياضياً: نعبر عن الرابط logic or بالشكل (p v q)

3_معامل not رمزه ¬ وهو الرابط الوحيد الذي يمكن له التعامل مع العبارة من النوع البسيط:

(لكنها ليست قاعدة) فيمكن أن يتعامل مع العبارات البسيطة simple أو المركبة compound

في هذا الرابط يكون ناتج العبارة المشكّلة هو true إذا كان أساس العبارة يحمل القيمة false

أو العكس .

مثال: لتكن لدينا العبارتين

.p: peanut is the source of protien

.q: meat is the source of protien

المطلوب: عبر لغوياً عن الروابط المنطقية التالية:

(p ∧ q) , ¬ q

الحل :

(p ∧ q): * peanut and meat are the source of protein

القيمة: (true)

¬ q : * peanut is not the source of protein

القيمة: (false)

4_الرابط المنطقي if condition then statement أو الرابط "إذا كان فإن":

وهو رابط منطقي يربط بين عبارتين أو أكثر من النوع البسيط أو المركب و يكون ناتج العبارة

صحيح في جميع الحالات باستثناء الحالة التي تكون فيها العبارة التي تعبر عن الشرط

condition صحيحة و العبارة التي تمثل النتيجة خاطئة.

ملاحظة: يعد الرابط if then هو الرابط الوحيد الذي يولي أهمية لترتيب العبارات بحيث تكون

دائماً إحدى هذه العبارات هي الشرط الذي ينتج عنه العبارة الثانية والتي ندعوها بالنتيجة

* يجب أن يكون الشرط condition واضح

* في حال كان أكثر من statement تعطى الأولوية ابتداءً من أقصى اليسار إلى اليمين

رياضياً: نعبر عن الرابط بالشكل

(condition) p ⟹ q (statement)

(q تقتضي p)

اما باللغة المحكية هناك عدة أشكال لتمثيل هذا الرابط:

1. if p then q
2. p is sufficient for q
3. p is sufficient condition for q
4. q is necessary for p
5. q is necessary condition for p
6. p only if q

حالات خاصة لِ q ⟵ p

* خطأ صح: " if + conditon " عبارة خاطئة (صحيح لايؤدي إلى خاطئ)

* صح خطأ: " if + condition

5_الرابط if and only if : أو ما يعرف بالشرط اللازم والكافي هو رابط يصل بين عبارتين أو أكثر

من النوع البسيط أو المركب و يكون ناتج العبارة الجديدة يحمل القيمة true في الحالة التي

تتشابه فيها قيمة الحقيقة للعبارتين المكونتين للرابط.

رياضياً: نعبر عن هذا الرابط بالشكل

تعريف التكافؤ المنطقي:

العبارتين المصاغتين بشكل جيد α, β

(أي Syntax لهما صحيح حسب قواعد PL .)

متكافئين منطقيا .. إذا كان لهما نفس قيم الحقيقة.. ونرمز لذلك بالرمز $\alpha \equiv \beta$

قواعد التكافؤ المنطقي:

- خاصة التبديلية: $(\alpha\cap\beta)\equiv(\beta\cap\alpha)$

$(\alpha\cup\beta)\equiv(\beta\cup\alpha)$

- الخاصة التجميعية (لا أهمية لوجود الأقواس) $(\alpha\cap\beta)\cap\gamma \equiv \alpha\cap(\beta\cap\gamma)$

$(\alpha\cup\beta)\cup\gamma \equiv \alpha\cup(\beta\cup\gamma)$

- نفي النفي: $\alpha\equiv(\alpha\sim)\sim$
- خاصة المعكوس الإيجابي: contrapositive $(\alpha\rightarrow\beta)\equiv(\sim\beta\rightarrow\sim\alpha)$
- إزالة واستدلال Implication – elimination : $(\alpha\rightarrow\beta)\equiv(\sim\alpha\cup\beta)$
- قانوني دي مورغان (الخاصة التوزيعية): $\alpha\cap\beta\equiv(\neg\alpha\cup\neg\beta)\neg$

مثال: لتكن لدينا العبارات الثلاث التالية:

p: Omar goes for a walk

q: The sun is out

r: It's raining

المطلوب: عبّر لفظياً عن العبارات التالية ثم مثل كل عبارة بجدول الحقيقة الموافق لها.

$(q \wedge \neg r) \longrightarrow p$:

.If the sun is out and it is not raining then Omar goes for a walk

$(q \longrightarrow \neg r) \longrightarrow p$:

If the sun is out then if it is not raining then Omar goes for a walk.

$\neg (P \longleftrightarrow (r \vee q))$:

It is not the case that , Omar goes for a walk if and only if it is raining or the sun is out

قواعد الاستدلال Inference :

من خلال قواعد الاستدلال يمكننا اثبات أن $K: \beta \vDash \alpha$ عندما يكون عدد الفرضيات كبير..

تعريف الاستدلال: عملية تتم من خلالها اشتقاق عبارات جديدة من عبارات موجودة في قاعدة المعرفة...

(الاستنباط reasoning هو الحالة الأعم من الاستدلال)

Proof : (البرهان): سلسلة من الاستنتاجات تعود إلى الهدف المطلوب...

Theorem Proving : برهان النظرية: هي مجموعة القواعد لاستنتاج الاستتباعات لمجموعة من العبارات:

حيث: $(Knowledge-Based)(\beta \vDash \alpha)$

هو Theorem Proving ونبرهن على صحة هذه النظرية عن طريق الاستدلال باستخدام مجموعة من قواعد الاستدلال – α هي Theorem (نظرية)...

وسميت **نظرية**: لأنه عندما سوف نبرهن $K: \beta \vDash \alpha$ يجب أن نبرهن صحة α من مجموعة من القواعد المعطاة...

قواعد الاستدلال:

قاعدة الاستدلال: هي قاعدة لها الشكل العام التالي:

$$\frac{\alpha 1 \ ,\alpha 2 \ldots\ldots \alpha n}{\beta}$$

بالأعلى: تسمى مقدمات أو فرضيات Premises

بالأسفل: نتيجة: Conclusion

أيّ: من مجموعة المقدمات $\alpha 1 \ ,\alpha 2 \ldots\ldots \alpha n$ يمكن الوصول إلى β

- من قواعد الاستدلال:

1_ MP : Mudus Ponens $\frac{\alpha\rightarrow\beta\ ,\alpha}{\beta}$

2_ E∩ : And – Elimination∩ $\frac{\alpha\cap\beta}{\beta}$ $\frac{\alpha\cap\beta}{\alpha}$

3_ I∩ : And – Introduction∩ $\frac{\alpha\ ,\beta}{\alpha\cap\beta}$

4_ I : Or – Introduction∪ $\frac{\alpha\ ,\beta}{\alpha\cup\beta}$

5_ Unit – Resolution $\frac{\alpha\cup\beta\ ,\sim\beta}{\alpha}$

6_ Resolution $\frac{\alpha\cup\beta\ ,\sim\beta\cup\gamma}{\alpha\cup\gamma}$

7_ $e\Leftrightarrow$ $\frac{\alpha\leftrightarrow\beta}{(\alpha\rightarrow\beta)\cap(\beta\rightarrow\alpha)}$

8_ $i\Leftrightarrow$ $\frac{(\alpha\rightarrow\beta)\cap(\beta\rightarrow\alpha)}{\alpha\leftrightarrow\beta}$

NoTe : الفرق بين ≡ و ↔ :

↔ : رابط منطقي سوا ء بين عبارتين أو جملتين (تكون صحيحة إذا كلاهما T أو كلاهما F ....)

≡ : التكافؤ المنطقي ما بين عبارتين (تكون صحيحة إذا كان لهما نفس النموذج) ...

أي: $(\alpha\equiv\beta)\ iff\ (\alpha\leftrightarrow\beta\ ToTology)$

مثال

ليكن لدينا مجموعة القواعد التالية :

P : أنت مصاب بالرشح .

q : أنت اجتزت الامتحان النهائي .

r : أنت اجتزت الدورة التدريبية .

عبر عن القضايا التالية منطقياً :

- أنت مصاب بالرشح إذا أنت فشلت بالامتحان النهائي $p \rightarrow \neg q$

- أنت فشلت بالامتحان النهائي إذا أنت لم تجتز الدورة التدريبية $\neg q \rightarrow \neg r$

- إما أنت مصاب بالرشح عندئذ أنت لم تجتز الدورة التدريبية أو أنت فشلت في الامتحان النهائي عندئذ أنت لم تجتز الدورة التدريبية

$(p \rightarrow \neg r) \vee (\neg q \rightarrow \neg r)$

- إما أنت مصاب بالرشح وفشلت بالامتحان النهائي أو أنت لم تفشل بالامتحان النهائي واجتزت الدورة التدريبية $(p \wedge \neg q) \vee (q \wedge r)$

<u>برهن باستخدام قواعد التكافؤ المنطقية صحة التكافؤ التالي :</u>

1. **$(p \leftrightarrow q) \Leftrightarrow ? (p \wedge q) \vee (\neg p \wedge \neg q)$**

$L1 : (p \rightarrow q) \wedge (q \rightarrow p) \Leftrightarrow (\neg p \vee q) \wedge (q \rightarrow p)$

$\Leftrightarrow (\neg p \vee q) \wedge (\neg q \vee p) \Leftrightarrow (\neg p \wedge \neg q) \vee (q \wedge p) = L2$

2. **$(p \rightarrow r) \vee (q \rightarrow r) \Leftrightarrow ? (p \wedge q) \rightarrow r$**

$L1: (\neg p \vee r) \vee (\neg q \vee r) \Leftrightarrow (\neg p \vee \neg q) \vee r$

$\Leftrightarrow \neg (p \wedge q) \vee r \Leftrightarrow (p \wedge q) \rightarrow r = L2$

<u>برهن باستخدام قواعد التكافؤ المنطقية صحة العلاقة التالية (أثبت أنها totology)</u>

[(p ∧ ٦ (٦ p ∨ q) ∨ (p ∧ q)] → p

⇔ [(p ∧ (p ∧ ٦ q)) ∨ (p ∧ q)] → p

⇔ [((p ∧ p) ∧ ٦ q) ∨ (p ∧ q)] → p

⇔ [(p ∧ ٦ q) ∨ (p ∧ q)] → p

⇔ ٦ [(p ∧ ٦ q) ∨ (p ∧ q)] ∨ p

⇔ (٦ p ∨ q) ∧ (٦ p ∨ ٦ q)] ∨ p

⇔[(٦ p ∨ q) ∨ p] ∧ [(٦ p ∨ ٦ q) ∨ p]

⇔[(٦ p ∨ p) ∨ q] ∧ [(٦ p ∨ p) ∨ ٦ q]

⇔ ($T0$ ∨ q) ∧ ($T0$ ∨ ٦ q)

⇔ $T0$ ∧ $T0$ ⇔ $T0$

(p ^ q) → (p ∨ q)

⇔٦ (p ^ q) ∨ (p ∨ q)

⇔ (٦ p ∨ ٦ q) ∨ (p ∨ q)

⇔ (٦ p ∨ p) ∨ (٦ q ∨ q) ⇔ $T0$ ∨ $T0$ ⇔ $T0$

(p ^ q) → p

⇔٦ (p ^ q) ∨ p

⇔٦ p ∨ ٦ q ∨ p

⇔$T0$ ∨ ٦ q ⇔ $T0$

Artifical Intelligence

أنواع علوم الذكاء :

AI : Artifical Intelligence الذكاء الصنعي

CI : computational Intelligence الذكاء المحسوب

SI :swarm intelligence ذكاء الاسراب

الذكاء الصنعي : هو علم يقوم بعملية محاكاة للأداء الذكي للإنسان

الذكاء المحوسب : ظهر مصطلح الذكاء المحوسب عام 1990 وهو على ثلاثة أقسام رئيسية

1- Fuzzy computation حساب ضبابي
2- Evolutionarily computation الحوسبة التطورية
3- Neural computation الحوسبة العصبونية

1-حساب ضبابي: يحاكي عملية التفكير عند الإنسان.

2-الحوسبة التطورية : تحاكي عملية التطور البيولوجي عند الكائنات الحية وهي تحاكي نظرية دارون في النشوء والارتقاء .

3- الحوسبة العصبونية: تحاكي عمل الدماغ عند الانسان

(أي المقدرة على التعلم والتذكر).

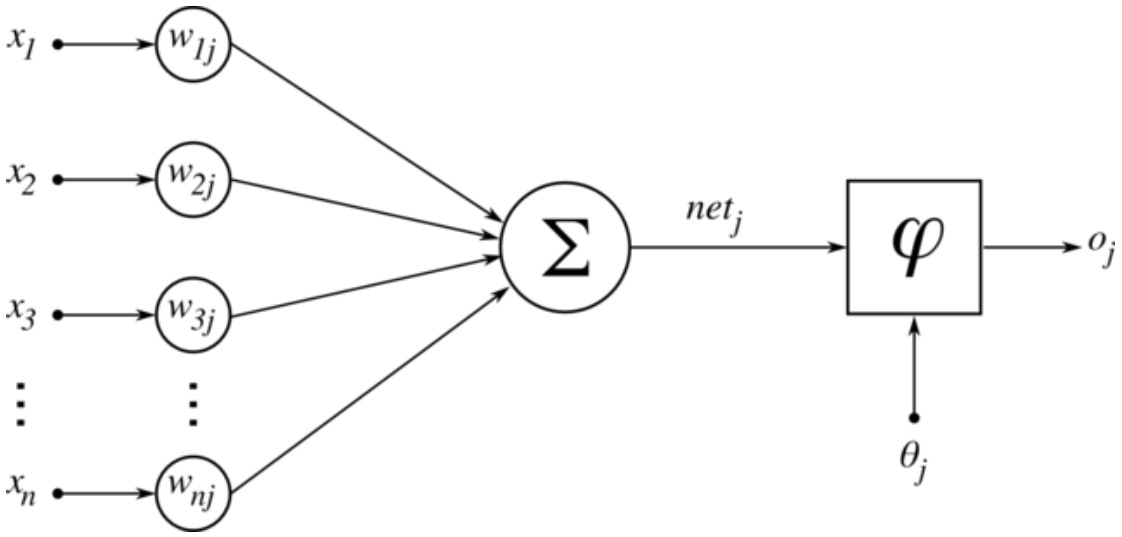
الخلية العصبونية (العصبون البيولوجي): لا تتكاثر لأنها تخزن المعلومات خلال فترات طويلة , الخلية العصبونية تستقبل المعلومات وتحتفظ بها وتستعيدها

العصبون يتألف من 4 أقسام رئيسية :

تفرعات الخلية العصبونية , نقاط اشتباك العصبون , جسم الخلية , المحور العصبي

والمطلوب تحويل هدا العصبون البيولوجي إلى ما يسمى : Artifical Neuron

العصبون الصنعي :



كيفية بناءه :

X=[x1 x2 x3 ------xn] شعاع الدخل

W=[w1 w2 w3 ------wn]شعاع الوزن

$Net = \sum wi\ xi = w_1x_1 + w_2x_2 + w_3x_3 + \text{--------} + w_nx_n$

$$F(net) = \begin{cases} 1 \ ; \ net \geq \varphi \\ -1 \ ; \ net \leq \varphi \end{cases}$$

حيث φ عتبة معينة

إذا كان التابع يساوي 1 عندها يكون في حالة استثارة (نشط) إذا كان التابع يساوي -1 عندها يكون في حالة تثبيط . لدلك التابع f(net) يسمى تابع تنشيط ((التفعيل))

مثال : بفرض لدينا شعاع الدخل وشعاع الوزن احسب خرج العصبون .

X=[1 2 3]

W=[-2 4 3]

φ = 5

الحل : f(net)=1 وهو في حالة فعال .

آلية التعلم learning :

إذا كان لدينا المعطيات التالية :

X=[1 2 3 4]

Y=[1 4 9 16]

ماهي العلاقة بين x وy ؟

Y هو مربع x .

وقد علمنا دلك عن طريق التدريب والتعليم .

أي أثناء التعلم نعطي نموذج للدخل والخرج ثم نكرر هدا النموذج على أمثلة أخرى ثم تكتسب الخبرة عن طريق العديد من المسائل .

إذا كان لدينا الدخل التالي

input	output
(1,2,3)	5
(2,3,4)	11
(3,4,5)	19

العلاقة بينهما : ضرب الأول بالأخير ثم نجمعها مع الحد الأوسط

الدخل مع الخرج في الشبكات العصبونية تسمى أنماط التدريب **training (patterns)**

نبدأ بتدريب الآلة على نموذج فتصبح قادرة على ما يسمى التعميم **generalization**

أي التعميم قادر على اكتشاف حلول في مسائل غير موجودة في أنماط التدريب

أمثلة عن الشبكات العصبونية :

مثل برامج موجودة على أجهزة الموبايل : التعرف على الصوت – على بصمة اليد- التعرف على خط اليد.

عينة عن توابع التفعيل الشهيرة :

1) $F(net)= \frac{2}{1+e^{-ynet}} -1$; y> 0

هذا التابع معرف على R ويأخذ قيمة ضمن المجال المغلق [-1,+1] ولكن يدنو من 1 ولا يساويها لأن عندما

Net $\rightarrow +\infty$ فبالتالي $f(net) \rightarrow 1$

Net $\rightarrow -\infty$ فبالتالي $f(net) \rightarrow -1$

وهو مار من المبدأ لأن $f(net) = 0$ فبالتالي Net = 0

وهذا التابع يسمى : تابع مستمر قطبي ((bipolar continuous function))

2) $F(net)= \frac{1}{1+e^{-ynet}}$; y> 0

هذا التابع معرف على R ويأخذ قيمة ضمن المجال المغلق [0,+1] ولكن يدنو من 1 ولايساويها

وهذا التابع يسمى : تابع مستمر لاقطبي ((unbipolar continuous function))

3) bipolar binary function

$$f(net)=\begin{cases} +1 & ; net \geq 0 \\ -1 & ; net < 0 \end{cases}$$

ويأخذ توابعه وفقا لإشارة ال net لذلك أحيانا يسمى بتابع الإشارة sing function

4) unbipolar binary function

$$f(net)=\begin{cases} +1 & ; net \geq 0 \\ 0 & ; net < 0 \end{cases}$$

يسمى بتابع الإشارة step function

هناك نوعان للبيانات إما البيانات القابلة للفصل خطيا أو البيانات الغير قابلة للفصل

لدينا نوعين من البوابات الشهيرة القابلة للفصل خطيا : OR AND

AND

X	Y	X AND Y
1	1	1
1	0	0
0	1	0
0	0	0

لماذا قابلة للفصل الخطي ؟ لنمثل هدا التابع بالمستوي X0Y بحيث نمثل النقاط التي خرجها (1) بدائرة سوداء والنقاط التي خرجها صفر بدائرة بيضاء

نلاحظ يوجد خط مستقيم يفصل الخرج الذي قيمته (1) عن قيم الخرج (0)

OR

X	Y	X OR Y
1	1	1
1	0	1
0	1	1
0	0	0

XOR

X	Y	X XOR Y
1	1	0
1	0	1
0	1	1
0	0	0

وبالتالي التابعان AND OR قابلان للفصل خطيا أي يوجد خط مستقيم يفصل بين الأصفار والواحدات

التابع XOR غير قابل للفصل خطيا لأننا احتجنا إلى خطين للفصل بين الأصفار والواحدات

هذا العصبون الصنعي ب n دخل وخرج وحيد يدعى : المدرك أو آلة الإدراك (Perceptron)

هذا الPerceptron له المقدرة فقط على تعلم البيانات القابلة للفصل خطيا

وقد تعلمنا كيف نحسب خرج هذا العصبون

ليكن لدينا نماذج التدريب التالية :

P1=[1,1] p2=[1,0] p3=[0,1] p4=[0,0] ولدينا الخرج المقابل على الترتيب

d1=1 d2=0 d3=0 d4=0 (الدخل مع الخرج يدعى أنماط التدريب)

عملية التعلم في الشبكات العصبونية الهدف منها دوما البحث عن قيم الأوزان التي تجعل الخرج المعطى من نماذج التدريب(d) يساوي الخرج المحسوب من العصبون(0) أي : ((D=0))

الحل : الخرج في المثال السابق يدل على بوابة And على سبيل المثال لو اخترنا الأوزان بالقيم التالية : w1=w2=1/2

$$f(net)=\begin{cases} +1 & ; net \geq 0.6 \\ 0 & ; net < 0.6 \end{cases}$$

((عدد المداخل في الPerceptron يحددها عدد المركبات في نموذج الدخل))

P1: $net_1= w_1x_1+w_2x_2= x_1/2+x_2/2= 1/2+ 1/2 =1 >0.6 \rightarrow O_1=f(net_1)=1$

$P_2: net_2= 1/2+ 0 =1 <0.6 \rightarrow O_2=f(net_2)=0$

$P_3: net_3= 0 +1/2=1 <0.6 \rightarrow O_3=f(net_3)=0$

$P_4: net_4= 0 +0=1 <0.6 \rightarrow O_4=f(net_4)=0$

ماهي الوظيفة الأساسية لهذا العصبون في الشبكات العصبونية ؟

يقوم بوظيفة هامة تدعى التصنيف ((classification))

ما معنى التصنيف : يقوم بتصنيف الواحدات عن الأصفار .

أو : يكتشف بأن مركبات الدخل نوعين من البيانات نوع خرجه (1) ونوع خرجه (0)

مثال : لنفترض لدي معمل خضراوات يتلقى نوعين من الفواكه : تفاح وبرتقال ولدي حساسات ((الحساس يأخذ مواصفات التفاح والبرتقال)) الحساسات تمثل الشبكة العصبونية التي تفرز التفاح في مكان والبرتقال في مكان

هدا الPerceptron البسيط يسمى linearclassific يصنف نوعين من البيانات

الخط المستقيم الفاصل بين نوعين من البيانات يدعى حد القرار أي كل البيانات فوق المستقيم تنتمي إلى صنف معين والبيانات التي تحت المستقيم تنتمي إلى صنف آخر

كيف نقوم بحساب معادلة هدا المستقيم ؟

البيانات فوق المستقيم ← الخرج (1)← $Net = w_1x_1+w_2x_2 \geq 0$

البيانات تحت المستقيم ← الخرج (0)← $Net = w_1x_1+w_2x_2 < 0$

البيانات على المستقيم ← $Net = w_1x_1+w_2x_2 = 0$

وهده المعادلة تعطينا معادلة حد القرار نوجد المعادلة من أجل بوابة AND

$w_1x_1+w_2x_2=$ العتبة ←$= x_1/2+x_2/2= 0.6 \rightarrow x_1+x_2= 1.2$

وهي معادلة المستقيم الفاصل بالنسبة لبوابة and الذي يفصل بين الواحدات والاصفار

معتدلة المستقيم الفاصل بالنسبة لبوابة or $x_1+x_2= 0.6$

تطبيق ليكن لدينا البيانات التالية :

$P_0=[-1,-1,-1]$ $P_1=[-1,-1,1]$ $P_2=[-1,1,-1]$ $P_3=[-1,1,1]$

$P_4=[1,-1,-1]$ $P_5=[1,-1,1]$ $P_6=[1,1,-1]$ $P_7=[1,1,1]$

هل يوجد perceptron قادر على تعلم هده البيانات ؟

أولا نرى إذا كانت هده البيانات قابلة للفصل خطيا فادا كانت كذلك فحتما يوجد perceptron

التعلم بإشراف أي هناك أمثلة وحلول ثم التعميم

الperceptron ينطوي تحت التعلم بإشراف أي نستخرج من البيانات دخل وخرج

الحل : هده البيانات تنتمي إلى صنفين إما مجموع بعض النقاط أكبر من الصفر أو أخرى أصغر من الصفر

$P_3\ P_5\ P_6\ P_7 > 0$; d=1 (class1)

$P_0\ P_2\ P_4\ P_1 < 0$; d=-1 (class2)

هل يوجد لهده البياناتlinear classific ؟

هل هده البيانات قابلة للفصل خطيا ؟

ليست قابلة للفصل خطيا ولكنها قابلة للفصل في مستوى

$$f(net)=\begin{cases} +1 & ; net \geq 0 \\ -1 & ; net < 0 \end{cases}$$

مداخل perceptronهي 3 مداخل لأنها 3 مركبات شعاعية وعصبون وحيد لأنه لدينا خرج وحيد $w_1=w_2=w_3=1$ و$y=0$ ومعادلة المستقيم له تساوي $x_1+x_2+x_3=0$

خوارزمية تدريب ال perceptron :

1) start with random weights

نقوم بتوليد مجموعة من الأوزان العشوائية ويفضل أن تكون محصورة في المجال [-1,+1]

2) select input patern

نقوم باختيار نمط دخل

3) calculate the perceptron output

حساب قيمة الخرج من العصبون , ودلك باستخدام maclloch –pitts ,
O=f(net)

4) if d=! O

ادا كان الخرج المعطى من نمادج التدريب لايساوي الخرج المحسوب من العصبون فنقوم بتعديل الأوزان :

$W_{new} = W_{old} + C(d - O) X$

X : شعاع الدخل الذي اخترناه في الخطوة (2)

5) go to (2)

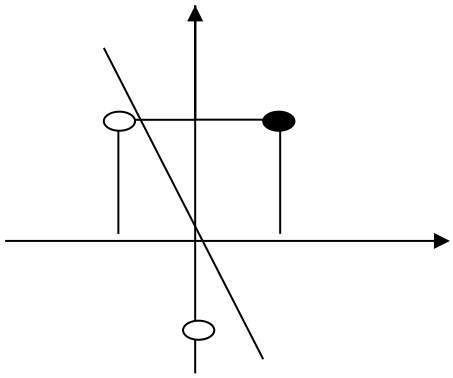
مثال : ليكن لدينا أنماط التدريب التالية :

$P_1 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$, $d_1=1$

$P_2 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$, $d_2=0$

$P_3 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$, $d_3=0$

والمطلوب : هل يوجد perceptron قادر على تعلم هده البيانات ؟



الحل :نتحقق ادا كان قابلة للفصل خطياً,

واضح من الشكل قابل للفصل خطياً أي يوجد

Perceptron قابل للتعلم على هده البيانات .

فما هو شكل هدا الperceptron ؟

له دخلين وخرج وحيد ((خرج واحد لأن العصبون الواحد قادر على الفصل بين صنفين))

نفرض : $w_1=1.0 \quad w_2= -0.8$

(تابع التفعيل هو تابع ثنائي لاقطبي منفصل لأننا نتعامل مع perceptron وهي من صفاته)

$$f(net)=\begin{cases} +1 & ; net \geq 0 \\ 0 & ; net < 0 \end{cases}$$

ملاحظة : أنماط التدريب تحدد شكل تابع التفعيل

عدد المداخل وعدد الأصناف تحدد بنية الشبكة العصبونية

الآن نبدأ بتطبيق الخوارزمية :

$$\text{For } p_1 : net_1 = \begin{pmatrix} 1 \\ -0.8 \end{pmatrix} [1 \quad 2] = -0.6$$

$\rightarrow O_1=f(net)=0$

$D_1 \neq O_1$

$$W_{new}=\begin{bmatrix}1\\-0.8\end{bmatrix}+(1)(1-0)\begin{bmatrix}1\\2\end{bmatrix}=\begin{bmatrix}2.0\\1.2\end{bmatrix}$$

For p_2 : $net_2 = 0.4$

$\rightarrow O_2 = f(net_2) = 1$, $d_2 = O_2$

$$\rightarrow W_{new}=\begin{bmatrix}2.0\\1.2\end{bmatrix}+(1)(0-1)\begin{bmatrix}-1\\2\end{bmatrix}=\begin{bmatrix}3.0\\-0.8\end{bmatrix}$$

For p_3 : $net_3 = 0.8$

$\rightarrow O_3 = f(net_3) = 1$, $d_3 = O_3$

$$\rightarrow W_{new}=\begin{bmatrix}3.0\\-0.8\end{bmatrix}+(1)(0-1)\begin{bmatrix}0\\-1\end{bmatrix}=\begin{bmatrix}3.0\\0.2\end{bmatrix}$$

والآن نكون قد أنهينا cycle1 في عملية التدريب ((عملية المرور على كل نماذج التدريب ولمرة واحدة تسمى cycle))

الآن سندخل في cycle ثاني ونكرر نفس الخطوات ولكن مع شعاع الوزن الجديد الأخير

(0.2 3.0) فنلاحظ أن : $d_3=O_3$ $d_2=O_2$ $d_1=O_1$ أي لايوجد تعديل أي الشبكة تدربت وتوقفنا عند هده cycle .

إدا واحد من بينهم يحتاج إلى تعديل أي ندخل في cycle أخرى وهكذا .....

ملاحظة : تكرر ال cycle حتى نصل الى حالة $d=O$ لكل نماذج التدريب .

ملاحظة : كل نمط تدريب نمرره يكون هناك مستقيم يتحرك في المستوي حتى يستقر في مكان معين تكون الشبكة تدربت .

معادلة المستقيم الفاصل بين البيانات : $3.0X_1+0.2X_2 = 0$

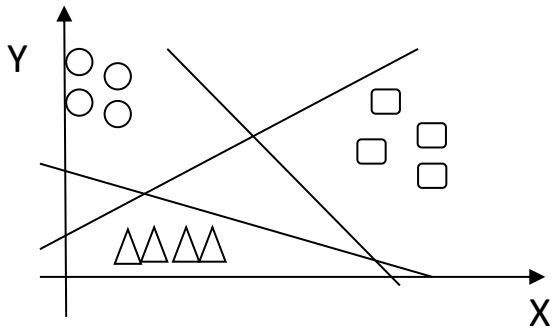
نظرية تقارب ال perceptron

خوارزمية تدريب ال **perceptron** تتقارب في عدد منته من الخطوات إذا وفقط إذا كانت البيانات قابلة للفصل خطيا

إذا كانت البيانات غير قابلة للفصل خطيا فالخوارزمية متباعدة ((موجود مستقيم الفصل ولكنه غير مستقر أي لن يتوقف))

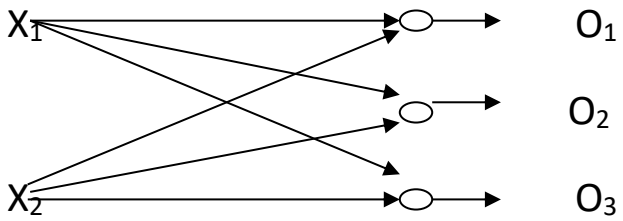
الشبكات العصبونية وحيدة الطبقة بخرج متعدد :

بفرض لدينا البيانات التالية :



والمطلوب ماهي معمارية الشبكة العصبونية القادرة على تصنيف هذه البيانات

المقصود بالمعمارية : هو عدد الطبقات في الشبكة العصبونية وعدد العصبونات في كل طبقة .



((في منطقة الدخل عصبونين حيث كل صنف له إحداثيات على x وy))

في منطقة الخرج ثلاث عصبونات لأنه لدي 3 أصناف كل عصبون يمثل صنف))

تسمى هده الشبكة :

Single layer يقصد بها طبقة وحيدة من الخرج .

Full connect كاملة الارتباط : كل عصبون مرتبط مع كل العصبونات في الطبقة التالية له .

Feed forwad nn أمامية التغذية : أي أن الخرج لايمكن أن يصبح دخل مرة أخرى .

ملاحظة : عند تدريب ال perceptron على أكثر من صنفين فإنه يتم فصل كل نوعين على حدى أي لازال شرط linearly separable محقق .

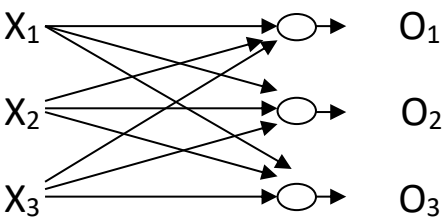
تمرين : ليكن لدينا الأصناف التالية :

$$Class1 : X_1 = \begin{pmatrix} 10 \\ 2 \\ -1 \end{pmatrix} \quad , d_1 = \begin{pmatrix} 1 \\ -1 \\ -1 \end{pmatrix}$$

$$Class2 : X_2 = \begin{pmatrix} 2 \\ -5 \\ -1 \end{pmatrix} \quad , d_2 = \begin{pmatrix} -1 \\ 1 \\ -1 \end{pmatrix}$$

$$Class3 : X_3 = \begin{pmatrix} -5 \\ 5 \\ -1 \end{pmatrix} \quad , d_3 = \begin{pmatrix} -1 \\ -1 \\ 1 \end{pmatrix}$$

ما هو الشكل المعياري لهذه الشبكة ؟



$$W = \begin{pmatrix} 1 & -2 & 0 \\ 0 & -1 & 2 \\ 1 & 3 & -1 \end{pmatrix}$$

((الأوزان معطاة بشكل عشوائي))

الأسطر تعبر عن الأوزان المتوجهة للعصبون ((مثلا السطر الأول موجه للعصبون الأول والسطر الثاني موجه للعصبون الثاني وهكذا))

الأعمدة تعبر عن الأوزان الخارجة من العصبون .

هل هذه البيانات قابلة للفصل خطيا مثنى مثنى أو هل يوجد Perceptron تدريب لهذه الشبكة.

الحل : يمكن إيجاد ثلاث مستقيمات تفصل كل نقطة عن النقاط المتبقية

$$f(net)=\begin{cases} +1 & ; net \geq 0 \\ 0 & ; net < 0 \end{cases} \qquad C= 1/2 ;$$

نباشر عميلة التدريب :

For X_1 : $O_1 = ([1 \quad -2 \quad 0] \begin{pmatrix} 1 \\ 0 \\ 2 \\ -1 \end{pmatrix}) = 1$

$O_2 = ([0 \quad -1 \quad 2] \begin{pmatrix} 1 \\ 0 \\ 2 \\ -1 \end{pmatrix}) = -1$

$O_3 = ([1 \quad 3 \quad -2] \begin{pmatrix} 1 \\ 0 \\ 2 \\ -1 \end{pmatrix}) = 1$

نلاحظ أن الخرج المحسوب الأول يساوي الخرج المعطى والخرج المحسوب الثاني يساوي الخرج المعطى ولكن الخرج الثالث المحسوب لا يساوي الخرج المعطى لذا نعدل أوزان السطر الثالث .

$$W_3=\begin{pmatrix}1\\3\\-1\end{pmatrix}\begin{pmatrix}10\\2\\-1\end{pmatrix}=\begin{pmatrix}-9\\1\\0\end{pmatrix} \quad \rightarrow \quad W=\begin{pmatrix}1 & -2 & 0\\0 & -1 & 2\\-9 & 1 & 0\end{pmatrix}$$

For X_2 : $O_1 = \left([1 \quad -2 \quad 0]\begin{pmatrix}2\\-5\\-1\end{pmatrix}\right) = 1$

$$O_1 = \left([0 \quad -1 \quad 2]\begin{pmatrix}2\\-5\\-1\end{pmatrix}\right) = 1$$

$$O_1 = \left([-9 \quad 1 \quad 0]\begin{pmatrix}2\\-5\\-1\end{pmatrix}\right) = -1$$

لذلك نعدل أوزان العصبون الأول :

$$W_3=\begin{pmatrix}1\\-2\\0\end{pmatrix}-\begin{pmatrix}2\\-5\\-1\end{pmatrix}=\begin{pmatrix}-1\\3\\1\end{pmatrix} \quad \rightarrow \quad W=\begin{pmatrix}-1 & 3 & 5\\0 & -1 & 2\\-9 & 1 & 0\end{pmatrix}$$

For X_3 : $O_1 = 1$, $O_2 = -1$, $O_3 = 1$

$$W_1=\begin{pmatrix}-1\\3\\1\end{pmatrix}-\begin{pmatrix}-5\\5\\-1\end{pmatrix}=\begin{pmatrix}4\\-2\\2\end{pmatrix} \quad \rightarrow \quad W=\begin{pmatrix}4 & -2 & 2\\0 & -1 & 2\\-9 & 1 & 0\end{pmatrix}$$

ندخل في CYCLE , هده الشبكة بعد 8 دورات تتوقف لتصبح الأوزان النهائية :

$$\rightarrow \quad W = \begin{pmatrix} 5 & 3 & 5 \\ 0 & -1 & 2 \\ -9 & 1 & 0 \end{pmatrix}$$

نسمي هذا النوع من الذكاء بالذكاء الحسابي لأنه كله عبارة عن حسابات

يوجد ثلاث مستقيمات تدور في المستوي

معادلات الفصل :

$5X_1+3X_2-5 = 0$

$- X_2-2 = 0$

$-9X_1+X_2 = 0$

Delta learning Rule :

تذكر عن طريق المقدار {r=d-o } استطعنا الوصول لقاعدة تعلم ال perceptron حيث

{d-o} يعبر عن الخطأ وكلما كان هناك خطأ وجب علينا إجراء المزيد من عمليات التغيير .

وفي قاعدة تعلم ال delta لدينا :

قاعدة تعلم delta $R= (d_i - o_i).f`(net_i)$

اشارة التعلم target actual مشتق تابع التفعيل

طالما أن هذا المقدار فيه جداء مع مشتق تابع التفعيل فإن قاعدة delta تصلح فقط للتعامل مع توابع التفعيل المستمرة ((القابلة للاشتقاق)) .

سميت هذه القاعدة بـdelta لأنه يمكن كتابتها كما في بعض المراجع بالشكل :

$R=\delta$; $\delta= (d_i - o_i).f '(net_i)$

كيف سأعبر عن تابع التفعيل الذي يحوي مشتق حاسوبيا ؟؟

1) لو أن لدي تابع التفعيل y= 1 ; $O= F(net)= \frac{1}{1+e^{-net}}$ هذا التابع يأخذ قيم بين (0,1)

مشتقه : $F ' (net)= \frac{e^{-net}}{(1+e^{-net})}$

بتفريق الكسر وإضافة (1)وطرح (1) يصبح لدينا :

$F ' (net)= f(net). (1-f(net))\rightarrow f ' (net)= O.(1-O)$; $O=f(net).$

تمكنا من إيجاد مشتق التابع بدلالة بنفسه , وبذلك أخفض من عدد الحسابات

((حيث نحسب O لمرة واحدة ونستخدمها في أكثر من مكان))

ملاحظة : كل تابع تفعيل له حيلة

إذا وكقاعدة : إذا كان تابع التفعيل هو $F(net)= \frac{1}{1+e^{-net}}$ فإن مشتقه دوماً

$f ' (net)= O.(1-O)$

وبهذا فإن معدل التغير في الوزن :

$\Delta W_i=crx\rightarrow \Delta w_i=c(d_i-O_i)O_i(1-O_i)X$

ومنه: $W_i(t+1)=Wi(t)+ c(d_i-O_i)O_i(1-O_i)X$

لو أن لدي تابع التفعيل :

$O= F(net)= \frac{2}{1+e^{-net}} -1$

سنلاحظ أن

$f ' (net)= 1/2(1-O^2)$

$W_i(t+1)=Wi(t)+ c/2(d_i-O_i).(1-O_i^2)X$

سؤال كيف أحدد تابع التفعيل (القاعدة)الذي يجب استخدامها ؟

الجواب حسب الخرج , حيث إذا كان الخرج محصور بين (0,1) أختار الأول إذا كان الخرج محصور بين (-1,1) أختار الثاني , فطبيعة البيانات هي التي تحدد تابع التفعيل المستخدم .

مثال لتكن لدي البيانات التالية :

X= [1 -2 0 -1] , D=1

X =[0 1.5 -0.5 -1] , D=-1

X= [-1 1 0.5 -1] , D=1

W=[1 -1 0 0.5] , C=0.1

الحل : سنختار القاعدة الثانية لأن d قيمها بين (-1,1), $f'(net)= 1/2(1-O^2)$

من أجل الحل سنتبع نفس خطوات خوارزمية ال perceptron مع اختلاف وحيد هو طريقة حساب الأوزان الجديدة , أذا نختار تابع التفعيل :

$O= F(net)= \frac{2}{1+e^{-net}} - 1$

For X_1: net = W^TX_1= 2.5

$\rightarrow O_1= F(net)= 0.848 = d_1$

$f'(net)= 1/2(1-O^2)= 0.140$

$W_{new}=W_{old}+ C/2(d-O)(1-O^2)X$

=[1 -1 0 0.5]+ 0.1(-1-0.848)(0.140)[1 -2 0 -1]

=[0.974 0.978 0 0.52]

For X_2: net = $W^T X_2$= -1.948

$\rightarrow O_2$= F(net)=-0.759 = d_2

f ' (net)= 1/2(1-O^2)= 0.218

W_{new}= [0.974 -0.956 0.002 0.531]

For X_3: net = ---------- نكرر نفس الخطوات

السؤال متى نتوقف عن التدريب ؟

الجواب : نتوقف عن التدريب عندما يكون الفرق (d-O) لكل نماذج التدريب قريب من الصفر d-O≈0 أي دوما أقارن هذا المقدار مع الخطأ معطى وليكن E= $\frac{1}{1000}$ فإذا كان من رتبته أتوقف عن عملية التدريب

إذا الفرق بين ال Perceptron و الdelta :

Perceptron	Delta
تستخدم توابع منفصلة	تستخدم توابع مستمرة
نستمر بعملية التدريب حتى d=O	نستمر بعملية التدريب حتىd-O≈0
من أجل كل نماذج التدريب	من أجل كل نماذج التدريب

- قاعدة ال delta لا تقوم بعملية الفصل 100% حيث يأتي الخطأ من أن بعض الأصناف متداخلة مع بعضها البعض أي نصنف قدر الإمكان ولكن بخطأ معطى .
- لا يزال شرط الفصل قائما حيث الdelta قادرة على تعلم البيانات القابلة للفصل الخطي ولكن بنسبة خطأ معطى بسبب وجود تداخل في البيانات .

Hebbian learning Rule:

إشارة التعلم لهذه القاعدة :

$r=f(net_i)=O_i$

تستطيع استخدام كلا النوعين من التوابع (المستمرة والمنفصلة).

نلاحظ من إشارة التعلم أنها مباشرة تساوي الخرج الحقيقي للعصبون أي أن هدا التعلم بدون إشراف , لأنه دوما في التعلم بإشراف يكون معطى (d) و ال(O)محسوبة

تم تطوير قاعدة التعلم هذه لاستخدامها لاسترداد الذاكرة (التذكر)

((لو أن لدي صورة تم مسح جزء منها تحاول استرداد الصورة الأصلية))

$\Delta W_i(t)=CO_iX \rightarrow W_i(t+1)= W_i(t) +CO_iX$

مثال :

X_1 =[1 -2 1.5 0]

X_2 =[1 -0.5 -2 -1.5]

X_3 =[0 1 -1 -1.5]

W=[1 -1 0 0.5] , C=1, Function activation = sgn

الحل :

For X_1: net = $W^T X_1$= 3

$W_{new}=W_{old}+ C\ Sgn(net_1)X_1 = W_{old} + X_1$; Sgn (net) = 1

= [1 -1 0 0.5] + [1 -2 1.5 0]= [2 -3 1.5 0.5]

For X_2: net = $W^T X_2$= -0.25

$W_{new}=W_{old}+ C\ Sgn(net_2)X_2 = W_{old} - X_2$ = [1 -2.5 3.5 2]

For X_3: net ---------

W_{new} = [1 -3.5 4.5 0.5]

شرط التوقف : يتم التوقف عن التدريب عند استرجاع جميع النماذج المخزنة .

Widrow – Hoff – learning Rule :

إشارة التعلم لهذه القاعدة $r=d_i - net_i \rightarrow f(net)=net$

من العلاقة نلاحظ أن التعلم هنا بإشراف ((حيث d_i موجودة في القاعدة))

سؤال : ماهي العلاقة بين قاعدة ال delta وقاعدة الwidrow – hoff ؟

الجواب : في قاعدة تعلم ال widrow –Hoff :

f(net)=net , f ' (net)=1

أي تعتبر قاعدة widrow –Hoff حالة خاصة من قاعدة تعلم ال delta بشرط تابع التفعيل

f(net)=net ومشتقه f ' (net)=1

إذا $W_i(t+1)= W_i(t) +(d_i –net_i)X$

تلميح :

$\Delta W_i=crx\rightarrow\Delta w_i=c(d_i-O_i)f\ '\ (net_i)X$

لكن :

$O_i = f(net_i)= net_i\ ,\ f\ '\ (net_i)=1$

$\Delta w_i=c(d_i- net_i)X$

تلخيص لقواعد التعلم :

Learning Rule	$\Delta W_i(t)$	Learning Type
Hebbian	CO_iX	Unsupervised
Perceptron	$C(d_i - O_i)X$	Supervised
Delta	$C(d_i - O_i)\ .\ f\ '\ (net)X$	Supervised
Widrow - Hoff	$C(d_i - net_i)X$	Supervised

النظم الخبيرة

تعتبر النظم الخبيرة من أكثر تطبيقات تقانات المحاكمة في الذكاء الصنعي . والتي تستخدم الحقائق و القواعد لتضمين المعارف حول حقل معين من حقول المعارف البشرية مثل الطب والهندسة والأعمال .

تصمم النظم الخبيرة عموما لحل مسائل التصتيف واتخاد القرارات مثل (التشخيص الطبي , الوصفات العلاجية , تنظيم البورصات وغيرها ...)

النظم الخبيرة هي أدوات دكاء صنعي . وهدا يعني أننا لا نستعملها إلا في المسائل التي ليس لها أي خوارزمية واضحة أكيدة لحلها .

تتطلب النظم الخبيرة وجود خبرة نود نمدجتها أي أنه لامعنى للنظم الخبيرة إلا في المجالات التي توجد فيها خبرة بشرية . والخبير هو الشخص الدي يعرف مجال التطبيق . ويعرف نوعا ما , كيف ينقل معرفته للآخرين .

التفكير البشري معقد جدا ومن الصعب تمثيله بخوارزمية . ومع دلك فإن معظم الخبراء قادرين على وضع معارفهم لحل المسائل على شكل قواعد من الشكل (إلا أنهم غير قادرين على ضمان كمال هده القواعد ولا تكاملها معا).

IF<antecedent> مقدمة THEN<consequent>النتيجة

مثال تعليمي :

يمكن أن نتصور مثلا موظف في بنك مسؤول عن إعطاء القروض للمتعاملين . يستخدم نظام خبير يساعده في تقرير فيما إدا كان من المناسب إعطاء قرض لمتعامل . لنفرض أن الدرات التالية تستخدم المعاني الموافقة :

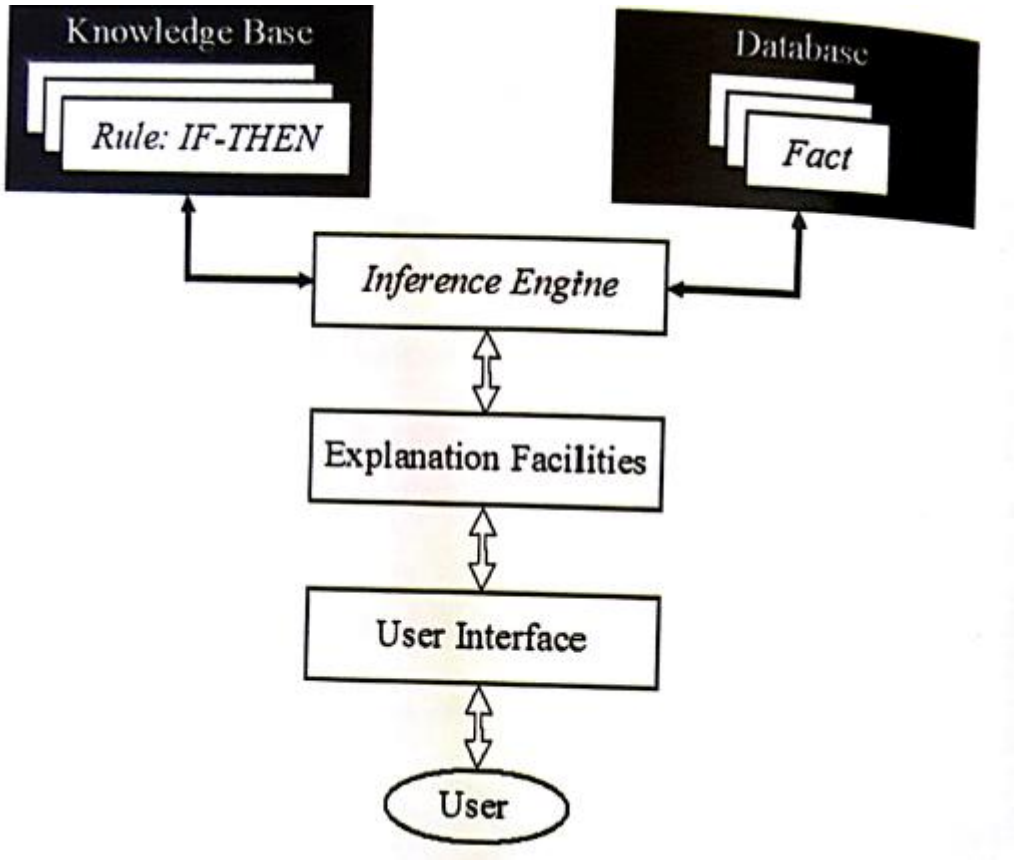
يجب أن يتم الموافقة على القرض	OK
ضمانة القرض المقنعة	COLLAT
العميل قادر على تسديد دفعات القرض	PYMT
للعميل سمعة مالية جيدة	REP
تخمين الضمانة أكبر من مبلغ القرض	APP
للعميل دفعات دورية منتظمة	RATING
يتجاوز دخل العميل مصاريفه	INC
للعميل صفحة متوازية ممتازة	BAL

يمكن استخدام القواعد التالية بهدف إقرار القرض :

1. COLLAT ^ PYMT ^ REP → OK
2. APP→ COLLAT
3. RATING → REP
4. INC → PYMT
5. BAL ^ REP → OK

البنية العاملة للنظام الخبير :

يبين الشكل التالي البنية العامة للنظام الخبير :



قاعدة المعرفة Knowledge Base

تحوي قاعدة العرفة معارف المجال والمفيدة في حل المسائل . تمثل المعارف على شكل قواعد استنتاج .

IF<antecedent> مقدمة THEN<consequent>النتيجة

عندما يكون الشرط محققا يتم تطبيق القاعدة وتنفيذ الفعل الموافق .

قاعدة البيانات Data Base

تحوي قاعدة البيانات مجموعة من الحقائق Facts والتي تستخدم كدخل لشروط القواعد المخزنة في قاعدة المعرفة .

محرك الاستدلال Inference engine

يقوم محرك الاستدلال بالمحاكمة اللازمة للوصول الى الحل . عبر تطبيق قواعد المعرفة على حقائق قاعدة البيانات لاستنتاج حقائق جديدة .

نظام الشرح Explanation System

يتنصت نظام الشرح على محرك الاستدلال ليعطي للمستخدم تفسيرا لكيفية وصوله للحقائق الجديدة انطلاقا من حقائق قاعدة البيانات

واجهة الاستخدام User Interface

وهو وسيلة التخاطب بين المستخدم والخبير .

استراتيجيات محرك الاستدلال :

تمثل المعرفة في النظم الخبيرة باستخدام القواعد Rules . وتوضع البيانات على شكل حقائق Facts .

يقوم محرك الاستدلال بمقارنة كل قاعدة موجودة في قاعدة المعرفة مع الحقائق الموجودة في قاعدة البيانات .وعندما يجد المحرك ذأن شرط قاعدة محقق يقوم بتنفيد القاعدة .

تولد عمليات المطابقة والتنفيذ سلسة معينة ندعوها سلسة الاستدلال

chains Inference .

تحدد سلسلة الاستدلال كيفية تطبيق القواعد للوصول إلى نتيجة .

السلسة الأمامية forward Chaining

تعتمد هده الاستراتيجية على فكرة التالية : ((ولد مايمكن توليده حتى الوصول الى نتيجة المطلوبة))

يقوم محرك الاستدلال في هده الإستراتيجية وفي كل دورة cycle بمسح القواعد لتحديد القواعد القابلة للتطبيق مع قاعدة الحقائق

يطبق محرك الاستدلال هده القواعد اعتبارا من القاعدة الاعلى مما سيضيف حقائق جديدة الى قاعدة الحقائق

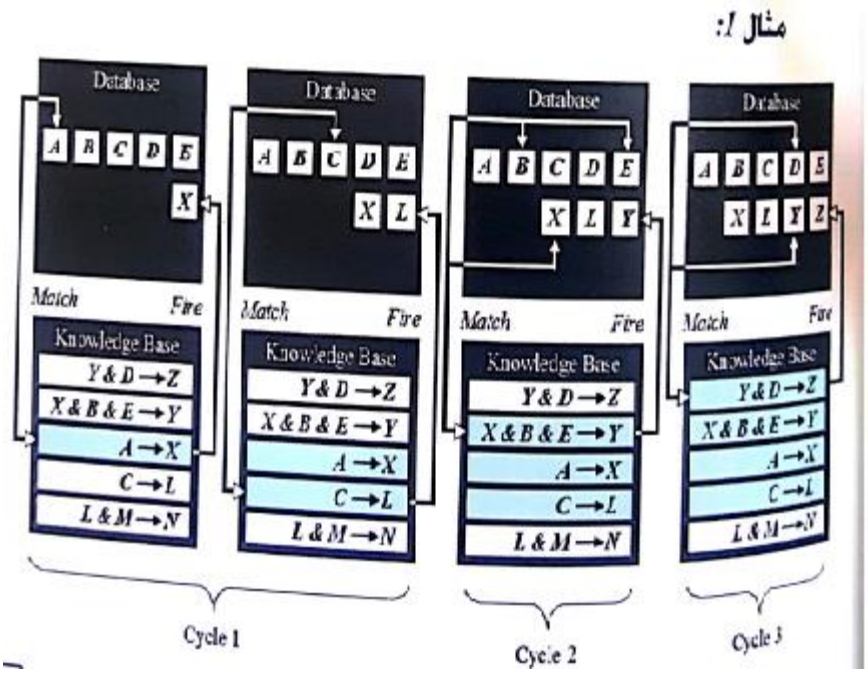
ثم ينقل الى الدورة التالية وهكدا حتى توليد النتيجة المطلوبة

تولد السلسلة الأمامية كل ما يمكن توليده

يمكن في السلسلة الأمامية تطبيق قواعد لا علاقة لها بالمسألة المطلوبة

يمكن في بعض الأحيان وحين يكون الهدف استنتاج حقيقة معينة ألا تكون هده الإستراتيجية فعالة .

مثال 1:



مثال 2 :

ليكن لدينا قاعدة الحقائق {B,C} والقواعد :

1) B^ D ^ E ⟶ F
2) G ^ D ⟶ A
3) C ^ F ⟶ A
4) B ⟶ X
5) D⟶ E
6) X ^ A ⟶ H
7) C⟶ D
8) X ^ C ⟶ A
9) X ^ B ⟶ D

المطلوب استنتاج الحقيقة H

في الدورة الأولى : القاعدة 4 والقاعدة 7 قابلتين للتطبيق .

تضيف القاعدة 4 الحقيقة X الى قاعدة الحقائق . وتضيف القاعدة 7 الحقيقة D الى قاعدة الحقائق .

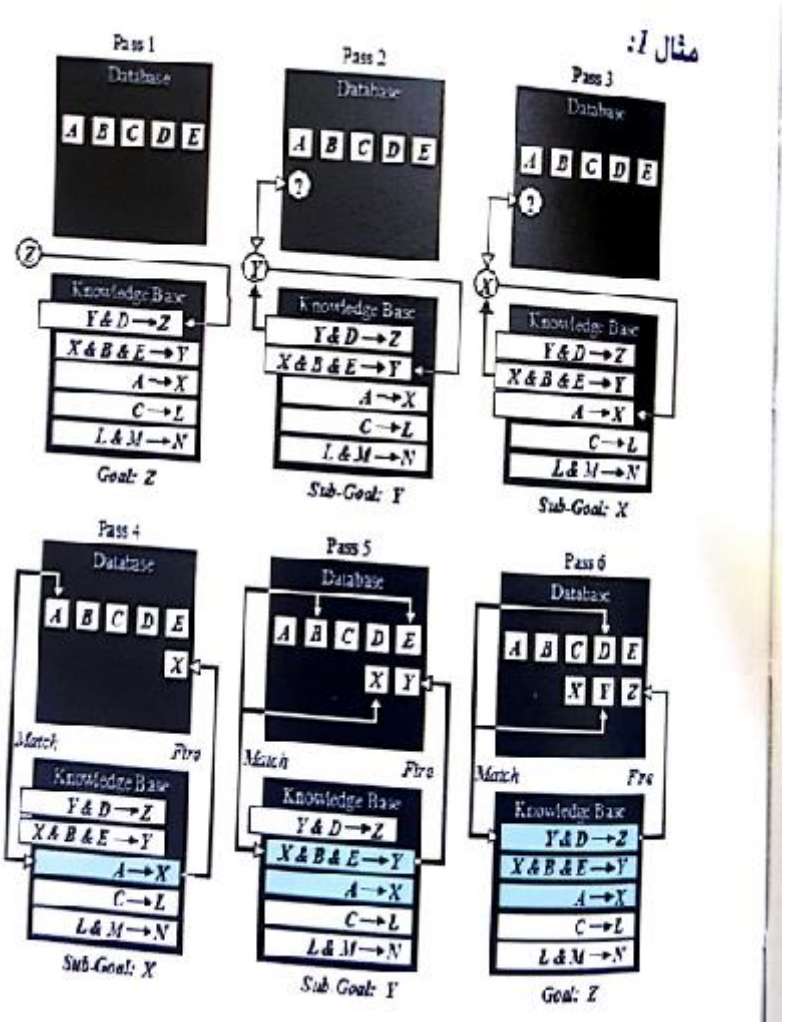
في الدورة الثانية :

القاعدة 8 القاعدة 5 قابلتين للتطبيق . تضيف القاعدة 8 الحقيقة A الى قاعدة الحقائق .وتضيف القاعدة 5 الحقيقة E الى قاعدة الحقائق .

في الدورة الثالثة : القاعدة I الحقيقة F الى قاعدة الحقائق . وتضيف القاعدة 7 الحقيقة H الى قاعدة الحقائق .
وبدلك تتحقق H وتتوقف عملية البحث .

السلسة الخلفية Backward Chaining :

تعتمد هده الإستراتيجية على الفكرة التالية : " ابدأ من الهدف وحاول برهان كل مايلزم لتحقيقه " .
تعتمد آلية السلسلة الخلفية على البدء من الحقيقة الهدف . والبحث في مجموعة القواعد عن القواعد التي تقع هده الحقيقة في نتيجتها . ومن ثم إنشاء قائمة بالحقائق الواجب برهانها لنتمكن من تطبيق القواعد السابقة ثم نعاود تطبيق هده الآلية عوديا على الحقائق الموجودة في هده القوائم , وهكدا .



مثال 2 :

ليكن لدينا مجموعة الحقائق { B , C } والقواعد :

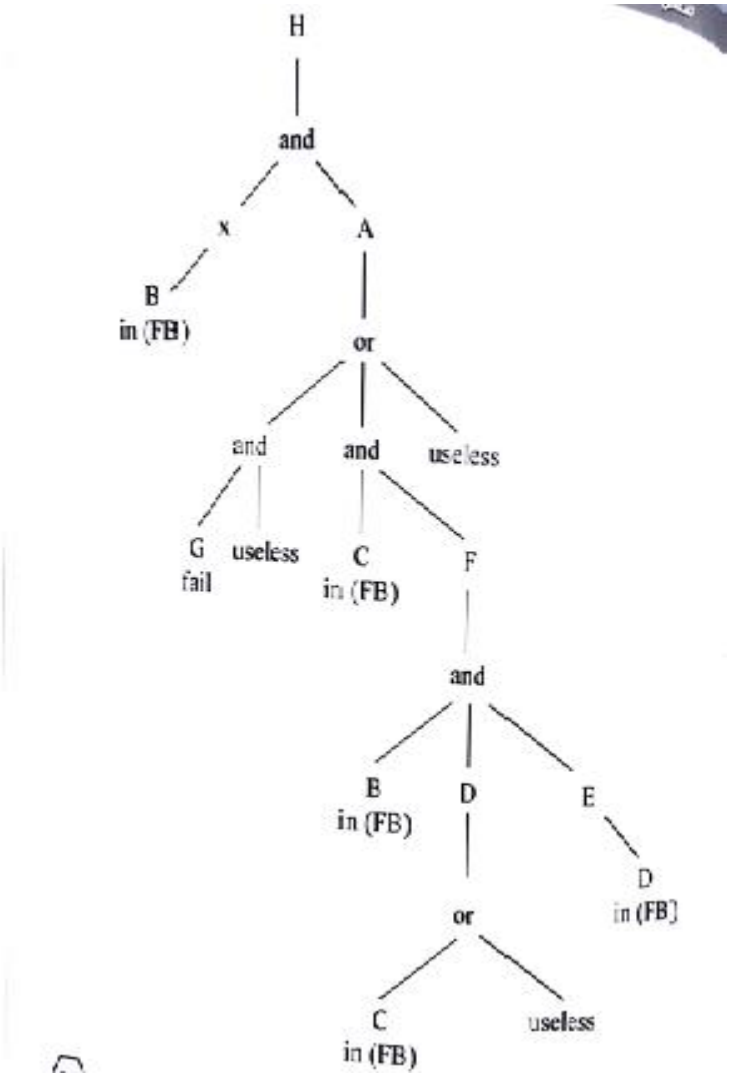
1) B ^ D^ E⟶ F
2) G ^ D ⟶ A
3) C ^ F ⟶ A
4) B ⟶ X
5) D ⟶ E
6) X ^ A ⟶ H
7) C ⟶ D

8) X ^ C → A

9) X ^ B → D

المطلوب استنتاج الحقيقة H .

يمكن توصيف تنفيذ خوارزمية السلسلة الخلفية بشجرة عقدها إما حقائق إما



حل التضارب

ليكن لدينا القواعد التالية في قاعدة المعرفة :

Rule 1 :

IF the ' traffic light ' is green THEN the action is go

Rule 2 :

IF the ' traffic light ' is red THEN the action is stop

Rule 3 :

IF the ' traffic light ' is red THEN the action is go

نلاحظ أن لكلا القاعدتين 2 و 3 نفس الشرط . وبالتالي فيمكن تطبيق كل منهما في حال تحقق الشرط .
تؤلف هده القواعد مجموعة تضارب . وعلى محرك الاستدلال أن يقرر أي من القواعد يجب تطبيقه من هده المجوعة .
تدعى الطريقة التي تقود الى اختبار القاعدة الواجب تطبيقها من مجموعة التضارب بحل التضارب
ستقوم إستراتيجية السلسلة الأمامية بتطبيق كلتا القاعدتين 2 و 3
تطبق القاعدة 2 أولا لأن ترتيبها قيل القاعدة 3 وبالتالي سيتم توليد النتيجة stop كذلك فإن القاعدة 3 ستطبق لأنها شرطها محقق وسيتم توليد النتيجة go .

طرق حل التضارب

طبق القاعدة ذات الأولوية الأكبر highest priority
يمكن في تطبيق بسيط إعطاء الأولويات للقواعد عن طريق ترتيبها في قاعدة المعرفة
(حيث ستطبق القواعد حسب ترتيبها)
طبق القاعدة الخاصة أكثر most specific rule
تدعى هده الطريقة أيضا بإستراتيجية المطابقة الأكبر longest strategy matching وتعتمد على فرضية أن القاعدة الأكثر خصوصية تعالج معلومات أكثر من القواعد العامة .
طبق قاعدة التي تستخدم البيانات الأحدث most recently entered data
تقوم هده الطريقة باضافة معرف الزمن الى كل حقيقة في قاعدة البيانات .
في حال التضارب نطبق القاعدة التي نعتمد شروطها على الحقائق الأحدث .
تعلم القواعد Rule Learning
يوجد العديد من الطرق المقترحة لتعلم القواعد بشكل استنتاجي .
سنشرح واحدة منها فيما يلي .
ولتوضيح اجرائية التعلم . سنستخدم ثانية مثال الموافقة على قرض مصرفي . إلا أنه عوضا عن البدء باعطاء القواعد لهده المسألة .
سنفرض أننا أعطينا مجموعة أمثلة للتعلم تتألف من قيم الحقائق لعدد من العملاء .

لنعتبر مثلا مجموعة التعلم المعطاة في الجدول التالي
(نستخدم ال 1 True و ال 0 Fales)

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

يمكن الحصول على هدا الجدول بالرجوع الى سجلات طلبات القروض المصرفية والقرارات التي اتخذها موظفو المصرف بشأن الموافقة على القرض . ندعو الأمثلة التي تأخذ فيها OK القيمة True بأمثلة موجبة . والتي تأخذ فيها القيمة Fales بأمثلة سالبة . ونريد باستخدام مجموعة التعلم السابقة استنتاج قواعد من الشكل :

$$a_1 \wedge a_2 \wedge \ldots\ldots a_n \rightarrow ok$$

ادا كان لمقدمة قاعدة القيمة True من أجل مثال معين من مجموعة التعلم . فإننا نقول بأن القاعدة تغطي cover المثال . يمكن أن نغير في أي قاعدة لجعلها تغطي عددا أقل من الأمثلة ودلك بأضافة درة لمقدمتها .
سيجعل مثل هدا التعبير القاعدة أكثر خصوصية specific .
يمكن لقاعدتين أن تغطيا أمثلة أكثر مما تغطيه قاعدة واحدة . يجعل اضافة قاعدة النظام الذي يستخدم هده القواعد أكثر عمومية General .
سنبحث عن مجموعة من القواعد التي تغطي كل وفقط الأمثلة الموجبة في مجموعة التعلم .