# Ontology Learning as a Use-Case for Neural-Symbolic Integration (position paper)

**Pascal Hitzler[1]*, Sebastian Bader[2]†, Artur Garcez[3]**
[1]AIFB, University of Karlsruhe, Germany
[2]International Center for Computational Logic, TU Dresden, Germany
[3]Department of Computing, City University London, UK

## Abstract

We argue that the field of neural-symbolic integration is in need of identifying application scenarios for guiding further research. We furthermore argue that ontology learning — as occuring in the context of semantic technologies — provides such an application scenario with potential for success and high impact on neural-symbolic integration.

## 1 Neural-Symbolic Integration

Intelligent systems based on symbolic knowledge processing, on the one hand, and on artificial neural networks (also called connectionist systems), on the other, differ substantially. Nevertheless, these are both standard approaches to artificial intelligence and it would be very desirable to combine the robustness of neural networks with the expressivity of symbolic knowledge representation. This is the reason why the importance of the efforts to bridge the gap between the connectionist and symbolic paradigms of Artificial Intelligence has been widely recognised. As the amount of hybrid data containing symbolic and statistical elements as well as noise increases in diverse areas such as bioinformatics or text and web mining, neural-symbolic learning and reasoning becomes of particular practical importance. Notwithstanding, this is not an easy task, as illustrated in the sequel.

The merging of theory (background knowledge) and data learning (learning from examples) in neural networks has been indicated to provide learning systems that are more effective than purely symbolic and purely connectionist systems, especially when data are noisy [16]. This has contributed decisively to the growing interest in developing neural-symbolic systems, i.e. hybrid systems based on neural networks that are capable of learning from examples and background knowledge, and of performing reasoning tasks in a massively parallel fashion. Typically, translation algorithms from a symbolic to a connectionist representation and vice-versa are employed to provide either (i) a neural implementation of a logic, (ii) a logical characterization of a neural system, or (iii) a hybrid system that brings together features from connectionism and symbolic Artificial Intelligence.

However, while symbolic knowledge representation is highly recursive and well understood from a declarative point of view, neural networks encode knowledge implicitly in their weights as a result of learning and generalisation from raw data, which are usually characterized by simple feature vectors. While significant theoretical progress has recently been made on knowledge representation and reasoning using neural networks, and on direct processing of symbolic and structured data using neural methods, the integration of neural computation and expressive logics such as first order logic is still in its early stages of methodological development.

Concerning knowledge extraction, we know that neural networks have been applied to a variety of real-world problems (e.g. in bioinformatics, engineering, robotics), and they were particularly successful when data are noisy. But entirely satisfactory methods for extracting symbolic knowledge from such trained networks in terms of accuracy, efficiency, rule comprehensibility, and soundness are still to be found. And problems on the stability and learnability of recursive models currently impose further restrictions on connectionist systems.

In order to advance the state of the art, we believe that it is necessary to look at the biological inspiration for neural-symbolic integration, to use more formal approaches for translating between the connectionist and symbolic paradigms, and to pay more attention to potential application scenarios. We will argue in the following that ontology learning provides such an application scenario with potential for success and high impact.

## 2 The Need for Use Cases

The general motivation for research in the field of neural-symbolic integration (just given) arises from conceptual observations on the complementary nature of symbolic and neural network based artificial intelligence described above. This conceptual perspective is sufficient for justifying the mainly foundations-driven lines of research being undertaken in this area so far. However, it appears that this conceptual approach to the study of neural-symbolic integration has now reached an impasse which requires the identification of use cases and application scenarios in order to drive future research.

*Pascal Hitzler is supported by the German Federal Ministry of Education and Research under the SmartWeb project, and by the European Commission under contract IST-2003-506826 SEKT.

†Sebastian Bader is supported by the GK334 of the German Research Foundation (DFG).

Indeed, the theory of integrated neural-symbolic systems has reached a quite mature state but has not been tested extensively so far on real application data. From the pioneering work by McCulloch and Pitts [27], a number of systems has been developed in the 80s and 90s, including Towell and Shavlik's KBANN [33], Shastri's SHRUTI [31], the work by Pinkas [29], Hölldobler [21], and d'Avila Garcez et al. [15; 17], to mention a few. The reader is referred to [9; 16; 19] for comprehensive literature overviews. These systems, however, have been developed for the study of general principles, and are in general not suitable for real data or application scenarios that go beyond propositional logic. Nevertheless, these studies provide methods which can be exploited for the development of tools for use cases, and significant progress can now only be expected as a continuation of the fundamental research undertaken in the past.

The systems just mentioned — and most of the research on neural-symbolic integration to date — is based on propositional logic or similarly finitistic paradigms. Significantly large and expressible fragments of first order logic are rarely being used because the integration task becomes much harder due to the fact that the underlying language is infinite but shall be encoded using networks with a finite number of nodes [6]. The few approaches known to us to overcome this problem are (a) the work on recursive autoassociative memory, RAAM, initiated by Pollack [30], which concerns the learning of recursive terms over a first-order language, and (b) research based on a proposal by Hölldobler et al. [23], spelled out first for the propositional case in [22], and reported also in [20]. It is based on the idea that logic programs can be represented — at least up to subsumption equivalence [26] — by their associated single-step or immediate consequence operators. Such an operator can then be mapped to a function on the real numbers that can, under certain conditions, in turn be encoded or approximated e.g. by feedforward networks with sigmoidal activation functions using an approximation theorem due to Funahashi [13], and (c) more recently, the idea of fibring neural networks [12], which follows from Gabbay's fibring methodology to combine logical systems [14], has been used to represent acyclic first order logic programs with infinitely many ground instances in a (simple, by comparison) neural network [5].

In addition to these and a number of other sophisticated theoretical results — reported e.g. in [4; 5; 6; 7; 20; 23] —, first-order neural-symbolic integration still remains a widely open issue, where advances are very difficult, and it is very hard to judge to date to what extent the theoretical approaches can work in practice. We argue that the development of use cases with varying levels of expressive complexity is, as a result, needed to drive the development of methods for neural-symbolic integration beyond propositional logic.

## 3 Semantic Technologies and Ontology Learning

With amazing speed, the world wide web has become a widespread means of communication and information sharing. Today, it is an integral part of our society, and will continue to grow. However, most of the information available cannot be processed easily by machines, but has to be read and interpreted by humans. In order to overcome this limitation, a world-wide research effort is currently being undertaken, following the vision put forward by Berners-Lee et al. [8] to make the contents of the world wide web accessible, interpretable, and usable by machines. The resulting extension of the World Wide Web is commonly referred to as the *Semantic Web*, and the underlying technological infrastructure which is currently being developed is referred to as *Semantic Technologies*.

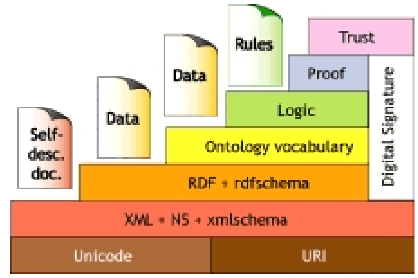


Figure 1: The Semantic Web Layer Cake

In this process, a key idea is that web content should be provided with conceptual background — often referred to as *ontologies* [32] — which allows machines to put information into context, making it interpretable. These research efforts are grouped around the so-called semantic web layer cake, shown in Figure 1; it depicts subsequent layers of functionality and expressiveness, which shall be put in place incrementally. Most recently — having established RDF and RDF-Schema as basic syntax — the OWL Web Ontology Language [2; 28], which is a decidable fragment of first-order logic, has been recommended by the world wide web consortium (W3C) for the ontology vocabulary.

Conceptual knowledge is provided by means of statements in some logical framework, and the discussion concerning suitable logics is still ongoing. Description Logics [3] will most likely play a major role, as they provide the foundation for OWL, but other approaches are also being considered. Currently, the development of an expressive rule-based logic layer on top of OWL for the inference of ontological knowledge is being investigated. But also fragments of OWL, including Horn and propositional languages, are being used, as different application scenarios necessitate different trade-offs between expressiveness, conceptual and computational complexity, and scalability.

The construction of ontologies in whatever language, however, appears as a narrow bottleneck to the proliferation of the Semantic Web and other applications of Semantic Technologies. The success of the Semantic Web and its technologies indeed depends on the rapid and inexpensive development, coordination, and evolution of ontologies. Currently, however, these steps all require cumbersome engineering processes, associated with high costs and heavy time strain on domain experts. It is therefore desirable to automate the ontology creation and ontology refinement process, or at least to provide intelligent ontology learning systems that aid the ontology engineer in this task.

From a bird's eye's view, such a system should be able to handle terms and synonyms, in order to build abstract concepts and concept hierarchies from text-based websites. This basic ontological knowledge then needs to be further refined using relations and rules, in accordance with established or to-be-established standards for ontology representation. Current systems [10; 11; 25] use only very basic ontology languages, but technological advances are expected soon, since the need for expressive ontology languages is generally agreed upon.

## 4 Ontology Learning as Use Case

We argue that ontology learning, as just described, constitutes a highly interesting application area for neural-symbolic integration. As a use case, it appears to be conceptually sound, technically feasible, and of potential high impact. Let us now give our arguments in more detail.

### 4.1 Conceptually Sound

Machine learning methods based on artificial neural networks are known to perform well in the presence of noisy data. If ontologies are to be learned from such uncontrolled data like real existing webpages or other large data repositories, the handling of noise becomes a real issue. At the same time, we can only expect to be able to make reasonable generalizations from data given in the form of html pages if background knowledge is also taken into account. It would be natural for such background knowledge to be ontology-based and therefore symbolic. Furthermore, the required output necessarily has to be in a logic-based format because it will have to be processed by standard tools from the semantic web context. This would require the use of efficient knowledge extraction algorithms to derive compact symbolic representations from large-scale trained neural networks.

It looks as though ontology learning requires the integration of symbolic and neural networks-based approaches, which is provided by the methods developed in the field of neural-symbolic integration. Current results and systems indicate that machine learning of ontologies is a very difficult task, and that the most suitable methods and approaches still remain to be identified. We believe that in the end mixed strategies will have to be used to arrive at practical tools, and due to the above mentioned reasons neural-symbolic learning systems can be expected to play a significant role.

### 4.2 Technically Feasible

The specific nature of ontology research led to the development of a variety of different ontology representation languages, and various further modifications of these. Some of them are depicted in Figure 2. Standardization efforts are successfully being undertaken, but it is to be expected that a number of ontology languages of different logical expressivity will remain in practical use. This diversity is natural due to the different particular needs of application scenarios.

As we have identified earlier, the different levels of expressivity correspond well to the specific requirements on a use case scenario to drive neural-symbolic integration research. Propositional methods can be applied to the learning

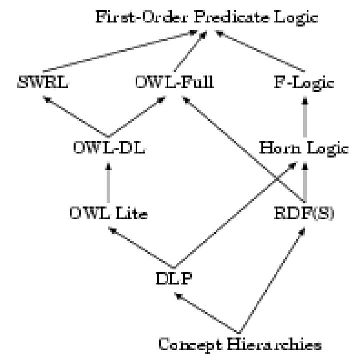


Figure 2: Some ontology languages. Arrows indicate inclusions between the languages. Concept hierarches are simple 'is-a' hierarchies corresponding to certain fragments of propositional logic. The standard OWL [2; 28] already comes in different versions. DLP [18; 34] refers to a weak but practically interesting datalog fragment of OWL. F-Logic [24; 1] provides an alternative ontology paradigm.

of concept hierarchies or DLP ontologies. Decidable fragments such as the different versions of OWL provide more sophisticated challenges without having to tackle the full range of difficulties inherent of first order neural-symbolic integration. As for learning, we also expect that the learning of conceptual knowledge should harmonize naturally with learning paradigms based on Kohonen maps or similar architectures.

### 4.3 High Potential Impact

The learning of ontologies from raw data has been identified as an important topic for the development of Semantic Technologies. These, in turn, are currently migrating into various research and application areas in artificial intelligence and elsewhere, including knowledge management, ambient computing, cognitive systems, bioinformatics, etc. At the same time, ontology learning appears to be a very hard task, and suitable new learning methods are currently being sought. Neural-symbolic integration has the potential for significant contricution to this area and thus to one of the currently prominent streams in computer science.

## 5 Conclusions

We have identified ontology learning as a potential use case for neural-symbolic integration. We believe that this would further neural-symbolic integration as a field, and provide significant contributions to the development of Semantic Technologies.

*Acknowledgement* We are greatful for a number of very interesting and stimulating comments by the anonymous referees, containing substantial further ideas and related thoughts, which we could not incorporate in full in the final version.

## References

[1] Jürgen Angele and Georg Lausen. Ontologies in F-logic. In Staab and Studer [32], pages 29–50.

[2] Grigoris Antoniou and Frank van Harmelen. Web Ontology Language: OWL. In Staab and Studer [32], pages 67–92.

[3] Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2003.

[4] Sebastian Bader and Pascal Hitzler. Logic programs, iterated function systems, and recurrent radial basis function networks. *Journal of Applied Logic*, 2(3):273–300, 2004.

[5] Sebastian Bader, Pascal Hitzler, and Artur S. d'Avila Garcez. Computing first-order logic programs by fibring artificial neural network. In *Proceedings of the 18th International FLAIRS Conference, Clearwater Beach, Florida, May 2005*, 2005. To appear.

[6] Sebastian Bader, Pascal Hitzler, and Steffen Hölldobler. The integration of connectionism and knowledge representation and reasoning as a challenge for artificial intelligence. In L. Li and K.K. Yen, editors, *Proceedings of the Third International Conference on Information, Tokyo, Japan*, pages 22–33. International Information Institute, 2004. ISBN 4-901329-02-2.

[7] Sebastian Bader, Pascal Hitzler, and Andreas Witzel. Integrating first-order logic programs and connectionist systems — a constructive approach. In *Proceedings of the IJCAI-05 Workshop on Neural-Symbolic Learning and Reasoning, NeSy'05, Edinburgh, UK*, 2005. To appear.

[8] Tim Berners-Lee, James Hendler, and Ora Lassila. The semantic web. *Scientific American*, May 2001.

[9] Anthony Browne and Ron Sun. Connectionist inference models. *Neural Networks*, 14(10):1331–1355, 2001.

[10] Philipp Cimiano, Andreas Hotho, and Steffen Staab. Comparing conceptual, partitional and agglomerative clustering for learning taxonomies from text. In *Proceedings of the European Conference on Artificial Intelligence (ECAI'04)*, pages 435–439. IOS Press, 2004.

[11] Philipp Cimiano, Andreas Hotho, and Steffen Staab. Learning concept hierarchies from text using formal concept analysis. *Journal of Artifical Intelligence Research*, 200x. To appear.

[12] Artur S. d'Avila Garcez and Dov M. Gabbay. Fibring neural networks. In *Proceedings of 19th National Conference on Artificial Intelligence AAAI'04*, pages 342–347, San Jose, California, USA, July 2004. AAAI Press.

[13] Ken-Ichi Funahashi. On the approximate realization of continuous mappings by neural networks. *Neural Networks*, 2:183–192, 1989.

[14] Dov M. Gabbay. *Fibring Logics*. Oxford Univesity Press, 1999.

[15] Artur S. d'Avila Garcez, Krysia Broda, and Dov M. Gabbay. Symbolic knowledge extraction from trained neural networks: A sound approach. *Artificial Intelligence*, 125:155–207, 2001.

[16] Artur S. d'Avila Garcez, Krysia B. Broda, and Dov M. Gabbay. *Neural-Symbolic Learning Systems — Foundations and Applications*. Perspectives in Neural Computing. Springer, Berlin, 2002.

[17] Artur S. d'Avila Garcez and Gerson Zaverucha. The connectionist inductive lerarning and logic programming system. *Applied Intelligence, Special Issue on Neural networks and Structured Knowledge*, 11(1):59–77, 1999.

[18] Benjamin Grosof, Ian Horrocks, Raphael Volz, and Stefan Decker. Description logic programs: Combining logic programs with description logics. In *Proc. of WWW 2003, Budapest, Hungary, May 2003*, pages 48–57. ACM, 2003.

[19] Hans W. Güsgen and Steffen Hölldobler. Connectionist inference systems. In Bertram Fronhöfer and Graham Wrightson, editors, *Parallelization in Inference Systems*, volume 590 of *Lecture Notes in Artificial Intelligence*, pages 82–120. Springer, Berlin, 1992.

[20] Pascal Hitzler, Steffen Hölldobler, and Anthony K. Seda. Logic programs and connectionist networks. *Journal of Applied Logic*, 3(2):245–272, 2004.

[21] Steffen Hölldobler. *Automated Inferencing and Connectionist Models*. Fakultät Informatik, Technische Hochschule Darmstadt, 1993. Habilitationsschrift.

[22] Steffen Hölldobler and Yvonne Kalinke. Towards a massively parallel computational model for logic programming. In *Proceedings ECAI94 Workshop on Combining Symbolic and Connectionist Processing*, pages 68–77. ECCAI, 1994.

[23] Steffen Hölldobler, Yvonne Kalinke, and Hans-Peter Störr. Approximating the semantics of logic programs by recurrent neural networks. *Applied Intelligence*, 11:45–58, 1999.

[24] Michael Kifer, Georg Lausen, and James Wu. Logical foundations of object-oriented and frame-based languages. *Journal of the ACM*, 42:741–843, 1995.

[25] Alexander Maedche and Steffen Staab. Ontology learning. In Staab and Studer [32].

[26] Michael J. Maher. Equivalences of logic programs. In Jack Minker, editor, *Foundations of Deductive Databases and Logic Programming*, pages 627–658. Morgan Kaufmann, Los Altos, CA, 1988.

[27] Warren S. McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5:115–133, 1943.

[28] Web ontology language (OWL). www.w3.org/2004/OWL/, 2004.

[29] Gadi Pinkas. Propositional non-monotonic reasoning and inconsistency in symmetric neural networks. In John Mylopoulos and Raymond Reiter, editors, *Proceedings of the 12th International Joint Conference on Artificial Intelligence*, pages 525–530. Morgan Kaufmann, 1991.

[30] Jordan B. Pollack. Recursive distributed representations. *Artificial Intelligence*, 46(1):77–105, 1990.

[31] Lokenda Shastri. Advances in Shruti — A neurally motivated model of relational knowledge representation and rapid inference using temporal synchrony. *Applied Intelligence*, 11:78–108, 1999.

[32] Steffen Staab and Rudi Studer, editors. *Handbook on Ontologies*. International Handbooks on Information Systems. Springer, 2004.

[33] Geoffrey G. Towell and Jude W. Shavlik. Knowledge-based artificial neural networks. *Artificial Intelligence*, 70(1–2):119–165, 1994.

[34] Raphael Volz. *Web Ontology Reasoning with Logic Databases*. PhD thesis, AIFB, University of Karlsruhe, 2004.