Report from Dagstuhl Seminar 25061

Logic and Neural Networks

Vaishak Belle*[1], Michael Benedikt*[2], Dana Drachsler-Cohen*[3], Daniel Neider*[4], and Tom Yuviler†[5]

1 **University of Edinburgh, GB.** vaishak@ed.ac.uk
2 **University of Oxford, GB.** michael.benedikt@cs.ox.ac.uk
3 **Technion, IL.** ddana@ee.technion.ac.il
4 **TU Dortmund University, DE.** daniel.neider@tu-dortmund.de
5 **Technion – Haifa, IL.** tom.yuviler@campus.technion.ac.il

Abstract

Logic and learning are central to Computer Science, and in particular to AI-related research. Already Alan Turing envisioned in his 1950 "Computing Machinery and Intelligence" paper a combination of statistical (ab initio) machine learning and an "unemotional" symbolic language such as logic. The combination of logic and learning has received new impetus from the spectacular success of deep learning systems.

This report documents the program and the outcomes of Dagstuhl Seminar 25061 "Logic and Neural Networks". The goal of this Dagstuhl Seminar was to bring together researchers from various communities related to utilizing logical constraints in deep learning and to create bridges between them via the exchange of ideas. The seminar focused on a set of interrelated topics: enforcement of constraints on neural networks, verifying logical constraints on neural networks, training using logic to supplement traditional supervision, and explanation and approximation via logic. This Dagstuhl Seminar aimed not at studying these areas as separate components, but in exploring common techniques among them as well as connections to other communities in machine learning that share the same broad goals.

The seminar format consisted of long and short talks, as well as breakout sessions. We summarize the motivations and proceedings of the seminar, and report on the abstracts of the talks and the results of the breakout sessions.

Seminar February 02–07, 2025 – https://www.dagstuhl.de/25061

2012 ACM Subject Classification Computing methodologies → Artificial intelligence; Computing methodologies → Machine learning approaches; Computing methodologies → Logical and relational learning; Theory of computation → Constraint and logic programming; Theory of computation → Models of learning; General and reference → Surveys and overviews; Theory of computation → Modal and temporal logics

Keywords and phrases machine learning, learning theory, logic, computational complexity, databases, verification, safety

Digital Object Identifier 10.4230/DagRep.15.2.1

* Editor / Organizer

† Editorial Assistant / Collector

Logic and Neural Networks, *Dagstuhl Reports*, Vol. 15, Issue 2, pp. 1–20
Editors: Vaishak Belle, Michael Benedikt, Dana Drachsler-Cohen, Daniel Neider, and Tom Yuviler
DAGSTUHL REPORTS Dagstuhl Reports
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Executive Summary

Vaishak Belle (University of Edinburgh, GB, vaishak@ed.ac.uk)
Michael Benedikt (University of Oxford, GB, michael.benedikt@cs.ox.ac.uk)
Dana Drachsler-Cohen (Technion, IL, ddana@ee.technion.ac.il)
Daniel Neider (TU Dortmund University, DE, daniel.neider@tu-dortmund.de)

Motivation

Logic and learning are central to Computer Science, and in particular to AI-related research. Already Alan Turing envisioned in his 1950 "Computing Machinery and Intelligence" paper [1] a combination of statistical (ab initio) machine learning and an "unemotional" symbolic language such as logic. The combination of logic and learning has received new impetus from the spectacular success of deep learning systems. As part of these developments, several key roles for logical rules have been identified: As a means of expressing safety properties that a network should satisfy; As a way of providing "weak supervision", that can be utilized in training, to augment or to substitute for direct supervision; As a means of explaining properties of networks, or explanations of the decisions produced by them. With the identification of these roles, a number of core challenges have arisen: Verifying logic-based properties of networks, Enforcing logic-based properties during training; Utilizing logic-based properties in tandem with traditional supervision within learning to train networks; and Producing logic-based explanations of neural network outcomes. Clearly, these challenges have significant synergy between them. The goal of this seminar was to bring together researchers from various communities related to utilizing constraints in deep learning, and to create bridges between them via the exchange of ideas.

Design of the Seminar

The seminar focused on a set of interrelated topics connected to logic and neural networks:

- **Verifying logical constraints on neural networks.** Despite being successful in various tasks, neural networks have also been shown to be susceptible to various attacks (e.g., adversarial attacks [2]) or prone to biased decisions (e.g., in Amazon's systems[1]). To understand the resilience of networks to these phenomena, it is crucial to prove that networks satisfy *safety properties*, such as local robustness and fairness. These are captured via *logical constraints*, defined on specific inputs in a given dataset (e.g., local robustness) or universally on any input (e.g., fairness and global robustness). Many works have proposed verification systems for these properties [3], typically leveraging constraint solvers [4, 5] or static analysis [6, 7]. Constraints can derive from a number of motivations: security/safety, fairness, or interpretability. Despite the active research on verifying these properties, existing approaches still do not scale to very deep networks, which are ubiquitous in practice. We believe it is viable to understand how to push forward the analysis capabilities to use them for large and deep networks. This will have an impact both for academy and industry, since it will increase the users' trust in practical neural network-based systems.

[1] e.g., https://www.reuters.com/article/us-amazon-com-jobs-automation-insight-idUSKCN1MK08G

- **Enforcement of constraints on neural networks.** Logical rules can represent important safety properties and prior knowledge into the training of neural networks. For example, in a manufacturing setting, we may wish to encode that an actuator for a robotic arm does not exceed some threshold (e.g., causing the arm to move at a hazardous speed). Another example is a self-driving car, where a controller should be known to operate within a predefined set of constraints (e.g., the car should always stop completely when facing a human). In such safety-critical domains, machine learning solutions must guarantee to operate within distinct boundaries that are specified by logical, arithmetic, and geometric rules. Techniques include specialized loss functions [8], which can be augmented with additional layers within a neural architecture. These approaches compile the constraints into the loss function of the training algorithm, by quantifying the extent to which the output of the network violates the constraints. This is appealing as logical constraints are easy to elicit from people. However, the solution outputted by the network is designed to minimize the loss function – which combines both data and constraints – rather than to guarantee the satisfaction of the domain constraints. So this is an important open problem.
- **Training using logic and traditional supervision.** A major of impetus for a synthesis of logic and learning relates to paucity of supervision. In many regimes explicit supervision is extremely limited, and synthetic data generation may be infeasible. A promising approach to augment supervision is via the use of external knowledge. The approach has been used in domains as distinct as scene recognition [9] and parsing [10]. Approaches that integrate constraint-based supervision with traditional supervision have arisen simultaneously in many areas of artificial intelligence. While the focus of our seminar is constraints expressed in general-purpose logics, we look for connections with constraint-based approaches to learning from other areas, such as physics.
- **Explaining neural networks via logic.** A critical issue with black box models, particularly neural networks, is understanding their decision boundaries. An important strategy employed in recent years involves attempting to extract decision trees, logical rules, and other deterministic machines from these neural networks [11, 12, 13, 14]. This can be seen as a strategy for post-hoc explanation [15]. Most approaches for rule extraction use template-based approaches to explore patterns in pre-trained models, with a focus on characteristics and properties of entities such as people, places, or things. However, template-based approaches do show sensitivity to template formulation, highlighting the need to explore alternative strategies to probe pre-trained models. They are often based on a combination of techniques from Bayesian Structure Learning, Inductive Logic Programming [16], and Distillation [15]. Explanation and approximation via logics have also arisen in Graph Neural Networks [17]. An interesting phenomenon is that one of the languages used for explanation is Datalog, which is also prominent in the verification community. The ability to approximate networks by logics is closely-related to attempts to understand the expressiveness of neural approaches in terms of logics [18, 19].

Summary of Seminar Activities

The seminar was attended by 38 researchers across various communities including logic, formal verification, machine learning, deep learning, program synthesis, graph neural networks, expressiveness, explainability, theorem proving, neural-symbolic learning, and databases. The seminar participants included senior and junior researchers, including graduate students, postdoctoral researchers, faculty members, and industry experts. The seminar was conducted through talks and breakout sessions, with breaks for discussion between the attendees. Overall, there were 19 talks, and two main breakout sessions. The talks included a range of presentations on recent advances in the interrelated fields of logic and neural networks, as previously discussed. Some talks also provided broader overviews of related areas, such as formal verification of neural networks and program synthesis. The first breakout session was divided into four groups based on the participants' main areas of research: verification, expressivity, explainability, and learning with background knowledge and constraints. Each group discussed several topic-specific questions: (1) the open challenges, (2) the value proposition, (3) potential "killer applications" or teaching curricula, and (4) drafting a concise manifesto. The second breakout session was divided into three groups (based on participants' choices), each focused on integrating interrelated topics: (1) verification and constraints, (2) explainability, expressiveness, and constraints, and (3) verification and explainability. Each group examined several issues concerning the interplay of these areas, including: (1) prior work, (2) open challenges, (3) real-world motivations and applications, and (4) short- and long-term project ideas.

Conclusion

We consider the seminar a success and believe it achieved several goals that will help strengthen connections among the fields of neural-network verification, logic, explainability, and expressivity. These include: (1) fostering links among the participating researchers, (2) generating a set of open challenges, goals, and future research directions, and (3) providing a more unified view of current approaches to these interrelated topics. We also hope the seminar will catalyze the further development of benchmarks for applying logic in neural networks. Finally, the seminar's format – featuring talks, ample time for discussion, and breakout sessions – received positive feedback from participants.

References

1 A. M. Turing: Computing machinery and intelligence. In Mind, vol. LIX (1950)

2 Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, Rob Fergus: Intriguing properties of neural networks. In ICLR (2014).

3 Linyi Li, Tao Xie, Bo Li. SoK: Certified robustness for deep neural networks. In IEEE Symposium on Security and Privacy (2023).

4 Guy Katz, Clark W. Barrett, David L. Dill, Kyle Julian, Mykel J. Kochenderfer: Reluplex: a calculus for reasoning about deep neural networks. In Formal Methods Syst. Des. (2022).

5 Vincent Tjeng, Kai Yuanqing Xiao, Russ Tedrake: Evaluating robustness of neural networks with mixed integer programming. In ICLR (2019).

6 Timon Gehr, Matthew Mirman, Dana Drachsler-Cohen, Petar Tsankov, Swarat Chaudhuri, Martin T. Vechev: AI2: safety and robustness certification of neural networks with abstract interpretation. In IEEE Symposium on Security and Privacy (2018).

7 Huan Zhang, Tsui-Wei Weng, Pin-Yu Chen, Cho-Jui Hsieh, Luca Daniel: Efficient neural network robustness certification with general activation functions. In NeurIPS (2018).

8 Jingyi Xu, Zilu Zhang, Tal Friedman, Yitao Liang, Guy Van den Broeck: A semantic loss function for deep learning with symbolic knowledge. In ICML (2018).

9 Eleonora Giunchiglia, Mihaela Catalina Stoian, Salman Khan, Fabio Cuzzolin, Thomas Lukasiewicz: ROAD-R: the autonomous driving dataset with logical requirements. In Mach. Learn. (2023).

10 Chen Liang, Jonathan Berant, Quoc V. Le, Ken Forbus, Ni Lao. Neural symbolic machines: Learning semantic parsers on freebase with weak supervision. In ACL (2017).

11 Jan Ruben Zilke, Eneldo Loza Mencía, Frederik Janssen: Deepred–rule extraction from deep neural networks. In DS (2016).

12 Robert Andrews, Joachim Diederich, Alan B Tickle: Survey and critique of techniques for extracting rules from trained artificial neural networks. In Knowledge-based systems (1995).

13 Tameru Hailesilassie: Rule extraction algorithm for deep neural networks: A review. In CoRR abs/1610.05267 (2016).

14 Hiroshi Tsukimoto: Extracting rules from trained neural networks. In IEEE Transactions on Neural networks (2000).

15 Vaishak Belle, Ioannis Papantonis. Principles and practice of explainable machine learning. In Frontiers in big Data (2021).

16 Stephen Muggleton, Luc De Raedt, David Poole, Ivan Bratko, Peter Flach, Katsumi Inoue, Ashwin Srinivasan: ILP turns 20. In Machine learning (2012).

17 David Tena Cucala, Bernardo Cuenca Grau, Boris Motik, and Egor V. Kostylev: On the correspondence between monotonic max-sum gnns and datalog. In KR (2023).

18 Floris Geerts, Juan L. Reutter: Expressiveness and approximation properties of graph neural networks. In ICLR (2022).

19 David Chiang, Peter Cholak, Anand Pillay: Tighter bounds on the expressivity of transformer encoders. In ICML (2023).

2 Table of Contents

Executive Summary

Overview of Talks

Breakout Sessions

3 Overview of Talks

3.1 Learning Symmetric Rules with SATNet

Hongseok Yang (KAIST – Daejeon, KR, hongseok00@gmail.com)

Joint work of Hongseok Yang, Sangho Lim, Eungyeol Oh
Main reference Sangho Lim, Eun-Gyeol Oh, Hongseok Yang: "Learning Symmetric Rules with SATNet", in Proc. of the Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 – December 9, 2022, 2022.
URL http://papers.nips.cc/paper_files/paper/2022/hash/5642b9811a9ac5281be1cc84c275f251-Abstract-Conference.html

SATNet is a differentiable constraint solver with a custom backpropagation algorithm, which can be used as a layer in a deep-learning system. It is a promising proposal for bridging deep learning and logical reasoning. In fact, SATNet has been successfully applied to learn, among others, the rules of a complex logical puzzle, such as Sudoku, just from input and output pairs where inputs are given as images. In this talk, I explain our work on improving the learning of SATNet by exploiting symmetries in the target rules of a given but unknown logical puzzle or more generally a logical formula. I present SymSATNet, a variant of SATNet that translates the given symmetries of the target rules to a condition on the parameters of SATNet and requires that the parameters should have a particular parametric form that guarantees the condition. The requirement dramatically reduces the number of parameters to learn for the rules with enough symmetries, and makes the parameter learning of SymSATNet much easier than that of SATNet. I also describe a technique for automatically discovering symmetries of the target rules from examples. Our experiments with Sudoku and Rubik's cube show the substantial improvement of SymSATNet over the baseline SATNet.

3.2 Query Languages for Machine Learning Models

Pablo Barcelo (PUC – Santiago de Chile, CL, pbarcelo@uc.cl)

Emerging challenges in machine learning (ML), such as explainability and verification, underscore the growing need for declarative query languages that enable users to extract relevant information from ML models and adapt it to diverse application-specific requirements. These query languages offer several advantages: they provide flexibility in information extraction, establish clear syntax and semantics for queries, and pave the way for query optimization. In this talk, we survey two recent proposals for query languages tailored to ML models – one designed for discrete classification models and another for real-valued models. We demonstrate how these languages can express meaningful queries over ML models, and we analyze their expressiveness and evaluation complexity. Our goal is to foster a productive discussion on advancing the development of practical query languages for ML models that can be effectively applied across a wide range of scenarios.

3.3 How to make logics neurosymbolic

Luc De Raedt (KU Leuven, BE, luc.deraedt@kuleuven.be)

Neurosymbolic AI (NeSy) is regarded as the third wave in AI. It aims at combining knowledge representation and reasoning with neural networks. Numerous approaches to NeSy are being developed and there exists an 'alphabet-soup' of different systems, whose relationships are often unclear. I discuss the state-of-the art in NeSy and argue that there are many similarities with statistical relational AI (StarAI). Taking inspiring from StarAI, and exploiting these similarities, I argue that Neurosymbolic AI = Logic + Probability + Neural Networks. I also provide a recipe for developing NeSy approaches: start from a logic, add a probabilistic interpretation, and then turn neural networks into 'neural predicates'. Probability is interpreted broadly here, and is necessary to provide a quantitative and differentiable component to the logic. At the semantic and the computation level, one can then combine logical circuits (aka proof structures) labelled with probability, and neural networks in computation graphs. I illustrate the recipe with NeSy systems such as DeepProbLog, a deep probabilistic extension of Prolog, and DeepStochLog, a neural network extension of stochastic definite clause grammars (or stochastic logic programs).

3.4 Bridging Generalization and Expressivity of Graph Neural Networks

Floris Geerts (University of Antwerp, BE, floris.geerts@uantwerp.be)

Joint work of Shouheng Li, Floris Geerts, Dongwoo Kim, Qing Wang
Main reference Shouheng Li, Floris Geerts, Dongwoo Kim, Qing Wang: "Towards Bridging Generalization and Expressivity of Graph Neural Networks", in Proc. of the The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025, OpenReview.net, 2025.
URL https://openreview.net/forum?id=BOQpRtI4F5

The expressive power of graph neural networks (GNNs) has been widely analysed through their connection to the 1-dimensional Weisfeiler–Leman (1-WL) algorithm, a key tool for addressing the graph isomorphism problem. While this link has deepened our understanding of how GNNs represent complex structures, it provides limited insight into their generalisation – specifically, their ability to accurately predict on unseen data. In this talk, we delve into the relationship between GNNs' expressive power and their generalisation capabilities, offering a perspective that bridges these two critical aspects of GNN performance.

3.5 Formal Verification of Machine Learning with the Industry: The Journey so Far, And the Future Ahead

Julien Girard-Satabin (CEA de Saclay – Gif-sur-Yvette, FR, julien.girard2@cea.fr)

Joint work of Julien Girard-Satabin, Augustin Lemesle, Julien Lehmann, Tristan Le Gall
Main reference Augustin Lemesle, Julien Lehmann, Tristan Le Gall: "Neural Network Verification with PyRAT", CoRR, Vol. abs/2410.23903, 2024.
URL https://doi.org/10.48550/ARXIV.2410.23903

Since the third AI revolution in 2012, industry displayed a keen interest in the newfound capabilities of machine learning. However, in the field of critical systems, existing regulations and practices require some degree of formal specification (and verification). Furthermore, machine learning specification is implicitly defined by hyperparameters that are impossible to formalise (the dataset, the architecture, the objective function, the intended goal). To address those newfound challenges and fulfill its mission to support industrial actors, the French Atomic Energy Commission develop and maintain several tools for the specification and verification of machine learning systems. For seven years, those tools were applied in industrial settings, in national and international projects. Through this presentation mixing science and technical retrospective, we present the successes, the limitations and potential future paths for formal verification informed by the needs of the French industry.

3.6 Learning with Constraints: Fuzzy Methods

Eleonora Giunchiglia (Imperial College London, GB, e.giunchiglia@imperial.ac.uk)

Joint work of Eleonora Giunchiglia, Mihaela Catalina Stoian, Thomas Lukasiewicz
Main reference Eleonora Giunchiglia, Mihaela Catalina Stoian, Thomas Lukasiewicz: "Deep Learning with Logical Constraints", in Proc. of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022, Vienna, Austria, 23-29 July 2022, pp. 5478–5485, ijcai.org, 2022.
URL https://doi.org/10.24963/IJCAI.2022/767

In this first segment of the tutorial I discuss methods based on fuzzy logic for learning with constraints. In this talk, I first provide an overview of the learning tasks where logical constraints can play a fundamental role. Then I introduce the most commonly used triangular norms, i.e., Gödel, Product and Lukasiewicz, describing their properties. This be followed by the introduction of "Logic Tensor Network" (LTN), which is one of the most famous methods to integrate constraints in neural networks' loss functions and "Coherent-by-Construction Network" (CCN+), a method to integrate constraints in a neural layer. Both methods are based on triangular norms. After this overview, I also discuss how – thanks to the versatility of fuzzy logic – we can now build neural layers integrating constraints as expressive as disjunctions over linear inequalities, which hence model non-convex and disconnected spaces. I conclude the talk with a discussion with the pros and cons of using fuzzy methods in learning with constraints.

3.7 Neural Continuous-Time Supermartingale Certificates

Anna Lukina (TU Delft, NL, A.Lukina@tudelft.nl)

Joint work of Grigory Neustroev, Mirco Giacobbe, Anna Lukina
Main reference Grigory Neustroev, Mirco Giacobbe, Anna Lukina: "Neural Continuous-Time Supermartingale Certificates", in Proc. of the AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence, February 25 – March 4, 2025, Philadelphia, PA, USA, pp. 27538–27546, AAAI Press, 2025.
URL https://doi.org/10.1609/AAAI.V39I26.34966

We introduce for the first time a neural-certificate framework for continuous-time stochastic dynamical systems. Autonomous learning systems in the physical world demand continuous-time reasoning, yet existing learnable certificates for probabilistic verification assume discretization of the time continuum. Inspired by the success of training neural Lyapunov certificates for deterministic continuous-time systems and neural supermartingale certificates for stochastic discrete-time systems, we propose a framework that bridges the gap between continuous-time and probabilistic neural certification for dynamical systems under complex requirements. Our method combines machine learning and symbolic reasoning to produce formally certified bounds on the probabilities that a nonlinear system satisfies specifications of reachability, avoidance, and persistence. We present both the theoretical justification and the algorithmic implementation of our framework and showcase its efficacy on popular benchmarks.

3.8 Challenges for the Certification of AI in Railway Systems

Pierre-Jean Meyer (Gustave Eiffel University – Villeneuve d'Ascq, FR, pierre-jean.meyer@univ-eiffel.fr)

The trend of AI and desire to develop autonomous rail vehicles has led to a surge of interest for the use of AI in the railway field, including in safety-critical functions. Traditionally in the railway field, formal methods have been strongly recommended for the certification of safety-related components, but currently applied approaches cannot be properly adapted for the certification of AI functions. This talk gives a brief overview of current and desired applications of AI in railway field, as well as the main identified challenges for the use of formal verification to certify the good behaviors of AI functions within safety-related modules in autonomous trains: primarily the computational complexity and the definition of formal specifications.

3.9 Distinguished In Uniform: Self Attention Vs. Virtual Nodes

Martin Ritzert (Universität Göttingen, DE, ritzert@informatik.uni-goettingen.de)

Joint work of Eran Rosenbluth, Jan Tönshoff, Martin Ritzert, Berke Kisin, Martin Grohe
Main reference Eran Rosenbluth, Jan Tönshoff, Martin Ritzert, Berke Kisin, Martin Grohe: Distinguished In Uniform: Self-Attention Vs. Virtual Nodes. In ICLR (2024).
URL https://openreview.net/forum?id=AcSChDWL6V

Graph Transformers (GTs) such as SAN and GPS are graph processing models that combine Message-Passing GNNs (MPGNNs) with global Self-Attention. They were shown to be universal function approximators, with two reservations: 1. The initial node features must be augmented with certain positional encodings. 2. The approximation is non-uniform: Graphs of different sizes may require a different approximating network. We first clarify that this form of universality is not unique to GTs: Using the same positional encodings, also pure MPGNNs and even 2-layer MLPs are non-uniform universal approximators. We then consider uniform expressivity: The target function is to be approximated by a single network for graphs of all sizes. There, we compare GTs to the more efficient MPGNN + Virtual Node architecture. The essential difference between the two model definitions is in their global computation method – Self-Attention Vs. Virtual Node. We prove that none of the models is a uniform-universal approximator, before proving our main result: Neither model's uniform expressivity subsumes the other's. We demonstrate the theory with experiments on synthetic data. We further augment our study with real-world datasets, observing mixed results which indicate no clear ranking in practice as well.

3.10 How Can Formal Methods Benefit Large Language Models

Gagandeep Singh (University of Illinois – Urbana-Champaign, US, ggnds@illinois.edu)

Joint work of Isha Chaudhary, Qian Hu, Manoj Kumar, Morteza Ziyadi, Rahul Gupta, Gagandeep Singh
Main reference Isha Chaudhary, Qian Hu, Manoj Kumar, Morteza Ziyadi, Rahul Gupta, Gagandeep Singh: "Certifying Counterfactual Bias in LLMs", in Proc. of the The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025, OpenReview.net, 2025.
URL https://openreview.net/forum?id=HQHnhVQznF

Despite impressive performance, state-of-the-art Large Language Models (LLMs) often hallucinate, produce toxic responses, and leak sensitive information. While increasing model sizes, using more training data, compute resources, and prompt engineering have some marginal impact on LLM behavior, these ad-hoc methods do not solve the core problems. Further. These solutions are unsustainable due to their huge environmental impact. In this talk, I discuss how formal methods can be leveraged to develop principled and systematic approaches to improve LLM performance and alignment, offering a path forward that is both effective and sustainable.

3.11 Program Synthesis Present and Future

Armando Solar-Lezama (MIT – Cambridge, US, asolar@csail.mit.edu)

Large Language Models have transformed the landscape of program synthesis, enabling us to solve previously intractable problems and opening up new applications. In this talk I give a high-level summary of the current state of the art in program synthesis and describe some of the open problems and opportunities in the field.

3.12 Refining Deep Generative Modelling using Background Knowledge

Mihaela Stoian (University of Oxford, GB, mihaela.stoian@cs.ox.ac.uk)

Joint work of Mihaela C. Stoian, Salijona Dyrmishi, Maxime Cordy, Thomas Lukasiewicz, Eleonora Giunchiglia
Main reference Mihaela C. Stoian, Salijona Dyrmishi, Maxime Cordy, Thomas Lukasiewicz, Eleonora Giunchiglia: "How Realistic Is Your Synthetic Data? Constraining Deep Generative Models for Tabular Data", in Proc. of the The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024, OpenReview.net, 2024.
URL https://openreview.net/forum?id=tBROYsEz9G

Synthesising realistic tabular data often relies on deep generative models. However, these models fail to account for inherent relationships between features, encoded as background knowledge, which synthetic samples must satisfy to be deemed realistic. Existing methods handle non-compliant samples by discarding them, leading to potentially indefinite inference times. In this talk, I present a novel approach that embeds a constraint layer into the topology of deep generative models to account for the relationships between the features. This layer automatically incorporates background knowledge and ensures compliance with these constraints during both training and inference. I first present our method for handling linear constraints and then discuss its extension to support quantifier-free linear real arithmetic constraints. Experimental results show that our layer significantly improves the machine learning efficacy of deep generative models without hindering sample generation times. This framework is part of our broader goal of bringing neuro-symbolic AI onto the stage of real-world applications.

3.13 Expressive Power of Graph Neural Networks via Datalog

David Tena Cucala (Royal Holloway, University of London, GB, David.TenaCucala@rhul.ac.uk)

This talk discusses recent results on the expressive power of Graph Neural Networks (GNNs) operating on relational datasets. We consider two sub-classes of GNNs: monotonic GNNs and max GNNs, and then we identify Datalog fragments or extensions that realize the same transformations as these GNNs. Monotonic GNNs are GNNs subject to restrictions ensuring that their behaviour is monotonic under homomorphisms applied to their input. Max GNNs are subject to the restriction that they use the max function to aggregate information. Finally, we illustrate some applications of these results, in the areas of GNN verification and explanation of predictions.

3.14 Static Analysis Methods for Neural Networks

Caterina Urban (INRIA and ENS Paris, FR, caterina.urban@inria.fr)

Joint work of Caterina Urban, Maria Christakis, Valentin Wüstholz, Fuyuan Zhang
Main reference Caterina Urban, Maria Christakis, Valentin Wüstholz, Fuyuan Zhang: "Perfectly parallel fairness certification of neural networks", Proc. ACM Program. Lang., Vol. 4(OOPSLA), pp. 185:1–185:30, 2020.
URL https://doi.org/10.1145/3428253

Formal methods provide rigorous guarantees of correctness for both hardware and software systems. Their use is well established in industry, notably to certify safety of critical applications subject to stringent certification processes. With the rising prominence of machine learning, the integration of machine-learned components into critical systems presents novel challenges for the soundness, precision, and scalability of formal methods. This talk serves as an introduction to formal methods tailed for machine learning software, with a focus on static analysis methods for neural networks. We present several verification approaches, highlighting their strengths and limitations, through the lens of different (hyper)safety properties. A neural network surrogate from a real-world avionics use case serves as a running example. We additionally survey the application of these verification approaches towards the additional goal of enhancing machine learning explainability. We conclude with perspectives on possible future research directions in this rapidly evolving field.

3.15 From Learning with Constraints to Partial Label Learning

Zsolt Zombori (Alfréd Rényi Institute of Mathematics – Budapest, HU, zombori@renyi.hu)

Joint work of Zsolt Zombori, Agapi Rissaki, Kristóf Szabó, Wolfgang Gatterbauer, Michael Benedikt
Main reference Zsolt Zombori, Agapi Rissaki, Kristóf Szabó, Wolfgang Gatterbauer, Michael Benedikt: "Towards Unbiased Exploration in Partial Label Learning", CoRR, Vol. abs/2307.00465, 2023.
URL https://doi.org/10.48550/ARXIV.2307.00465

In numerous learning setups, some background knowledge is available in the form of logical constraints. Such constraints can be useful both for increasing the safety of the trained models and for alleviating data shortage by making learning more effective. In this talk we review different types of constraints and how they can possibly be incorporated into the learning or inference process. We also identify a bias phenomenon that occurs during gradient descent based optimisation with constraints, preventing proper exploration of alternative options and making the dynamics of gradient descent overly sensitive to initialisation. We introduce a novel loss function that allows for unbiased exploration within the space of alternative outputs.

3.16 Learning with Constraints: Probabilistic Methods

Emile van Krieken (University of Edinburgh, GB, Emile.van.Krieken@ed.ac.uk)

Joint work of Emile van Krieken, Thiviyan Thanapalasingam, Jakub M. Tomczak, Frank van Harmelen, Annette ten Teije
Main reference Emile van Krieken, Thiviyan Thanapalasingam, Jakub M. Tomczak, Frank van Harmelen, Annette ten Teije: "A-NeSI: A Scalable Approximate Method for Probabilistic Neurosymbolic Inference", in Proc. of the Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 – 16, 2023, 2023.
URL http://papers.nips.cc/paper_files/paper/2023/hash/4d9944ab3330fe6af8efb9260aa9f307-Abstract-Conference.html

I discuss probabilistic methods for learning with constraints. First, I recap practical issues with fuzzy methods. Then, I introduce the weighted model count (WMC), the central equation underlying probabilistic methods for integrating constraints. The WMC gives many theoretical guarantees. With the WMC at hand, I describe a popular constraint loss method called "Semantic Loss", and a constraint layer called "Semantic Probabilistic Layers". This part ends with a comparison of the strengths and weaknesses of probabilistic and fuzzy methods. After this introduction to the core methods, I describe several issues with constraint losses, starting with Reasoning Shortcuts. This is the phenomenon that models may completely minimise the constraint training loss without learning underlying concepts. I also discuss issues with a conditional independence assumption that is frequently taken in practical setups. I end with a brief introduction of state-of-the-art methods for tackling these issues, and a recap of the methods discussed in this two-part tutorial.

4 Breakout Sessions

4.1 Verification

Vaishak Belle (University of Edinburgh, GB, vaishak@ed.ac.uk)
Michael Benedikt (University of Oxford, GB, michael.benedikt@cs.ox.ac.uk)
Dana Drachsler-Cohen (Technion, IL, ddana@ee.technion.ac.il)
Daniel Neider (TU Dortmund University, DE, daniel.neider@tu-dortmund.de)

Joint work of All participants

Formal verification of ML is currently overly focusing on a very specific set of properties, whose real-world applicability may not be fully correlated with the amount of work poured into it. The community must extend towards the ML community and regulators to provide expressive, sound tools that help better characterize complex systems (for instance, multiple NNs or complex constraints on data) with expressive languages and principled compilation toward provers. For this endeavor to be realized, languages should be accessible to non-experts (possibly through constrained means). Furthermore, verifiers should scale to realistic settings, and creative ways to devise specifications should be pursued, for instance by synthesizing properties.

4.2 Expressivity

Vaishak Belle (University of Edinburgh, GB, vaishak@ed.ac.uk)
Michael Benedikt (University of Oxford, GB, michael.benedikt@cs.ox.ac.uk)
Dana Drachsler-Cohen (Technion, IL, ddana@ee.technion.ac.il)
Daniel Neider (TU Dortmund University, DE, daniel.neider@tu-dortmund.de)

Joint work of All participants

Research on expressivity in advanced ML architectures highlights several open challenges, such as how Transformers handle compositions and whether components like positional encodings are truly necessary. The community must develop principled logical frameworks that clarify which expressions are learnable, while balancing "succinctness" and expressivity so that models remain trainable in realistic settings. Logical upper and lower bounds can guide the design of new architectures and help prevent unintended behaviors (e.g., through constrained losses or temporal constraints). Success stories such as the Weisfeiler-Lehman (WL) approach in graph learning show the value of bridging logical theory and ML practice, though some models (like k-WL) have proven impractical. Looking ahead, we should refine these analyses for GNNs, consider how different architectural features shape learning, and pursue sound yet accessible methods that integrate logic and machine learning across diverse applications.

4.3 Explainability

Vaishak Belle (University of Edinburgh, GB, vaishak@ed.ac.uk)
Michael Benedikt (University of Oxford, GB, michael.benedikt@cs.ox.ac.uk)
Dana Drachsler-Cohen (Technion, IL, ddana@ee.technion.ac.il)
Daniel Neider (TU Dortmund University, DE, daniel.neider@tu-dortmund.de)

Joint work of All participants

Research on explainability in ML highlights challenging trade-offs between model performance and interpretability, particularly in high-stakes domains where trust and transparency are paramount. While methods such as SHAP or LIME offer partial insights, the community still grapples with fundamental questions about how to align expert understanding with possibly less accurate yet more transparent models. Practical benefits extend beyond improved decision-making: interpretable systems can foster scientific discovery by exposing the reasoning behind model predictions, enabling knowledge transfer across tasks, and ensuring that ethical constraints are thoroughly verified. Ultimately, progress in explainability hinges on identifying scenarios where transparent models demonstrably outperform black-box approaches, attracting broader funding and community engagement, and integrating reverse-engineerable explanations that help pinpoint out-of-distribution cases and other critical failures.

4.4 Learning with Background Knowledge & Constraints

Vaishak Belle (University of Edinburgh, GB, vaishak@ed.ac.uk)
Michael Benedikt (University of Oxford, GB, michael.benedikt@cs.ox.ac.uk)
Dana Drachsler-Cohen (Technion, IL, ddana@ee.technion.ac.il)
Daniel Neider (TU Dortmund University, DE, daniel.neider@tu-dortmund.de)

Joint work of All participants

There are several key challenges and considerations in integrating constraints into AI systems, particularly in enhancing performance across various metrics such as safety, data efficiency, accuracy, model size, and generalizability. One of the main challenges identified is demonstrating measurable improvements in real-world scenarios. The discussion also emphasized the need for parametric synthetic datasets with controllable properties, particularly for out-of-distribution (OOD) testing. Additionally, the topic of constraint discovery is highlighted, exploring how constraints can be learned and analyzed in terms of their expressivity, complexity, and geometric properties. The value proposition centers on the advantages of incorporating constraints into AI models beyond just improving accuracy. These advantages include ensuring safer AI decisions, reducing the need for labeled data, enabling more compact models, and improving robustness. The discussion extends beyond traditional models to generative AI, emphasizing that constraints should not only enforce syntactic correctness but also contribute to semantic understanding. We also outline potential teaching material for conveying these neuro-symbolic (NeSy) concepts. The curriculum would begin with a general motivation for NeSy, explaining the complementary strengths of symbolic and statistical approaches. It would introduce key ingredients, including logic (e.g., knowledge graphs, description logics, and logic programming), probabilistic methods, fuzzy logic, neural predicates, and knowledge compilation. The discussion would then cover how these elements integrate into different architectures, addressing aspects such as layers, loss functions, and predicate grounding. A key theme underlying NeSy is encapsulated in the phrase "Why learn what you already know?", suggesting that constraints should guide AI systems by leveraging prior knowledge efficiently.

4.5 Combining Verification and Constraints

Vaishak Belle (University of Edinburgh, GB, vaishak@ed.ac.uk)
Michael Benedikt (University of Oxford, GB, michael.benedikt@cs.ox.ac.uk)
Dana Drachsler-Cohen (Technion, IL, ddana@ee.technion.ac.il)
Daniel Neider (TU Dortmund University, DE, daniel.neider@tu-dortmund.de)

Joint work of All participants

Research on combining constraints and verification in ML underscores the need to ensure that critical requirements remain satisfied, especially when models are treated as black boxes. While constraints can guide the design of more easily verifiable networks – by reducing complexity or limiting nonlinearities – they often need verification to confirm that these properties hold in practice. In turn, verification methods benefit from constraints by narrowing the solution space or allowing for surrogate models that can more efficiently detect potential errors. Challenges persist in communicating across different communities (e.g., security experts operating in black/grey-box settings), devising effective regularizations that maintain performance while improving verifiability, and tackling relational constraints that are notoriously difficult to encode directly. Achieving progress in these areas will require deeper collaboration and possibly new architectures, loss functions, or partitioning strategies that streamline verification while preserving robust performance.

4.6 Combining Explainability, Expressiveness, and Constraints

Vaishak Belle (University of Edinburgh, GB, vaishak@ed.ac.uk)
Michael Benedikt (University of Oxford, GB, michael.benedikt@cs.ox.ac.uk)
Dana Drachsler-Cohen (Technion, IL, ddana@ee.technion.ac.il)
Daniel Neider (TU Dortmund University, DE, daniel.neider@tu-dortmund.de)

Joint work of All participants

Negative results on expressiveness show that logical formulas can be difficult to apply for direct explainability. Traditionally, constraints are motivated by safety rather than explainability, but recent approaches use logical constraints to explain network internals – such as analyzing neuron correlations when given specific images. This strategy could simplify various explainability tasks by leveraging a suitable logical framework, potentially informed by knowledge representation techniques. Several open challenges include determining whether expressive architectures inherently complicate constraint enforcement, identifying parameter sets that minimize constraint violations, and discovering methods to isolate network components corresponding to specific constraints. Addressing these issues could lead to more transparent decision-making grounded in logic-based insights.

4.7 Combining Verification and Explainability

Vaishak Belle (University of Edinburgh, GB, vaishak@ed.ac.uk)
Michael Benedikt (University of Oxford, GB, michael.benedikt@cs.ox.ac.uk)
Dana Drachsler-Cohen (Technion, IL, ddana@ee.technion.ac.il)
Daniel Neider (TU Dortmund University, DE, daniel.neider@tu-dortmund.de)

Joint work of All participants

Formal explanation techniques from classical software – such as SMT solving and UNSAT cores – primarily focus on input/output properties and safety, yet they do not always capture the intricacies of neural networks. Meanwhile, gradient-based attributions in neural networks can be brittle or overly localized, raising questions about how to ensure explanations generalize to unseen instances and how to pinpoint "interventions" that actually shift predictions. Explainable AI (XAI) and verification both rely on abstraction to address these issues: verification uses abstraction to isolate properties that can be formally proven or disproven, whereas explanation refines the model's salient behaviors so users can understand how inputs map to outputs. By capturing properties at a higher level, we obtain amenable properties for explanations, that can be more easily communicated and interpreted. Consequently, joint XAI and verification efforts could devise abstractions that both enable rigorous checks on model correctness and illuminate the model's inner workings. This synergy fosters AI systems that are trustworthy and interpretable, bridging the gap between formal correctness and human-centered understanding.

Participants

- Pablo Barcelo
PUC – Santiago de Chile, CL
- Vaishak Belle
University of Edinburgh, GB
- Michael Benedikt
University of Oxford, GB
- Alexandra Bugariu
MPI-SWS – Kaiserslautern, DE
- Luc De Raedt
KU Leuven, BE
- Dana Drachsler-Cohen
Technion – Haifa, IL
- Sophie Fellenz
RPTU
Kaiserslautern-Landau, DE
- Floris Geerts
University of Antwerp, BE
- Julien Girard-Satabin
CEA de Saclay –
Gif-sur-Yvette, FR
- Eleonora Giunchiglia
Imperial College London, GB
- Dominik Hintersdorf
TU Darmstadt, DE
- Vivian Holzapfel
Leibniz Universität
Hannover, DE
- Omri Isac
The Hebrew University of
Jerusalem, IL
- Matthias Lanzinger
TU Wien, AT
- Anji Liu
UCLA, US
- Anna Lukina
TU Delft, NL
- Pierre-Jean Meyer
Gustave Eiffel University –
Villeneuve d'Ascq, FR
- Emmanuel Müller
TU Dortmund University, DE
- Daniel Neider
TU Dortmund University, DE
- Ana Ozaki
University of Oslo, NO
- Martin Ritzert
Universität Göttingen, DE
- Patrick Schramowski
DFKI – Darmstadt, DE
- Mahmood Sharif
Tel Aviv University, IL
- Gagandeep Singh
University of Illinois –
Urbana-Champaign, US
- Armando Solar-Lezama
MIT – Cambridge, US
- Mihaela Stoian
University of Oxford, GB
- Mislav Stojanovic
TU Dortmund University, DE
- Lukas Struppek
TU Darmstadt, DE
- David Tena Cucala
Royal Holloway, University of
London, GB
- Ashish Tiwari
Microsoft Corporation –
Redmond, US
- Caterina Urban
INRIA & ENS Paris, FR
- Jan Van den Bussche
Hasselt University, BE
- Frank van Harmelen
VU Amsterdam, NL
- Emile van Krieken
University of Edinburgh, GB
- Jonni Virtema
University of Sheffield, GB
- Hongseok Yang
KAIST – Daejeon, KR
- Tom Yuviler
Technion – Haifa, IL
- Zsolt Zombori
Alfréd Rényi Institute of
Mathematics – Budapest, HU