An Architecture For Relational Learning Through Iterative Search Over Hypothesis Space

KR2ML

Osama F Rama, Alessandra Russo, Krysia Broda
Department of Computing, Imperial College London

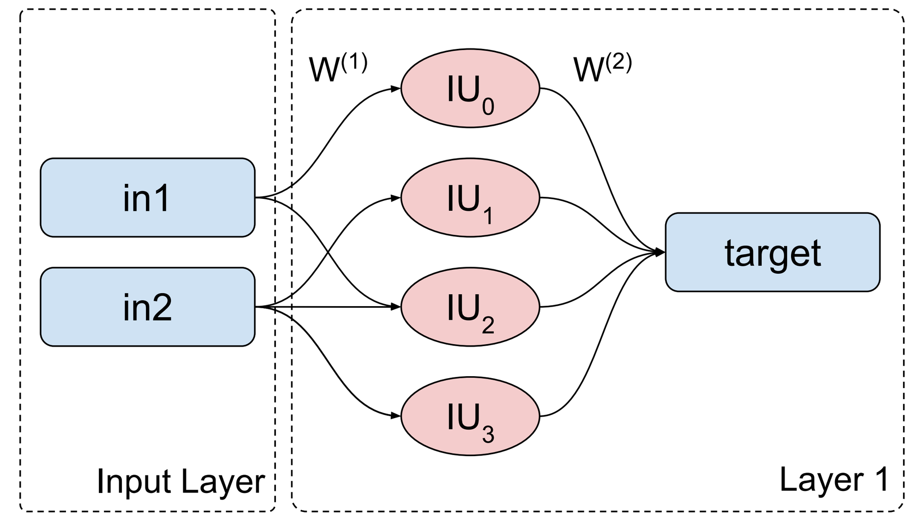
Introduction

We present Neural Concept Network (NCN), an architecture for relational learning through neural guided search. The architecture is composed of input nodes, which represent concepts given as facts in a knowledge base, output nodes, which represent positive and negative ground instances of a concept to learn, and a hidden implication layer that captures the space of possible solutions. As this space is in practice very big, the implication layer is dynamically constructed during the training process as a neural guided search that learns the most relevant solutions. Inference in NCN involves computing the gating tensor, based on the network structure, which semantically filters input nodes and maps them to ground instances of the output nodes. We evaluate the approach over two classes of problems, inductive learning and knowledge-base completion, and show that NCN achieves similar or better performance than existing methods.

Architecture

The architecture of NCN is characterised by two types of nodes, *Implication Unit* (IU) and *Concept unit* (CU). The latter refer to input and output nodes. An IU imposes a semantic filter over the input units, expressed as conjunction over the corresponding input concepts. It produces grounded output when the variable bindings are satisfied for in the instances of input concepts. As different IUs may produce the same ground instance of the target concept, the latter can be seen as a disjunction of alternative definitions that different IUs may yield to. Figure 1 illustrates an example instance of NCN with two input units, corresponding to two binary concepts *in1* and *in2*, one binary target concept *target*, and 4 IUs, each representing a (singleton) conjunction of the input concepts (see table in Figure 1). So the target concept can be seen as defined by the disjunction of the 4 IUs.

The forward propagation through a layer involves three tensor operations (Equations 1-3). Equation 1 filters and maps the input activation values onto the gating tensor and aggregates over the input facts. Θ^l is the intermediate value of the IL. Φ^l is the output value of the IL. Intuitively, Equations 1 and 2 together are similar to the convolution operator, except in this case, the filter (represented by the IU) is applied semantically to the input, i.e. the filter has a semantic prior as opposed to the structural prior based on locality in CNN. The filter is dynamically imposed on the input, where the imposition is captured by the gating tensor G^l in Equation 1. Equation 3 finally aggregates the output of the IUs connected to the output CUs, to obtain the final output A^l.



IU	Description
IU_0	*in1*(V2,V1)
IU_1	*in2*(V2,V1)
IU_2	*in1*(V1,V3) $\wedge$ *in2*(V3,V2)
IU_3	*in2*(V1,V3) $\wedge$ *in2*(V3,V2)

Figure 1: An example instance of NCN.

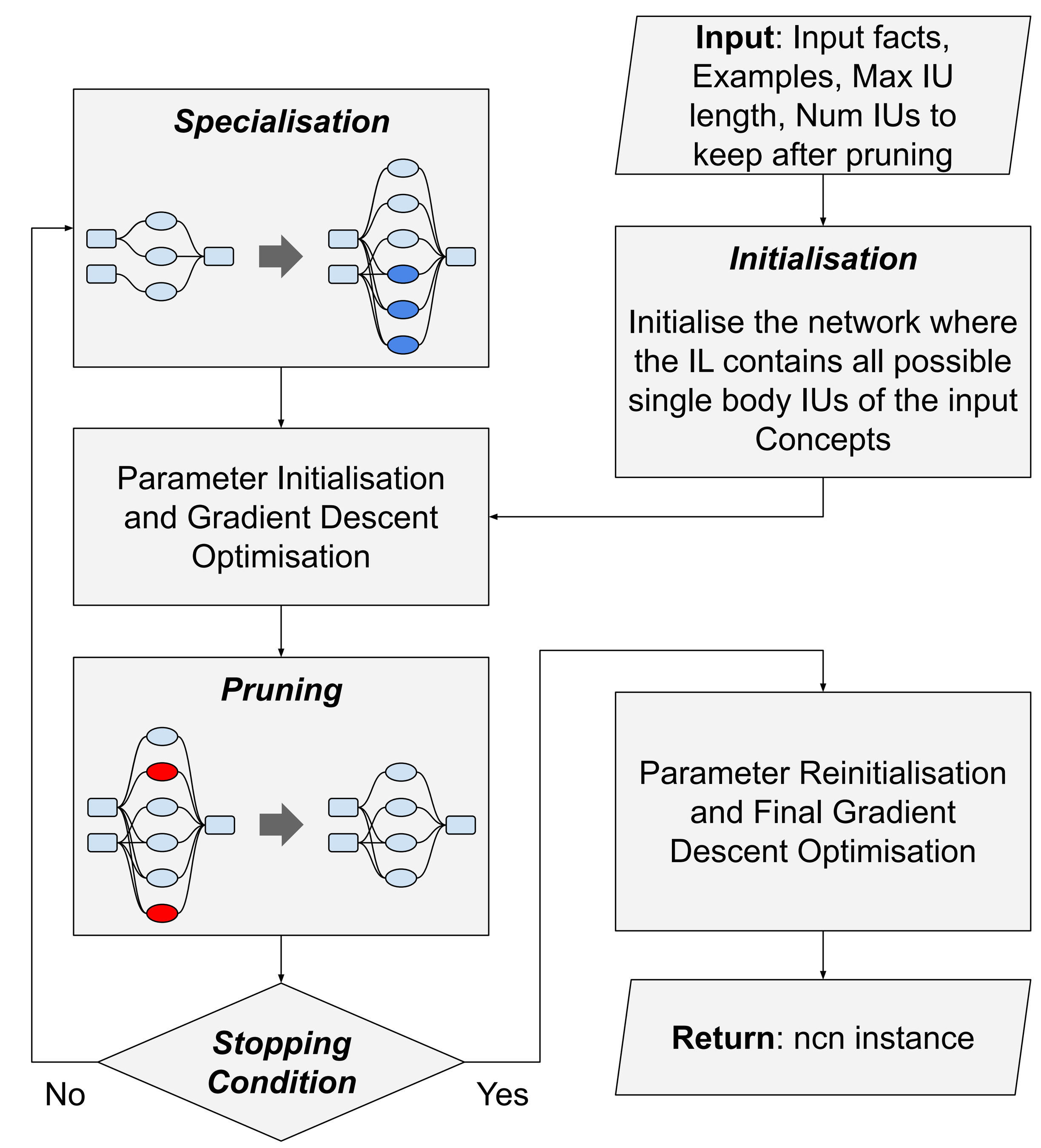
$$\Theta^l_{mqr} = \sum_j \sum_k \left(A^{l-1}_{jk} \odot G^l_{mjkqr} \right) \quad (1)$$

$$\Phi^l_{iq} = f^1 \left(\sum_r \left(W^{(1),l}_{im} \cdot \Theta^l_{mqr} \right) \right) \quad (2)$$

$$A^l_{pq} = f^2 \left(W^{(2),l}_{pi} \cdot \Phi^l_{iq} \right) \quad (3)$$

i — no. of IUs in layer l
j — no. of CUs in layer l - 1
k — no. of unique argument tuples of facts of CUs in layer l - 1
l — layer index
m — total no. of literals in the IL and is equal to $\Sigma_i length(IU^l_i)$
p — no. of output CUs in layer l
q — no. of unique argument tuples of facts of CUs in layer l
r — maximum number of proofs allowed for each output fact

Learning Algorithm



Neural guided structure learning algorithm for a single layer NCN.

Evaluation

We evaluate the approach over two classes of problems, inductive learning and knowledge-base completion, and show, in Table1 and Table3 respectively, that NCN achieves similar or better performance than existing methods for these tasks.

	Task								
System	Predecessor	Son	Grandparent	Husband	Uncle	Father	UE	ATR	TC
δ-ILP [1]	100	100	96.5	100	70	100	100	50.5	95
NTP [2]	100	100	**100**	–	–	100	100	**100**	0
NCN	100	100	**100**	100	**100**	100	100	**100**	**100**

Table 1: Results for inductive learning tasks including Undirected Edge (UE), Adjacent To Red(ATR), and Two Children (TC). The metric used is the percentage of runs that achieve less than 1e-4 mean squared test error.[1]

Dataset	Relations	Train (Positive)	Test	Total Facts	Entities
UMLS	46	5,896	633	6,529	135
Kinship	25	9,586	1,100	10,686	104
WN18	18	146,442	5,000	151,442	40,943
FB15k-237	237	289,650	20,466	310,116	14,541

Table 2: Main features of the knowledge base completion dataset.

Dataset	KBC Metric	Neural-LP [3]	NCN	IKBC Metric	AMIE+ [4]	NCN
UMLS	MRR	**0.733** (0.008)	0.681 (0.020)	Sensitivity	**0.897**	0.805 (0.015)
	Hits@5	**0.877** (0.007)	0.870 (0.019)	Precision	0.022	**0.210** (0.015)
	Hits@10	0.921 (0.005)	**0.951** (0.004)	F1	0.043	**0.333** (0.018)
UMLS (3)	MRR	0.730 (0.009)	**0.860** (0.009)	Sensitivity	**0.946**	0.818 (0.012)
	Hits@5	0.886 (0.013)	**0.956** (0.007)	Precision	0.017	**0.283** (0.011)
	Hits@10	0.930 (0.007)	**0.987** (0.003)	F1	0.034	**0.420** (0.011)
Kinship	MRR	0.612 (0.004)	**0.620** (0.006)	Sensitivity	**0.999**	0.896 (0.004)
	Hits@5	0.789 (0.003)	**0.861** (0.006)	Precision	0.020	**0.197** (0.005)
	Hits@10	0.906 (0.002)	**0.957** (0.003)	F1	0.039	**0.323** (0.007)
Kinship (3)	MRR	0.602 (0.008)	**0.651** (0.006)	Sensitivity	**1.000**	0.880 (0.006)
	Hits@5	0.783 (0.010)	**0.871** (0.007)	Precision	0.015	**0.243** (0.006)
	Hits@10	0.905 (0.004)	**0.961** (0.003)	F1	0.029	**0.381** (0.007)
WN18	MRR	0.944 (0.000)	**0.951** (0.029)	Sensitivity	0.747	**0.749** (0.000)
	Hits@5	0.946 (0.000)	**0.985** (0.008)	Precision	0.427	**0.789** (0.088)
	Hits@10	0.954 (0.000)	**0.992** (0.006)	F1	0.543	**0.766** (0.041)
FB15k-237	MRR	**0.251**	0.246	Sensitivity	0.335	**0.429**
	Hits@5	**0.315**	0.303	Precision	**0.029**	0.003
	Hits@10	0.373	**0.389**	F1	**0.053**	0.006

Table 3: Results for knowledge-base completion tasks. The max rule length allowed is 2, except for UMLS (3) and Kinship (3) where it is set to 3. We observe a significant increase in NCN's performance, as compared to Neural-LP and AMIE+ that appear to struggle particularly for the Kinship dataset, as the max length is increased. Based on the initial evaluation, NCN suffers in FB15k-237 as compared to AMIE+. Hyperparameter optimisation could potentially address NCN's low precision score, and we aim to explore this in the future work.

Future Work

We have presented a novel architecture for iterative relational learning through neural guided search. As part of our future work, we aim to extend the approach in two ways: firstly, to integrate NCN with other differentiable architectures (e.g., CNN) to support end-to-end relational learning from unstructured data (e.g., images); secondly, extending structure learning to multiple layers with the objective of learning latent and interdependent concepts.

References

[1] R. Evans and E. Grefenstette. Learning explanatory rules from noisy data. J. Artif. Int. Res., 61(1):1–64, jan 2018. ISSN 1076-9757.
[2] A. Campero, A. Pareja, T. Klinger, J. Tenenbaum, and S. Riedel. Logical rule induction and theory learning using neural theorem proving. CoRR, abs/1809.02193, 2018.
[3] F. Yang, Z. Yang, and W. W. Cohen. Differentiable learning of logical rules for knowledge base reasoning. In Advances in Neural Information Processing Systems 30, pages 2319–2328, 2017.
[4] L. Galárraga, C. Teflioudi, K. Hose, and F. M. Suchanek. Fast rule mining in ontological knowledge bases with amie+. The VLDB Journal, 24(6):707–730, 2015.