

Conception et implémentation

IFT-17586 Intelligence artificielle I – E2003



Séance 7

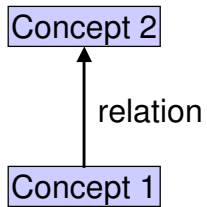
- ◆ Implémentation des modes de représentation des connaissances en Prolog

Plan



- ◆ Réseaux sémantiques
 - Relation, Réseau auxiliaire, Falsum
 - Preuve directe, Preuve par réfutation
- ◆ Logique des prédicats
 - Forme clausale, Clause de Horn
- ◆ Graphes conceptuels
 - graphe/1, concept/5, relation/5
- ◆ Exercices

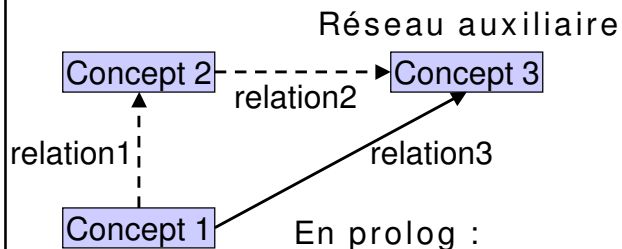
Réseaux sémantiques



En prolog :
`relation(concept1, concept2).`

Un réseau sémantique \Rightarrow un ensemble de faits
= la base de connaissances

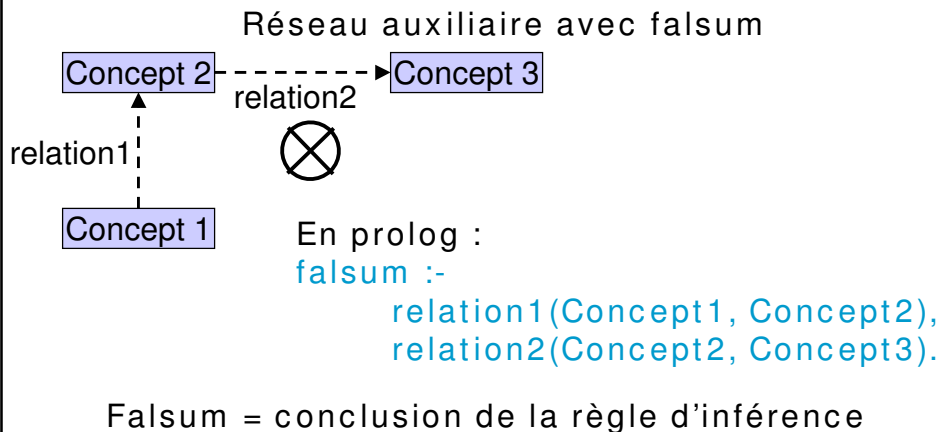
Réseaux sémantiques



En prolog :
`relation3(Concept1, Concept3) :-
relation1(Concept1, Concept2),
relation2(Concept2, Concept3).`

Un réseau auxiliaire \Rightarrow une règle d'inférence

Réseaux sémantiques



© Capus, Potvin et Tourigny, 2002

5

Réseaux sémantiques : preuve directe

- ◆ Construire le réseau sémantique
- ◆ Représenter les réseaux auxiliaires
- ◆ Poser la preuve sous forme de question :
Prouver que C1 est en relation r avec C2
`?- r(C1, C2).`
La réponse donne la solution de la preuve

© Capus, Potvin et Tourigny, 2002

6

Réseaux sémantiques : preuve par réfutation

- ◆ Construire le réseau sémantique
- ◆ Représenter les réseaux auxiliaires
- ◆ Ajouter le fait à réfuter
- ◆ Poser la question
?- falsum(...).
Si la réponse est positive alors le fait ajouté
doit être réfuté (car falsum est apparu dans
le réseau)

Logique des prédicats

- ◆ Transformer les expressions de la logique des prédicats en logique clausale
- ◆ Implémenter les clauses de Horn positives en Prolog
 - clause de Horn → au plus 1 littéral positif
 - clause de Horn positive → exactement 1 littéral positif

Logique des prédicats

- ◆ 1 littéral positif et aucun négatif
⇒ un fait

Logique clausale : `parent(paul, pierre)`

Prolog : `parent(paul, pierre).`

- ◆ 1 littéral positif et 1 ou plusieurs négatifs ⇒ une règle d'inférence

Logique clausale : $\neg p(x,y) \vee \neg r(y,x) \vee q(x,y)$

Prolog : `q(X, Y) :- p(X, Y), r(Y, X).`

Les graphes conceptuels

- ◆ Utiliser les prédicats :

- `graphe/1` pour identifier le graphe
`graphe(Identifiant_Graphe)`

- `concept/5`

`concept(Identifiant_Graphe, Identifiant_Concept,
Type_Concept, Référent_Concept,
Lien_CoRéférence)`

- `relation/5`

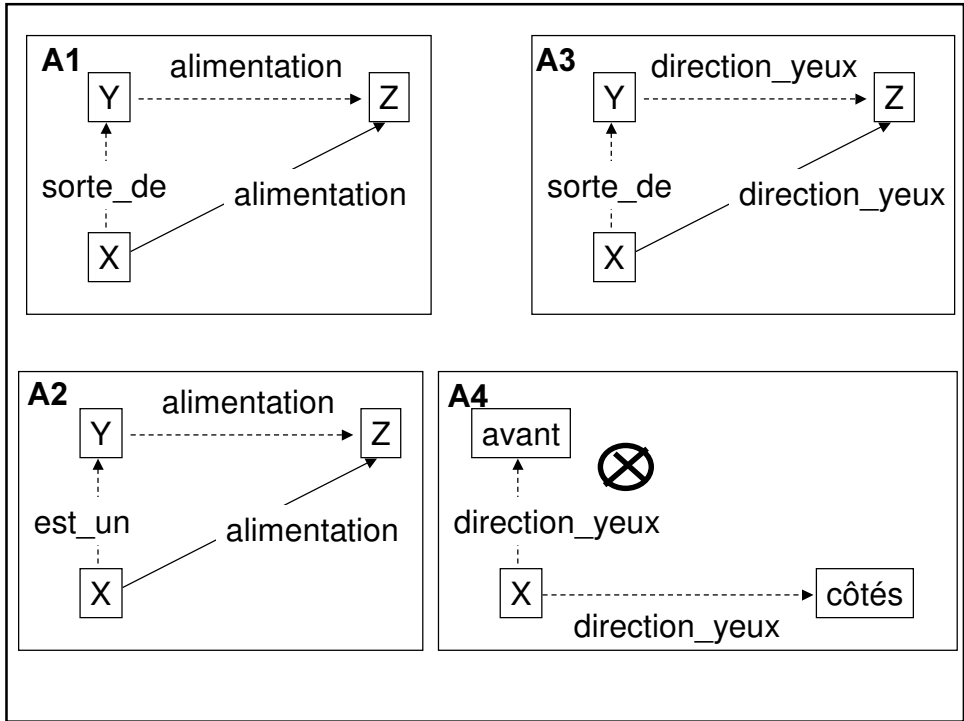
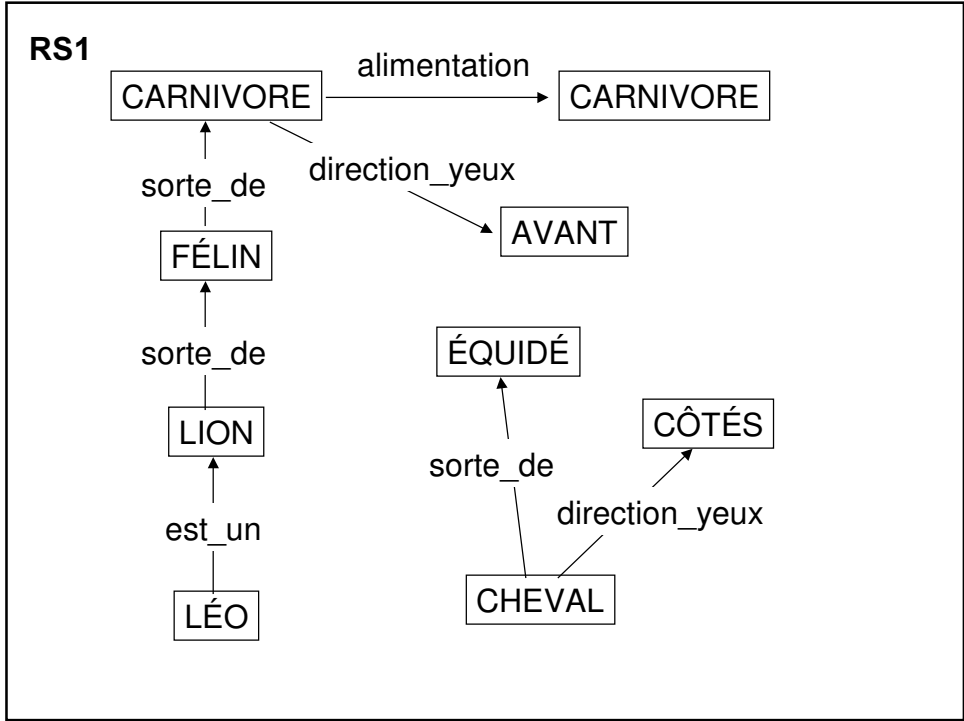
`relation(Identifiant_Graphe, Identifiant_Relation,
Type_Relation, Concepts_Entrants,
Concepts_Sortants)`

Les graphes conceptuels

- ◆ Les opérations de copie, jointure, restriction et simplification doivent être implémentées
- ◆ Autres développements en fonction des besoins des applications

Exercice 1

- ◆ Implémenter le réseau sémantique (RS1) et les réseaux auxiliaires (A1, A2, A3 et A4) en Prolog
- ◆ Comment prouver :
 - par preuve directe que :
Léo mange de la viande
 - par réfutation qu' :
Un équidé n'est pas un félin



Exercice 2

- ◆ Transformer en logique des prédicats, en logique clausale et en Prolog les énoncés suivants :
 - Nathalie possède les clés du placard
 - Si une personne souffre d'embonpoint ou aime cuisiner, elle a des chances d'être gourmande

Exercice 3

- ◆ Représenter en Prolog le graphe conceptuel suivant :

