Formation à la Recherche écrit

Marie-Christine ROUSSET

Equipe Intelligence Artificielle et Systèmes d'Inférence

Laboratoire de Recherche en Informatique (L.R.I)

Université Paris-Sud

Illustration sur l'article Knowledge Compilation and Theory Approximation

- Contexte général : le problème de recherche considéré
  - Conception d'algorithmes d'inférences les plus possibles efficaces sur des formules logiques les plus expressives possibles
- Le sous-problème spécifique étudié dans l'article
  - L'approximation d'un ensemble de formules par des ensembles de formules, ayant un ensemble de modèles inclus dans ou incluant l'ensemble des modèles de l'ensemble de départ, et pour lesquels on connaît des algorithmes d'inférence efficaces
- Intérêt de l'étude de ce sous-problème par rapport au problème général
  - Un compromis permettant de contourner la complexité intrinsèque du problème d'inférence en logique, en utilisant les algorithmes polynomiaux pour faire le plus grand nombre d'inférences possibles, et en faisant appel aux algorithmes plus complexes dans les cas restants

Illustration sur l'article (suite)

- Contributions de l'article :
  - Définition et caractérisation d'ensembles de clauses de Horn (appelées bornes supérieures LUB et inférieures GLB) encadrant logiquement un ensemble T de clauses propositionnelles
  - Utilisation de LUB et GLB pour répondre de façon linéaire à des requêtes sur T
  - Algorithmes ANYTIME de calcul du LUB et d'un GLB, pouvant être utilisés offline
  - Estimation de la complexité en temps et en place des algorithmes, et du gain de l'utilisation du LUB et du GLB pour la réponse à des requêtes sur T
  - Un cadre formel généralisant les notions de LUB et GLB définies en logique propositionnelle

Illustration sur l'article (suite)

- Définition et caractérisation d'ensembles de clauses de Horn (appelées bornes supérieures LUB et inférieures GLB) encadrant logiquement un ensemble T de clauses propositionnelles :
  - LUB et GLB sont telles que GLB $\models$ T $\models$ LUB et il n'existe pas d'ensembles de clauses de Horn T' et T'' tels que GLB $\models$ T'$\models$ T $\models$ T''$\models$LUB avec T' $\not\models$ GLB et LUB $\not\models$ T''
  - LUB est unique : l'ensemble des clauses de Horn impliqués premiers de T ; une clause de Horn C est impliqué premier de T ssi T $\models$ C et il n'existe pas de clauses de Horn C'(non équivalente à C) telle que T $\models$ C' $\models$ C
  - Il existe plusieurs GLBs possibles : autant que de « Horn-strenghtening » des clauses de T ; un Horn-strenghtening d'une clause C est une clause de Horn obtenue à partir de C en enlevant tous ses littéraux positifs sauf un

Illustration sur l'article (suite)

- Contributions de l'article
  - Leur utilisation pour l'inférence de requêtes :
    - Pour tester si T $\models$ q, on teste d'abord si LUB $\models$ q (se fait en temps linéaire en fonction de la taille de LUB et de la requête) : si la réponse est oui, on a déduit (en temps linéaire) que T$\models$q si la réponse est non, on teste si GLB $\models$ q : si la réponse est non, on peut en déduire (en temps linéaire) que T$\not\models$q , si la réponse est oui, alors soit : on répond « je ne sais pas » (algo linéaire mais incomplet), soit on applique un algorithme complet pour tester directement si T $\models$ q
    - Le problème est d'évaluer le nombre de requêtes q auxquelles on peut répondre en ne faisant appel qu'à LUB ou GLB, mais aussi de s'assurer que ces requêtes ne sont pas des instances faciles pour le test direct T $\models$ q, et enfin, d'évaluer le temps pour calculer LUB et GLB, ainsi que la taille de ces ensembles de clauses LUB et GLB

Illustration sur l'article (suite)

- Contributions de l'article
  - Algorithmes de calcul des GLBs et de la LUB :
    - 2 algorithmes ANYTIME (interruptibles à tout moment et fournissant alors un UB et un LB) qui, s'ils ne sont pas interrompus, garantissent de fournir le LUB et un GLB.
    - Leur complexité ? Dans le cas le pire exponentielle, à cause du théorème 1 qui dit que T est satisfiable ssi LUB ou GLB le sont aussi. On en déduit que s'il existait des algorithmes de complexité polynomiale pour calculer LUB ou GLB, on saurait résoudre SAT de façon polynomiale
    - Donc, le seul intérêt de calculer LUB et GLB est d'amortir le coût de leur calcul par le gain cumulé lors de l'interrogation de GLB et LUB (au lieu de T) pour un ensemble de requêtes.

Illustration sur l'article (suite)

- Contributions de l'article
  - Estimation du coût de calcul des GLB et LUB et du gain (en temps et en espace) de leur exploitation pour l'inférence de requêtes
  - En temps: plusieurs questions considérées
    - Les requêtes pour lesquelles on obtient une réponse par GLB ou LUB ne sont-elles pas des cas où le calcul direct est « facile » (polynomial) ? Un résultat datant de 86 montre que trouver le modèle d'une théorie propositionnelle qui n'a qu'un modèle est difficile (dans NP). Donc, on ne peut répondre directement à des requêtes de façon polynomiale (sinon, on pourrait utiliser cette méthode directe pour trouver le modèle en testant caque littéral comme une requête), alors qu'on peut répondre de façon linéaire à toutes les requêtes par GLB et LUB.
    - Evaluation théorique et expérimentale du gain réalisé pour évaluer des requêtes sur des théories « difficiles » pour SAT (clauses de longueur fixe(3) avec un ratio de 4.3 entre le nombre de clauses et le nombre de variables)
      - Calcul probabiliste de la proportion de requêtes traitables par LUB ou GLB
      - Evaluation expérimentale du gain du calcul par approximation pour un ensemble de 2000 requêtes (binaires ou ternaires) : gain important, même en tenant compte du calcul des bornes

Illustration sur l'article (suite)

- Contributions de l'article
  - Estimation du coût de calcul des GLB et LUB et du gain (en temps et en espace) de leur exploitation pour l'inférence de requêtes
  - En place : évaluation de la taille des approximations
    - GLB : par construction, sa taille est inférieure à celle de T
    - LUB :
      - Il peut exister des théories T de taille n dont le LUB soit de taille exponentielle par rapport à n (2^n)
      - Pour de tels cas, il y a des manières de modifier T en T' en ajoutant de nouvelles variables de façon que le LUB de T' ait une taille en O(n) par rapport à la taille de T'.
      - On ne peut pas toujours procéder à cet artifice.

Illustration sur l'article (suite)

- Un cadre formel généralisant les notions de LUB et GLB définies en logique propositionnelle
  - Définition d'un cadre général de définition d'approximation de théories logiques exprimées dans un langage L
    - Définition d'un système d'approximations de théories exprimées dans L par des théories d'un sous-langage de L (pour lequel le problème d'inférence est tractable) pour répondre à des requêtes d'un sous-langage de L
    - Nécessité d'avoir des algorithmes « anytime » pour calculer les approximations de théories de L dans les sous-langahes considérés.
    - Différentes instances du problème : le problème étudié dans la première partie de l'article n'est qu'un cas particulier du problème général

Illustration sur l'article (suite)

- Cadre formel(suite)
  - Pistes pour la faisabilité de sa mise en oeuvre pour des fragments de la logique des prédicats
    - Problème : l'existence de GLB n'est pas garantie en général,
    - Pour des théories clausales:
      - existence de GLBs et généralisation facile du cas propositionnel,
      - Problème pour le LUB: peut être infini
      - Approximation de théories en logiques de description : calculer pour chaque description de concepts d'une LD expressive un encadrement par 2 concepts (subsumant bet subsumé) d'une LD moins expressive
      - Pistes: approximer des théories indécidables par des théories décidables (éventuellement tractable)

Illustration sur l'article (suite)

- Situation par rapport aux travaux existants
  - Lien avec les travaux portant sur la généralisation (abstraction) et la spécialisation de théories :
    - Généralisation <-> UB, spécialisation <-> LB
    - Les travaux sur la généralisation de théories consiste à « abstraire » une théorie et le problème d'inférence associé (les requêtes) pour simplifier ce problème
    - Les travaux sur la spécialisation consistent à remplacer une théorie par un de ses modèles ou des clauses de Horn de taille k : ce sont des LBs pas des GLBs
  - Points distinctifs principaux de l'article
    - On ne simplifie pas le problème mais on essaie de résoudre le problème initial sur des théories plus simples avant de le résoudre dans la théorie initiale
    - On amortit le calcul des approximations sur un ensemble de résolution de requêtes
    - On définit un cadre formel général dans lequel les travaux précédents (faits indépendamment les uns des autres) peuvent être englobés et comparés

Illustration sur l'article (suite)

- Perspectives :
  - Explorer les différentes pistes suggérées dans la dernière partie de l'article pour définir et calculer des approximations pour différents formalismes de représentation de connaissances.