DIC9305 Logique, informatique et sciences cognitives

Modélisation, inférence, base de connaissances I

Roger Villemaire

Département d'informatique
UQAM

le 14 mars 2024

UQÀM
Université du Québec à Montréal

Plan

Plan

Monde ouvert

- Les logiques classiques présupposent l'*hypothèse d'un monde ouvert* :
  - quelque chose qui n'est pas une conséquence n'est pas nécessairement faux !
    - Exemple : De $Prof(jean)$, je ne peux rien déduire au sujet de $Prof(paul)$.
  - Ceci est cohérent avec la sémantique vue.
    - Il y a un modèle de $Prof(jean)$ et $Prof(paul)$, c.-à-d. une structure qui satisfait ces formules.
    - Il y a un modèle de $Prof(jean)$ et $\neg Prof(paul)$.
  - Donc $\{Prof(jean)\} \not\models Prof(paul)$, et $Prof(paul)$ n'est pas une conséquence de $\mathcal{E} = \{Prof(jean)\}$.

Monde ouvert (suite)

- Quelque chose est donc vraie, si c'est satisfait dans *tous* les modèles.
- Ceci est cohérent avec la *monotonicité* de la logique classique :
  - de nouvelles connaissances (j'ajoute aux axiomes $\mathcal{E}$) peuvent me permettre de trancher de nouvelles questions,
  - mais toutes les anciennes conséquences sont maintenues.

Monde fermé

- Dans les bases de données et en programmation logique on présuppose l'*hypothèse d'un monde fermé*
  - Quelque chose qui ne peut pas être trouvé/déduit est nécessairement faux.
    - Exemple : je n'ai que *Prof*(*jean*) dans ma base de données (ou mon programme Prolog), donc *Prof*(*paul*) est faux (échec en Prolog).
  - Ceci est cohérent avec une sémantique basée sur une procédure de recherche/d'inférence.
    - Je peux inférer *Prof*(*paul*), c'est donc vrai.
    - Je ne peux pas inférer *Prof*(*paul*), c'est donc faux.

Monde fermé (suite)

- Mes connaissances sont complètes, il n'y a rien d'autre à savoir ! (à l'instant présent).
- Ceci est *non-monotone* :
  - de nouvelles connaissances peuvent faire disparaître une conséquence qu'on avait antérieurement.

Monde ouvert et Web Sémantique

- L'hypothèse d'un monde ouvert est toute naturelle dans le monde du Web Sémantique et du partage de l'information :
  - j'ai de l'information sur certaines personnes, mais pas sur tous les habitants de la terre !
- On prend donc comme acquis qu'on a une information partielle, qui pourra être complétée.
  - Tout à coup on rend disponible des données démographiques précises et récentes sur la Chine, je peux donc compléter mes connaissances de l'humanité !

Monde ouvert et logique

- L'hypothèse d'un monde ouvert est le point de vue traditionnel en logique car on déduit en prenant des axiomes comme prémisses :
  - et en pratique on va rarement avoir un axiome qui délimite exhaustivement les entités pouvant exister.
  - On va plutôt donner des considérations générales, applicables à un nombre non spécifié d'entités.
    - Tous les hommes sont mortels, Socrate est un homme, Socrate est donc mortel.

Monde fermé et Bases de Données

- L'hypothèse d'un monde fermé est toute naturelle dans le monde des bases de données :
  - Paul n'est pas dans la bases de données des employés, ce n'est donc pas un employé !
- On prend donc comme acquis qu'on a une information complète.
  - Ma BD contient tous mes employés !

Monde fermé et logique

- L'hypothèse d'un monde fermé est aussi compatible avec la déduction (donc la logique), mais selon un schéma non-monotone (donc non classique) :
  - logique des défauts (*default logic*),
  - bases de données déductives,
  - programmation logique (Prolog) et par contraintes,
  - Answer Set Programming (ASP).

Plan

Inférence

- En pratique on veut obtenir une formalisation de (certaines) connaissances pour en inférer (beaucoup) d'autres.
- Ceci est possible autant avec
  - des approches non-monotones,
  - que monotones.

Bases de Données

- Dans une base de données, la connaissance est représentée sous forme de
  - *tables*, formées d'*enregistrements* qui sont déterminés par des valeurs d'*attributs*.
  - Une table correspond donc à ce qui s'appelle une *relation* en logique.

Base de Données

- Un système de gestion de base de données (SGBD) est un logiciel qui permet de
  - stocker les tables
  - et répondre à des *requêtes* pour obtenir tous les éléments (valeurs d'attributs) satisfaisant certaines contraintes.

Base de connaissance

- En logique de description, une *base de connaissance* est formée
  - de contraintes terminologiques (TBox) sur les concepts et rôles,
  - d'assertions (ABox) sur les individus.

Base de connaissance

- Un raisonneur est un logiciel qui permet
  - de traiter une base de connaissance (ontologie),
  - et de répondre à des *requêtes* combinant concepts, rôles et individus.

Information négative

- Bases de données et bases de connaissances se distinguent nettement lorsqu'on veut exprimer, par exemple,
  - qu'il n'y a pas d'autre professeur que *jean*,
  - ou que $jean \neq pierre$.
- Dans le cas d'un base de données, l'omission suffit normalement.
- Pour une base de connaissance, il faut l'exprimer explicitement.

Plan

Logique de description

- La logique de description permet donc de créer une représentation de la réalité, formée
  - d'une modélisation représentée par des contraintes terminologiques (TBox),
  - et de faits représentés sous forme d'assertions (ABox).

Modélisation logique

- La modélisation par une logique permet d'exprimer des contraintes complexes,
  - par exemple, de l'information partielle ou incomplète,
  - ce qui pourrait, a priori, sembler nécessiter la non-monotonicité,
  - mais peut tout-à-fait se faire dans un contexte monotone, donc classique.

Moteur d'inférence

- Concrètement, un moteur d'inférence permet de répondre aux questions suivantes :
  - la subsomption : est-ce que $C \sqsubseteq D$ est une conséquence ?
  - vérification d'instance : est-ce que $C(a)$ est une conséquence ?
- et donc à toutes les autres questions qui s'y ramènent :
  - l'équivalence de classes ($C \sqsubseteq D \wedge C \sqsupseteq D$),
  - déterminer si des classes sont disjointes ($C \sqcap D \equiv \bot$),
  - la consistance de classe ($C \equiv \bot$),
  - consistance globale (*false*),
  - trouver tous les individus contenus dans une classe (*instance retrieval*).