

DIC9305 Logique, informatique et sciences cognitives

Modélisation, inférence, base de connaissances II

Roger Villemaire

Département d'informatique
UQAM

21 mars 2024



© 2009-2024 Roger Villemaire, villemaire.roger@uqam.ca
Creative Commons Paternité - Pas d'Utilisation Commerciale - Pas de Modification 3.0 non transcrit.

Plan

- 1 Recherches bibliographiques
- 2 Complexité algorithmique
- 3 Types de logiques de description
- 4 Conclusion

Plan

- 1 Recherches bibliographiques
- 2 Complexité algorithmique
- 3 Types de logiques de description
- 4 Conclusion

Identification des références

- Dans le cadre de vos recherches doctorales, ainsi que tout probablement pour les travaux de ce cours, il vous faudra identifier les publications pertinentes.
- Il existe beaucoup d'outils de recherche, mais il est important d'être en mesure de trouver
 - des sources évaluées par les pairs
 - d'explorer les articles publiés pour trouver ce qui a été fait.

Bibliothèque UQAM

- La bibliothèque de l'UQAM engage des sommes très importantes pour vous permettre d'accéder les publications nécessaires à vos recherches. On note tout particulièrement
 - les articles revus par les pairs (conférences et revues),
 - les livres (souvent électroniques),
 - les outils de recherches bibliographiques.

Scopus

- La bibliothèque est, par exemple, abonnée à Scopus qui permet de
 - trouver des articles revus par les pairs,
 - et ceci tant à partir de mots-clés que de d'autres articles.
- Un usage important est d'explorer, à partir d'un premier article identifié, ceux qui l'ont cité, ceux qu'il cite et ainsi explorer progressivement toutes les publications liées.

Recherches

- Du site bibliotheques.uqam.ca
 - le moteur de recherche (pour les livres et certains articles) s'accède par l'icône Sofia,
 - on trouve aussi au bas de cette page, section *Bases de données de A à Z*, l'accès à Scopus.

Types de recherches

- Scopus permet la recherche
 - par le titre de l'article, ou nom de l'auteur,
 - et aussi par mots-clés, par exemple
 - climate "description logic"
- Il s'agit donc d'un outil très utile pour votre travail de session !

Plan

- 1 Recherches bibliographiques
- 2 Complexité algorithmique**
- 3 Types de logiques de description
- 4 Conclusion

Classes de complexité

- *P*TIME : calculable en temps polynomial,
- *NP*TIME : calculable en temps polynomial avec un algorithme non-déterministe,
- *co-NP* : la négation est NP,
- *PSPACE* : calculable en espace polynomial,
- *EXPTIME* : calculable en temps exponentiel,
- *NEXPTIME* : calculable en temps exponentiel avec un algorithme non-déterministe,
- *N²EXPTIME* : calculable en temps doublement exponentiel avec un algorithme non-déterministe.

problèmes difficiles et complets

- Un problème est dit X -difficile si tous les problèmes de la classe X s'y réduisent.
- Un problème est dit X -complet, s'il est X -difficile et dans la classe X .
- Par exemple, le problème SAT est NP-complet.

Comparaisons

- $P \subseteq NP \subseteq PSPACE \subseteq EXPTIME \subseteq NEXPTIME \subseteq EXPSPACE$
 - $P \subsetneq EXPTIME$,
 - $NP \subsetneq NEXPTIME$
 - $PSPACE \subsetneq EXPSPACE$.
- et on les considère normalement comme de difficultés croissantes même lorsque l'on ne sait pas si les classes sont distinctes.

Plan

- 1 Recherches bibliographiques
- 2 Complexité algorithmique
- 3 Types de logiques de description**
- 4 Conclusion

Logiques de description : opération sur les concepts

- *Attributive language* : $\mathcal{AL} = \{\top, \perp, \neg A, C \sqcap D, \forall r.C, \exists r\}$,
 - où A est *atomique* et $\exists r = \exists r\top$,
- *Complement* : \mathcal{ALC} est l'extension de \mathcal{AL} qui permet la \neg sur tous les concepts,
- *Union* : $\mathcal{ALU} = \mathcal{AL} \cup \{\sqcup\}$,
 - $\mathcal{ALU} \subseteq \mathcal{ALC}$, car $C \sqcup D \equiv \neg(\neg C \sqcap \neg D)$
- *Exists* : $\mathcal{ALE} = \mathcal{AL} \cup \{\exists r.C\}$,
 - $\mathcal{ALE} \subseteq \mathcal{ALC}$, car $\exists r.C \equiv \neg(\forall r.\neg C)$.
- *Numbering* : $\mathcal{ALN} = \mathcal{AL} \cup \{\geq n.r.C, \leq n.r.C\}$,
 - $\mathcal{ALE} \subseteq \mathcal{ALN}$, car $\exists r.C \equiv \geq 1.r.C$,

Logiques de description : opérations sur les rôles

- On considère aussi des opérations sur les rôles :
 - \sqcap , l'intersection sur les rôles.
 - Par exemple, $\mathcal{AL}(\sqcap)$ est \mathcal{AL} avec l'intersection sur les rôles.
 - $pereDe \sqcap habiteAvec$ est le rôle "x est le père et habite avec y".
 - \circ , la composition des rôles.
 - $\mathcal{AL}(\sqcap, \circ)$ est $\mathcal{AL}(\sqcap)$ avec la composition des rôles.
 - $pereDe \circ mereDe$ est le rôle "x est le grand-père maternel de y" car
 - $\exists z.pereDe(x, z) \wedge mereDe(z, y)$.
 - \neg , l'inversion des rôles.
 - $estLUnDesParentsDe^-$ est le rôle $estLEnfantDe$.
 - $+$, la fermeture transitive des rôles.
 - $estLEnfantDe^+$ est le rôle $estLeDescendantDe$.

Comparaison des logiques de description

- Plusieurs logiques de description ont été proposées pour
 - répondre à différents besoins d'expressivité,
 - comme cadre formel pour concevoir des algorithmes.
- Évidemment, on voudrait avoir :
 - la logique la plus expressive, pour pouvoir décrire facilement les concepts et les rôles,
 - mais aussi des algorithmes les plus efficaces possibles.
- C'est un domaine pragmatique, mais il y a toutefois des contraintes théoriques.

Complexité

- $\mathcal{AL} = \{\top, \perp, \neg A, C \sqcap D, \forall r.C, \exists r\}$,
 - subsomption PTIME Schmidt-Schauss and Smolka 1991,
 - satisfaisabilité co-NP-difficile Buchheit et al. 1994.
- \mathcal{ALC} , subsomption et satisfaisabilité sont PSPACE-complètes, Schmidt-Schauss and Smolka 1991.
- $\mathcal{ALN} = \mathcal{AL} \cup \{\geq n.r.C, \leq n.r.C\}$,
 - subsomption PTIME Donini et al. 1997
 - satisfaisabilité PSPACE-complète Küsters 1998.
- $\mathcal{AL}(\sqcap)$, subsomption et satisfaisabilité sont NP-complets, Donini et al. 1997.

Complexité (suite)

- OWL-Lite = \mathcal{SHIF}
 - $\mathcal{S} = \mathcal{ALC} + \mathcal{R}^+$ (clôture transitive des rôles),
 - $\mathcal{H} (\sqsubseteq)$,
 - \mathcal{I} (rôles inverses),
 - \mathcal{F} (fonctionnalité des rôles).
 - subsomption et satisfaisabilité sont EXPTIME-complètes, S. Tobies 2001.
- OWL-DL = \mathcal{SHOIN} :
 - \mathcal{O} , classes énumérée $\{i_1, \dots, i_n\}$,
 - $\mathcal{N} = \{\geq n.r.T, \leq n.r.T\}$,
 - subsomption et satisfaisabilité sont NEXPTIME-complètes, S. Tobies 2000 et 2001.
- OWL 2 DL = \mathcal{SROIQ} ,
 - \mathcal{R} , conjonction des rôles,
 - $\mathcal{Q} = \{\geq n.r.C, \leq n.r.C\}$,
 - subsomption et satisfaisabilité sont N2EXPTIME-complètes, Y. Kazakov 2008.

Profils OWL 2

- La norme OWL 2 du W3C introduit, en plus de OWL 2 DL, des profils, qui sont
 - des versions réduites du langage OWL 2 DL (fragments), limitant les constructions possibles,
 - et visant un certain équilibre en expressivité et performance de raisonnement.
- Il s'agit donc de permettre des applications plus performantes et le traitement de plus grands jeux de données.

- OWL 2 EL
 - permet, entre autres, la quantification existentielle, top, et la conjonction,
 - satisfiabilité et subsumption (pour un TBox) sont polynomiales (Baader, Brandt, Lutz, 2005).
 - permet donc de traiter de très grandes ontologies très efficacement,
 - utilisée, par exemple, par SNOMED une grande ontologie du secteur de la santé.

- OWL 2 QL
 - une variante de DL-Lite ($DL-Lite_R$) qui permet de plus des axiomes d'inclusions de rôles et de rôles disjoints.
 - les requêtes peuvent être transformées pour être traitées par un SGBD, ce qui permet une exécution efficace.

- OWL 2 RL
 - une logique de description dont les TBox peuvent être traduite en règles Datalog (BD déductives),
 - requêtes en temps polynomial (par rapport à la taille des données), (Dantsin, Eiter, Gottlob, Voronkow, 2001),
 - possible d'utiliser des moteurs d'inférences basés sur des règles, qui sont très efficaces.

Plan

- 1 Recherches bibliographiques
- 2 Complexité algorithmique
- 3 Types de logiques de description
- 4 Conclusion

Objectifs

- L'objectif d'une ontologie contruite à l'aide de la logique de description est de
 - représenter des connaissances d'intérêt, en vue
 - d'inférer des conséquences pour
 - compléter nos connaissances, ou encore les valider.

Processus

- On va donc
 - formaliser le minimum nécessaire aux buts visés,
 - en tentant de réutiliser des ontologies ou des fragments d'ontologies existantes,
 - tout en tâchant de tirer le maximum des moteurs d'inférence existants.
- Il s'agit donc de travailler de façon pragmatique, en fonction des objectifs visés.