Logic Programs and Neural Networks

Pascal Hitzler

(Joint work with **Anthony Karel Seda**)

Department of Mathematics
University College, Cork, Ireland
{phitzler,aks}@ucc.ie, http://maths.ucc.ie/~{pascal,seda}/

Logic programs and artificial neural networks are two important paradigms in artificial intelligence. Their abilities, and our theoretical understanding of them, however, seem to be rather complementary. Logic programs are inherently discrete, are highly recursive and well understood from the point of view of declarative semantics. Neural networks provide a continuous framework and can be trained, but yet lack a declarative reading.

Building on work by Hölldobler, Kalinke and Störr [2], we address the problem of converting logic programs into neural networks. We show that the immediate consequence operator associated with each logic program, which can be understood as implicitly determining its declarative semantics, can be approximated by 3-layer feedforward neural networks arbitrarily well in a certain measure-theoretic sense. If this operator is continuous in a topology known as the atomic topology, then the approximation is uniform in all points.

References

[1] P. Hitzler and A.K. Seda, A Note on the Relationships between Logic Programs and Neural Networks. In: P. Gibson and D. Sinclair (Eds.), Proceedings of the Fourth Irish Workshop on Formal Methods (IWFM'00), Maynooth, July 2000. Electronic Workshops in Computing (eWiC), British Computer Society, 9 pages. To appear.

[2] S. Hölldobler, H.-P. Störr and Y. Kalinke, Approximating the Semantics of Logic Programs by Recurrent Neural Networks. Applied Intelligence **11**, 1999, 45–58.