Online Learning of Logic Based Neural Network Structures

Victor Guimarães and Vítor Santos Costa

CRACS and DCC/FCUP – Universidade do Porto – Porto, Portugal

victorguimaraes13@gmail.com, vsc@dcc.fc.up.pt

INESCTEC | FACULDADE DE CIÊNCIAS UNIVERSIDADE DO PORTO | U.PORTO | FCT Fundação para a Ciência e a Tecnologia

Introduction

In this work, we present two online structure learning algorithms for **NeuralLog** [3]. Both algorithms are based on the **Online Structure Learner by Revision (OSLR)** [1].

- **NeuralLog+OSLR**, a ported version of **OSLR** to NeuralLog; and
- **NeuralLog+OMIL**, an **O**nline application of the **M**eta-**I**nterpretive **L**earning (MIL) algorithm [6].

NeuralLog is a first-order logic language that is compiled into a neural network [3]. In NeuralLog, the knowledge is represented as sparse matrices and the rules are evaluated by mathematical operations on those matrices. In addition, each fact has an associated weight that might be tuned by the neural network, based on examples.

Online Structure Learning Algorithms

Online Structure Learner by Revision (OSLR) is a theory revision algorithm that revises the logic theory in order to cope with the arrival of new examples [2, 1]. OSLR uses a tree structure to both represent the logic theory; and to store the incoming examples in its leaves, depending on the part of the theory that proofs the examples. Then, these leaves are used as revision points to which revision operators are applied.

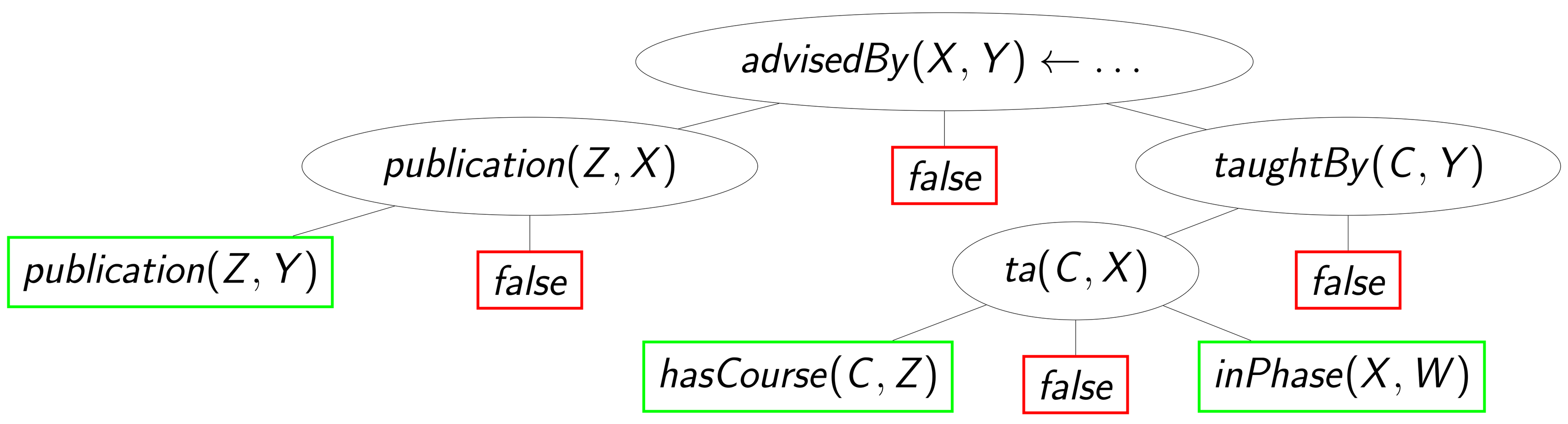


Figure 1: Tree Structure Representation of the UWCSE Theory Example

Table 1: Theory Example for the UWCSE Dataset

$$advisedBy(X, Y) \leftarrow taughtBy(C, Y) \wedge ta(C, X) \wedge hasCourse(C, Z).$$
$$advisedBy(X, Y) \leftarrow taughtBy(C, Y) \wedge ta(C, X) \wedge inPhase(X, W).$$
$$advisedBy(X, Y) \leftarrow publication(Z, X) \wedge publication(Z, Y).$$

NeuralLog+OSLR uses the same revision operators from OSLR, which can delete nodes from the tree, or add nodes to the tree based on the bottom clause [5]. On the other hand, **NeuralLog+OMIL** uses a higher-order logic theory in order to propose new nodes to be added to the tree. Table 2 shows an example of a higher-order logic theory.

Table 2: The Higher-order Logic Theory

$$P(X, Y) \leftarrow Q(X, Y).$$
$$P(X, Y) \leftarrow Q(Y, X).$$
$$P(X, Y) \leftarrow Q(X, Z) \wedge R(Z, Y).$$

Conclusion

In this work, we presented **NeuralLog+OSLR** and **NeuralLog+OMIL**, two online structure learning algorithms for NeuralLog.

We compared our proposal with OSLR [2, 1] and RDN-Boost [4] on link prediction tasks.

As future work, we would like to experiment NeuralLog structure learning algorithms on datasets mixing the relation part of the neural network with propositional neural network models.

References

[1] V. Guimarães, A. Paes, and G. Zaverucha. Online probabilistic theory revision from examples with ProPPR. *Machine Learning*, 108(7):1165–1189, Jul 2019.

[2] V. Guimarães. Online probabilistic theory revision from examples: A ProPPR approach. Master's thesis, PESC, COPPE, Federal University of Rio de Janeiro, Rio de Janeiro, RJ, Brazil, 2018.

[3] V. Guimarães and V. S. Costa. Neurallog: A neural logic language, 2021.

[4] T. Khot, S. Natarajan, K. Kersting, and J. Shavlik. Gradient-based boosting for statistical relational learning: the markov logic network and missing data cases. *Machine Learning*, 100(1):75–100, 2015.

[5] S. Muggleton. Inverse entailment and progol. *New Generation Computing*, 13(3):245–286, 1995.

[6] S. H. Muggleton, D. Lin, and A. Tamaddoni-Nezhad. Meta-interpretive learning of higher-order dyadic datalog: predicate invention revisited. *Mach. Learn.*, 100(1):49–73, 2015.

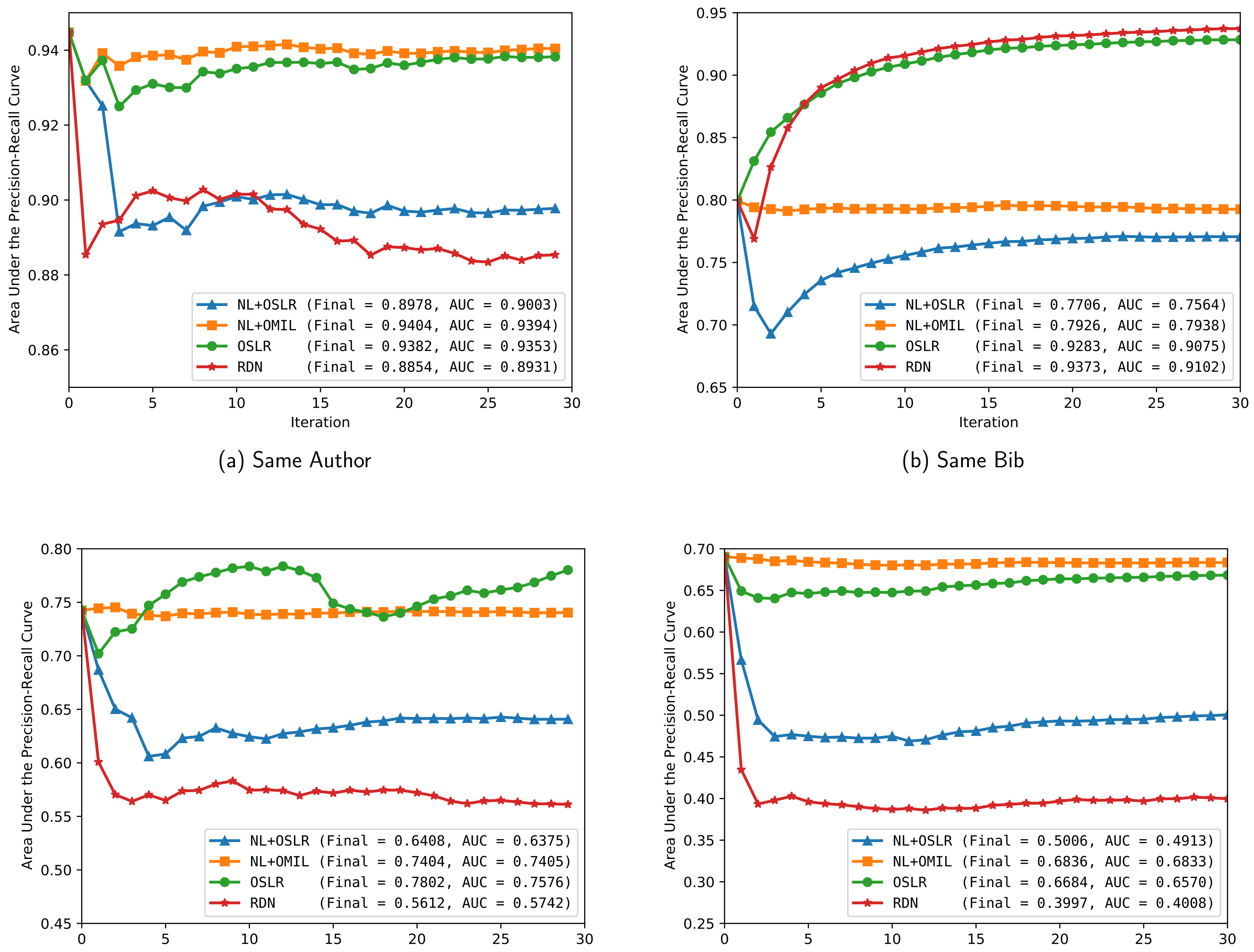
Results

In order to show the capabilities of **NeuralLog+OSLR** and **NeuralLog+OMIL**, we compared them with OSLR [1] and RDN-Boost [4], for link prediction, in six relations of three distinct datasets: Cora, Unified Medical Language System (UMLS) and UWCSE datasets. Table 3 shows the sizes of each dataset. In order to evaluate the online systems, we simulated an online environment, where batch of examples arrives over time.

Table 3: Size of the Datasets

Datasets	Relation	# Positives	# Negatives
Cora	Same Author	488	66
	Same Bib	30,971	21,952
	Same Title	661	714
	Same Venue	2,887	4,976
UMLS	Affects	1,022	-
UWCSE	Advised by	113	16,601

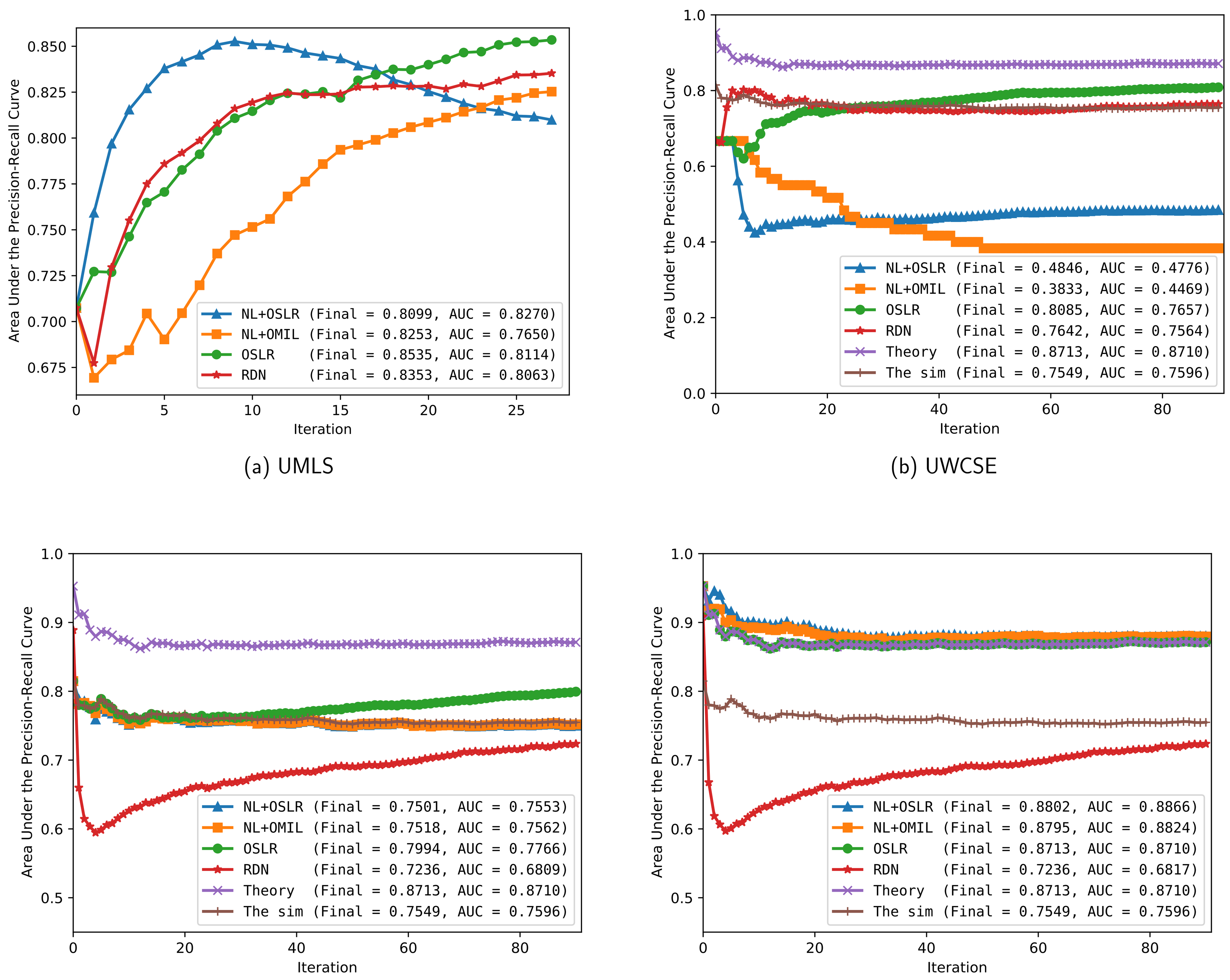
Figure 2 shows the evaluation of the systems in the Cora dataset over time.



(a) Same Author (b) Same Bib (c) Same Title (d) Same Venue

Figure 2: The Evaluation of the Cora Dataset

Figure 3a shows the results of the experiment for the UMLS dataset. While Figures 3b-3d show the results for the UWCSE dataset. Figure 3b shows an experiment where the systems start from an empty theory, like the previous experiments; while in Figure 3c, all systems start from a simplified theory; finally, in Figure 3d, all systems start from a complete theory.



(a) UMLS (b) UWCSE (c) UWCSE Simplified Initial Theory (d) UWCSE Complete Theory

Figure 3: The Evaluation of the UMLS and UWCSE Datasets

October 25, 2021