

Towards Exploring Inductive Logic Learning with the Andante Notebook

Sur l'explicitation de l'apprentissage inductif logique avec le notebook Andante

SIMON JACQUET, SARAH PINON, JEAN-MARIE JACQUET, ISABELLE LINDEN, and WIM VAN-HOOF, NADI Research Institute, University of Namur, Belgium

Inductive logic programming is a form of machine learning which consists in inducing logical formulas from a background theory as well as positive and negative examples. Many systems have been proposed, of which the most famous are FOIL, Progol, Aleph and Metagol (see [1] for a description). If the results produced are explainable because of their declarative character, the fine understanding of the learning process as well as the models produced raises many questions of intelligibility. The notebook Andante has been designed to tackle these issues. More precisely, in order to allow a user to analyze finely both the learning process as well as the models produced and, consequently, to increase the interpretability of the models generated, we have built a notebook offering the following functionalities: (i) the lecture of programs in tailored execution contexts, (ii) the execution of Prolog queries allowing to test the code introduced but also different working hypotheses, (iii) the definition of high level auxiliary concepts by means of suitable predicates, reusable in the learning process, (iv) the generation of models induced by different methods and the inspection of intermediate results produced during this generation.

Additional Key Words and Phrases: Explainability, Inductive Logic Programming

La programmation logique inductive consiste à apprendre des formules logiques à partir d'une théorie de base et d'exemples positifs et négatifs. De nombreux systèmes ont été proposés dont les plus connus sont FOIL, Progol, Aleph et Metagol (cf [1] pour une description). Si les résultats produits sont explicables de part leur caractère déclaratif, la compréhension fine du processus d'apprentissage ainsi que des modèles produits soulève néanmoins de nombreuses questions d'intelligibilité. C'est aux fins de pouvoir répondre à ces questions qu'a été développé le notebook Andante. Plus précisément, de manière à permettre à un utilisateur d'analyser finement le processus d'apprentissage ainsi que les modèles produits et, par suite d'augmenter l'interprétabilité des modèles générés, nous avons construit un notebook offrant les fonctionnalités suivantes : (i) la lecture de programmes dans des contextes d'exécution personnalisés, (ii) l'exécution de requêtes Prolog permettant de tester le code introduit mais aussi différentes hypothèses de travail, (iii) la définition de concepts auxiliaires de haut niveau par le biais de prédicats adaptés, réutilisables dans le processus d'apprentissage, (iv) la génération de modèles induits par différentes méthodes et l'inspection de résultats intermédiaires produits lors de cette génération.

Mots-clés additionnels : Explicabilité, Programmation logique inductive

ACM Reference Format:

Simon Jacquet, Sarah Pinon, Jean-Marie Jacquet, Isabelle Linden, and Wim Vanhoof. 2022. Towards Exploring Inductive Logic Learning with the Andante Notebook. In *IHM '22 : 33^e conférence Francophone sur l'Interaction Humain-Machine, April 05–08, 2022, Namur, Belgique*. ACM, New York, NY, USA, 1 page. <https://doi.org/10.1145/XXXXXXX.XXXXXXX>

REFERENCES

- [1] Andrew Cropper and Sebastijan Dumancic. 2020. Inductive logic programming at 30: a new introduction. *CoRR* abs/2008.07912 (2020). <https://arxiv.org/abs/2008.07912>

Unpublished working draft. Not for distribution.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2022 Association for Computing Machinery.

Manuscript submitted to ACM