

Programmation de méthodes de raisonnement

Description

Dans ce projet, on souhaite répondre de façon algorithmique à la question “est-ce qu’une formule α découle d’un ensemble de formules \mathcal{KB} (c’est à dire, est-ce que $\mathcal{KB} \models \alpha$)?”, où \mathcal{KB} est une base de connaissances (*knowledge base*) contenant un nombre fini de formules du langage sous jacent.

Plusieurs méthodes pour l’automatisation de cette question ont été proposées dans la littérature en IA symbolique. Parmi celles-là, on compte (1) la méthode par tables de vérités (pas optimale), (2) la méthode de résolution, (3) la méthode des tableaux sémantiques, et (4) la méthode des connexions, sans oublier de mentionner bien sûr l’énorme progrès obtenu dans la communauté des solvers SAT avec des implémentations hyper-optimisées.

Votre tâche consiste à :

1. Choisir deux parmi les quatre méthodes mentionnées ci-dessus ;
2. Rechercher les détails et définitions des méthodes choisies afin de comprendre leur fonctionnement ;
3. Les implémentez dans le langage de programmation de votre choix ;
4. Réalisez les tests nécessaires afin de vous assurer que votre implémentation fonctionne correctement.

Consultez les critères d’évaluation en annexe afin de connaître les points qui seront pris en compte lors de l’évaluation de votre programme.

Bien évidemment, vous pouvez vous servir de vos réponses aux exercices pratiques précédents, notamment le parser.

Modalités

- A faire en binôme.
- Communiquer la composition du binôme à l’enseignant par e-mail au préalable.
- Date limite de dépôt : le vendredi 10 mai 2024 à 13h00 heure de Paris.
- Lieu de dépôt : espace Moodle, section “Projet de recherche pratique”.

Travail à rendre

Un seul répertoire contenant le code source de l’implémentation incluant aussi la documentation générée (et oui, sinon comment on va pouvoir utiliser votre implémentation...) Déposez tout le contenu ci-dessus en un seul fichier .zip par binôme. Pensez à adoptez la convention suivante pour le nom du fichier : `<NomsEnOrdreAlphabétique>.zip`. Exemple : si votre binôme est composé de Rachid Bergougga et Jean-Marie Lafayette, nommez votre fichier à déposer `BergouggaLafayette.zip`.

Vous pouvez mettre à jour régulièrement le fichier soumis sur Moodle en écrasant la version précédente. Mais attention : seulement une soumission par binôme à la fin pour éviter des doublons lors de la correction.

Mais encore attention : des méthodes pointues de détection de plagiat seront utilisées lors de la correction (bah oui, *with great power comes great responsibility*!).

N’hésitez pas à contacter l’enseignant en cas de question pendant la réalisation du projet. Bon travail!
N.B. : Consultez régulièrement Moodle pour des éventuelles mises-à-jour de ce sujet.

Critères d'évaluation

Cas éliminatoires :	Note :
Le rendu ne respecte pas les consignes de dépôt (format de fichier, nommage, etc)	0
Le programme ne compile pas ou n'exécute pas	0
Le code ne passe pas les tests de plagiat	0

Cas évaluable :	Points :
La documentation mentionne les bibliothèques utilisées nécessaires pour exécuter le code ?	1
La documentation explique bien comment utiliser le programme ?	2
Le nombre d'atomes et la taille des formules ne sont pas limités ?	3
Le programme traite correctement les parenthèses ?	2
Le programme passe les tests si une formule découle d'une base de connaissance ?	12
Bonus : Nombre de méthodes choisies implémentées et fonctionnelles > taille du groupe ?	8

La note de votre travail correspondra à la somme des points indiqués ci-dessus.