Logic-Based Explainable and Incremental Machine Learning*

Gopal Gupta[1], Huaduo Wang[1], Kinjal Basu[2], Farhad Shakerin[1], Parth Padalkar[1], Elmer Salazar[1], Sarat Chandra Varanasi[1], Sopam Dasgupta[1]

[1]Computer Science Department, The University of Texas at Dallas, Richardson, USA
[2]IBM Research, USA
gupta@utdallas.edu

Abstract

Mainstream machine learning methods lack *interpretability*, *explainability*, *incrementality*, and *data-economy*. We propose using logic programming (LP) to rectify these problems. We discuss the FOLD family of rule-based machine learning algorithms that learn models from relational datasets as a set of default rules. These models are competitive with state-of-the-art machine learning systems in terms of accuracy and execution efficiency. We also motivate how logic programming can be useful for *theory revision* and *explanation based learning*.

Introduction

Dramatic success of machine learning has led to a plethora of artificial intelligence (AI) applications. The effectiveness of these machine learning systems, however, is limited in several ways:

1. **Lack of Interpretability:** The models learned by machine learning systems are opaque, i.e., they are not comprehensible by humans. This is mainly because these statistical machine learning methods produce models that are complex algebraic solutions to optimization problems such as risk minimization or likelihood maximization.
2. **Lack of Explainability:** These models are unable to produce a justification for a prediction they compute for a new data sample.
3. **Lack of Incrementality:** These methods are unable to incrementally update a learned model as new data is encountered.
4. **Lack of Data Economy:** These methods need large amounts of data to compute a model. Humans, in contrast, are able to learn from a small number of examples.

In this position paper we show that these problems are greatly alleviated if we develop machine learning methods that learn default theories coded in logic programming (LP). The whole field of inductive logic programming (ILP) has been developed in which Horn clauses are learned from background knowledge, positive, and negative examples (Cropper and Dumancic 2020). Rules with negated goals in the body are also learned in ILP as nonmonotonic logic programs and default rules (Srinivasan, Muggleton, and Bain 1992; Dimopoulos and Kakas 1995). Representing a model as default rules brings significant advantages wrt interpretability, explainability, incremental learning, and data economy. We present LP-based machine learning algorithms that are interpretable and explainable, as well as LP-based reinforcement learning for incremental learning, and LP-based explanation based learning for solving data economy issues. Our explainable LP-based machine learning methods (Shakerin, Salazar, and Gupta 2017; Wang and Gupta 2022, 2024) are competitive with state-of-the-art techniques such as XGBoost (Chen and Guestrin 2016) and Multilayer Perceptrons/Neural Networks (Aggarwal 2018). Table 1 shows the performance comparison of FOLD-SE (Wang and Gupta 2024), an explainable LP-based ML algorithm, with XGBoost and MLP on a binary classification task. What sets FOLD-SE apart from other explainable ML algorithms is that it can learn a succinct logic based rule-set from the data that can then be used to make predictions. Table 2 shows a comparison of FOLD-SE with another popular explainable ML algorithm RIPPER. FOLD-SE achieves greater or comparable accuracy while generating a significantly smaller rule-set.

NeSyFOLD (Padalkar, Wang, and Gupta 2023; Padalkar and Gupta 2023) is a framework that uses the FOLD-SE-M algorithm (for multiclass classification) to generate a global explanation from a CNN trained on an image classification task. The output of the last layer kernels are binarized for the entire train set. The FOLD-SE-M algorithm is then used to learn a rule-set wherein each predicate's truth value is determined by the binarized kernel's output. Each kernel can be mapped to the concept(s) that it learns to identify in the images and its corresponding predicate can be labelled as those concept(s). Fig 1 illustrates the NeSyFOLD framework used for a CNN trained to classify images of "bathroom", "bedroom" and "kitchen". The rule-set that is obtained can be scrutinized by a domain expert to check for biases that the CNN might have learnt.

Default rules are an excellent way of capturing the logic underlying a relational dataset. Defaults are used by humans in their day-to-day reasoning (Stenning and van Lambalgen 2008; Dietz Saldanha, Hölldobler, and Pereira 2021). Most datasets are generated from human-driven activity (e.g., loan

*Authors partially supported by NSF grant IIS 1910131.

Data Set			XGBoost			MLP			FOLD-SE		
Name	Rows	Cols	Acc	F1	T(ms)	Acc	F1	T(ms)	Acc	F1	T(ms)
acute	120	7	**1.0**	**1.0**	122	0.99	0.99	22	**1.0**	**1.0**	**1**
heart	270	14	**0.82**	**0.83**	247	0.76	0.78	95	0.74	0.77	**13**
breast-w	699	10	0.95	0.96	186	**0.97**	**0.98**	48	0.94	0.92	**9**
eeg	14980	15	0.64	**0.71**	46,472	**0.69**	**0.71**	9,001	0.67	0.68	**1,227**
credit card	30000	24	NA	NA	NA	NA	NA	NA	**0.82**	**0.89**	**3,513**
adult	32561	15	**0.87**	**0.92**	424,686	0.81	0.87	300,380	0.84	0.90	**1,746**
rain in aus	145460	24	**0.84**	**0.90**	385,456	0.81	0.88	243,990	0.82	0.89	**10,243**

Table 1: Comparison of XGBoost, MLP, and FOLD-SE

Data Set			RIPPER					FOLD-SE				
Name	Rows	Cols	Acc	F1	T(ms)	Rules	Preds	Acc	F1	T(ms)	Rules	Preds
acute	120	7	0.93	0.92	95	**2.0**	4.0	**1.0**	**1.0**	**1**	**2.0**	**3.0**
heart	270	14	0.76	0.77	317	5.4	12.9	0.74	0.77	**13**	**4.0**	**9.1**
breast-w	699	10	0.93	0.90	319	14.4	19.9	0.94	0.92	**9**	**3.5**	**6.3**
eeg	14980	15	0.55	0.36	12,996	43.4	134.7	0.67	0.68	**1,227**	**5.1**	**12.1**
cr. card	30000	24	0.76	0.84	49,940	36.5	150.7	**0.82**	**0.89**	**3,513**	**2.0**	**3.0**
adult	32561	15	0.71	0.77	63,480	41.4	168.4	**0.84**	**0.90**	**1,746**	**2.0**	**5.0**
rain in aus	145460	24	0.63	0.70	3118,025	180.1	776.4	**0.82**	**0.89**	**10,243**	**2.5**	**6.1**

Table 2: Comparison of RIPPER and FOLD-SE. Preds denotes the number of predicates in the rule-set.

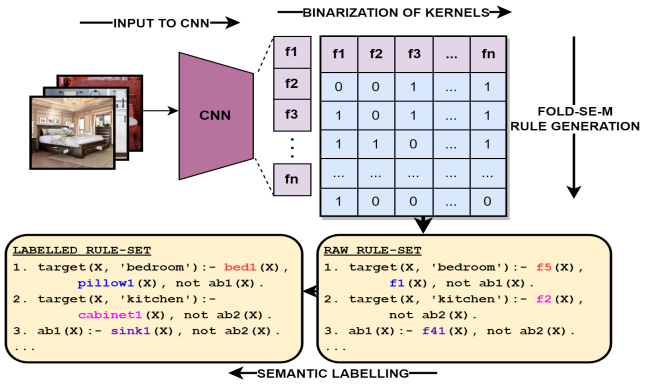

Figure 1: The NeSyFOLD framework.

approval by bank officials) and our experiments indicate that the rules underlying the model learned from these datasets can be represented quite faithfully and succinctly with default rules. Default rules are used by humans to learn a concept in an elaboration tolerant manner, as they allow humans to constantly adjust the decision boundary. We have developed machine learning algorithms that learn default rules (the *model*) from relational data containing categorical (i.e., discrete) and numerical values that are competitive with state-of-the-art machine learning techniques. These algorithms are *interpretable* and *explainable*.

Once a set of default rules has been learned from data, it is possible that these rules may be wrong (possibly because we over-generalized or under-generalized). When human beings learn from examples (by formulating a rule of thumb in their mind), then when they encounter an example that goes against the learned rule, they revise the rule in light of this new example. For example, suppose we learn the rule that if object `X` is a fruit, then it goes into the refrigerator. Later, we learn from experience or someone may tell us that pineapples must not go into the refrigerator. In that case, we will revise the rule, changing it to: if `X` is a fruit, it goes into the refrigerator, except for pineapples. This is a form of *incremental* or reinforcement learning (Aggarwal 2018). We will refer it to as logic-based reinforcement learning; we can think of it as theory revision. Logic-based reinforcement learning is elegantly modeled in logic programming using default theories as well.

Traditional machine learning methods need large amounts of data to learn. In contrast, humans can learn from a small number of examples. The problem of learning from a small number of examples has been explored under the topic of explanation-based learning (EBL) (Minton et al. 1989). Explanation-based learning can be further developed and applied to practical applications within the framework of logic programming through the use of default theories.

Finally, knowledge expressed as a logic program can be incorporated in the neural learning process. Thus, logic programming can play an important role in neuro-symbolic learning (Yang, Ishay, and Lee 2020; Mitchener et al. 2022). Logic programming can make a significant difference in this area.

More details on ideas described in this extended abstract can be found elsewhere (Gupta et al. 2023). Our hope is that more effort will be invested by the research community in investigating logic programming based solutions to explainability and interpretability for machine learning.

References

Aggarwal, C. C. 2018. *Neural Networks and Deep Learning - A Textbook*. Springer.

Chen, T.; and Guestrin, C. 2016. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD*, KDD '16, 785–794. ISBN 978-1-4503-4232-2.

Cropper, A.; and Dumancic, S. 2020. Inductive logic programming at 30: a new introduction. ArXiv:2008.07912.

Dietz Saldanha, E.; Hölldobler, S.; and Pereira, L. 2021. Our Themes on Abduction in Human Reasoning: A Synopsis. In *Abduction in Cognition and Action: Logical Reasoning, Scientific Inquiry, and Social Practice*, 279–293.

Dimopoulos, Y.; and Kakas, A. 1995. Learning non-monotonic logic programs: Learning exceptions. In Lavrac, N.; and Wrobel, S., eds., *Machine Learning: ECML-95*, 122–137. Berlin, Heidelberg.

Gupta, G.; Wang, H.; Basu, K.; Shakerin, F.; Salazar, E.; Varanasi, S. C.; Padalkar, P.; and Dasgupta, S. 2023. Logic-Based Explainable and Incremental Machine Learning. In *Prolog: The Next 50 Years*, volume 13900 of *Lecture Notes in Computer Science*, 346–358. Springer.

Minton, S.; Carbonell, J. G.; Knoblock, C. A.; Kuokka, D.; Etzioni, O.; and Gil, Y. 1989. Explanation-Based Learning: A Problem Solving Perspective. *Artif. Intell.*, 40(1-3): 63–118.

Mitchener, L.; Tuckey, D.; Crosby, M.; and Russo, A. 2022. Detect, Understand, Act: A Neuro-symbolic Hierarchical Reinforcement Learning Framework. *Mach. Learn.*, 111(4): 1523–1549.

Padalkar, P.; and Gupta, G. 2023. Using Logic Programming and Kernel-Grouping for Improving Interpretability of Convolutional Neural Networks. In *Proc. Symp. on Practical Aspects of Declarative Languages*. Springer Verlag. To appear.

Padalkar, P.; Wang, H.; and Gupta, G. 2023. NeSyFOLD: Neurosymbolic Framework for Interpretable Image Classification. arXiv:2301.12667.

Shakerin, F.; Salazar, E.; and Gupta, G. 2017. A new algorithm to automate inductive learning of default theories. *TPLP*, 17(5-6): 1010–1026.

Srinivasan, A.; Muggleton, S. H.; and Bain, M. 1992. Distinguishing Exceptions From Noise in Non-Monotonic Learning. Proc. International Workshop on Inductive Logic Programming.

Stenning, K.; and van Lambalgen, M. 2008. *Human Reasoning and Cognitive Science*. MIT Press.

Wang, H.; and Gupta, G. 2022. FOLD-R++: A Scalable Toolset for Automated Inductive Learning of Default Theories from Mixed Data. In *Functional and Logic Programming: 16th International Symposium, FLOPS 2022*, 224–242. Springer-Verlag. ISBN 978-3-030-99460-0.

Wang, H.; and Gupta, G. 2024. FOLD-SE: An Efficient Rule-based Machine Learning Algorithm with Scalable Explainability. Forthcoming.

Yang, Z.; Ishay, A.; and Lee, J. 2020. NeurASP: Embracing Neural Networks into Answer Set Programming. In Bessiere, C., ed., *IJCAI 2020*, 1755–1762.