HOW CAN RELIABILITY OF ARTIFICIAL INTELLIGENCE BE ENSURED?

GITTA KUTYNIOK

Artificial Intelligence (AI) is currently radically changing our society – a process often already referred to as the fourth industrial revolution. The recent advances in large language models and the release of ChatGPT, followed by GPT-4 have led to a significant increase in awareness of the potentials and dangers of artificial intelligence worldwide. This development was a main trigger, for instance, for the manifesto from Elon Musk and other tech leaders requesting a pause in the development of AI, for the EU AI Act, and for the G7 Hiroshima AI Process.

One significant problem of AI methodologies is their current lack of reliability, sometimes also referred to as a lack of trustworthiness. Indeed, AI-based approaches do encounter problems with safety. For instance, there have been already various accidents involving robots including autonomous vehicles. Another problem of concern is the lack of security since it is still possible to hack into AI systems such as in a hospital and take over control. Privacy rights and their violation are another obstacle still to be overcome since AI approaches require an enormous amount of training data. And, finally, most such algorithms still act like a black box and often also lead to biased decisions. Overcoming those problems and enabling reliable AI can only be achieved by a deep understanding in the sense of developing a theoretical underpinning of AI-based algorithms.

In this article, we will take a closer look from this perspective at the current workhorse of AI, namely artificial neural networks. Artificial neural networks are not as new as one might think. They were first introduced by [9] with the goal of developing an algorithmic approach to learning. Their idea was to mimic the human brain and introduce a mathematical model for its functionality. This led to the definition of an artificial neuron:

$$f(x_1, ..., x_n) = \rho\left(\sum_{j=1}^{n} x_j w_j - b\right), \quad f : \mathbb{R} \to \mathbb{R},$$

with weights $w_1, \ldots, w_n \in \mathbb{R}$, bias $b \in \mathbb{R}$, and activation function $\rho : \mathbb{R} \to \mathbb{R}$, typically chosen to be the so-called Rectifiable Linear Unit (ReLU) given by $\rho(x) = \max\{0, x\}$. Arranging artificial neurons in layers then yields the definition of an artificial neural network of depth L:

$$\Phi(x) = T_L\rho(T_{L-1}\rho(\ldots(\rho(T_1(x)))\ldots)), \quad \Phi : \mathbb{R}^n \to \mathbb{R},$$

where $T_k(x) = W_k(x) - b_k,\ k = 1, \ldots, L$ are affine-linear functions with W_k being the weight matrices and b_k being the bias vectors.

As can be seen, a neural network is in fact a purely mathematical object, thus also accessible to the full range of mathematical analysis tools, see [2]. Aiming to introduce the key research directions towards reliability, which are currently intensely studied, let us next consider the workflow of applying neural networks.

The key goal of applying a neural network is to learn a function that approximates a complicated relation such as a classification function of data on a manifold $\mathcal{M}$, i.e., $g : \mathcal{M} \to \{1, \ldots, K\}$. Given associated data $(x_i, y_i \approx g(x_i))_i$, one then proceeds as follows:

Step 1: The first step consists in splitting this data set into a training and a test data set.

Step 2: Next, an architecture has to be chosen, namely how many layers the network should have, how many neurons in each layer, etc.

Step 3: Once this is decided, the neural network is trained in the sense of optimizing the weight

Date: February 20, 2024.

matrices and bias vectors. This is typically done by a variant of gradient descent on the training data set, which solves the optimization problem

$$\min_{W_k, b_k} \sum_i \mathcal{L}(\Phi(x_i), y_i)$$

with $\mathcal{L}$ being a loss function such as the square loss.

Step 4: Finally, the neural network's ability to generalize to the test data, which the network has not seen during the training process, is checked.

From this, the following key theoretical questions arise:

(1) The first question to answer in our workflow is "Which is the best architecture for a particular problem setting?" The related research direction is termed "expressivity" and is currently the furthest explored one. The area is deeply rooted in approximation theory with one early highlight being the famous "Universal Approximation Theorem" by [7]. This result states that any continuous function can be approximated up to an arbitrary degree by a shallow neural network, i.e., a network with just one hidden layer. The central goal of this research tread is to analyze the best-case performance of a given architecture. One current research direction aims to analyze to which degree neural networks can mimic classical approximation methods. Surprisingly, current results show that neural networks are indeed amazingly universal and, for instance, achieve optimal approximation properties of all affine systems such as wavelets and shearlets combined, see [4].

(2) The training phase still bears many mysteries. The optimization problem of neural networks is highly non-convex, hence there can be spurious local minima in the loss landscape, and even saddle points and local maxima. This mean that a first-order method, such as gradient descent, is not guaranteed to converge to a global minimum, and even to any minimum at all. Yet, stochastic gradient descent typically finds "good" local minima, enabling the network to generalize well to unseen data sets. This research direction, classically termed "learning", requires methods from mathematical optimization and optimal control. Lately, also tools from areas such as algebraic geometry have been used to study connections between the shape of local minima and their suitability for generalization.

One thread of research focusses on the analysis of the loss landscape of overparametrized neural networks, showing that such loss landscapes tend to have only manifolds of global minima and no bad local minima. Another key phenomenon which was observed experimentally and subsequently analyzed theoretically is "neural collapse" (see [10]), which means that during the last stages of training the class features form well separated clusters in feature space. This shows that intriguingly training beyond zero training error does not lead to a highly overfitted model as one might expect, hence can be beneficial for the generalization performance.

(3) Finally, the performance of the trained neural network needs to be analyzed, which is often done jointly with the training process. One main goal of this research focus coined "generalization" is to unravel the theory behind the effects of overparameterization. Empirical evidence leads to the so-called double descent curve (cf. [1]), which shows this phenomenon. Analyzing the success of the trained network on the test data set and its dependence on the number of parameters typically requires a statistical/probabilistic viewpoint, and besides more traditional methods such as VC dimension and Rademacher complexity, novel techniques like the Neural Tangent Kernel were introduced. A particular difficulty also results from the fact that even if the global optimum is not discovered by a given algorithm in the optimization landscape, the statistical performance in terms of generalization might still be sufficient. Research in this area ultimately aims to introduce error bounds for the performance of trained neural networks, ensuring their reliability. In this regard, also robustness needs to be studied, as it is evident that the decision of neural networks is sometimes and often very sensitive to small changes of the input.

One main thread of research aims for such error bounds, which have by now been achieved in many, though often specialized, settings for (graph) neural networks. Another research direction focuses on the understanding the double descent curve, leading to results on an implicit bias during the training process such as an automatic smooth interpolation of the data, often termed the "universal law of robustness", cf. [5].

Regarding deep learning as a statistical learning problem, those three research directions constitute precisely the three components of the associated error, namely the approximation error, the error due to the learning algorithm, and the out-of-sample error.

While it is crucial to develop a deep theoretical understanding of the entire training process, in the future practitioners will often also encounter the situation where they do not have access to the training data or information about the training process. However, even without this knowledge, the "Right to Explanation" is often required for IT technology such as in the EU AI Act. This demand initiated the area of "explainability", which aims to analyze which aspects of the input data led the AI to a particular decision/output. Classical approaches highlight key features of the input, which are most relevant for a classification decision. With the ultimate goal being to enable a practitioner to communicate and question an AI-based approach as a human, the development of ChatGPT gave this field a significant boost.

One should however emphasize that for ensuring legal requirements, explainability approaches themselves have to also be reliable, a fact which is often ignored. This gives a preference to theoretically grounded concepts based on Shapley values or, recently, information theory, see [8]. In addition, finding meaningful features for the explanations constitutes a delicate problem with recent contributions suggesting to consider super pixels, wavelet clusters, or features from segmentations when considering images.

Despite all successes of deep neural networks and AI in general, those methods are not a Swiss army knife in the sense that they do have fundamental limitations. This research direction is however not sufficiently strongly pursued. One example of such a study concerns the fact that today most AI-based approaches are trained and run on digital hardware such as GPUs. Since many problems in science and engineering are of continuum nature (e.g., solving inverse problems in imaging science or (partial) differential equations in engineering), whereas digital hardware is intrinsically discrete, a discrepancy is unavoidable. Modeling digital hardware by the Turing model as it is custom, it was recently shown that this discrepancy unfortunately leads to non-computability of various problems such as certain classification problems, inverse problems, or even the pseudo-inverse which is central for many algorithms (see, e.g., [6]). This implies that for those tasks no training algorithm performed on digital hardware can guarantee any given accuracy, showing that the output of resulting neural networks is not reliable in the sense of having no guarantees. This could point towards why instabilities and non-robustness occurs for deep neural networks. At the same time, theory such as by [3] indicates that using analog hardware such as neuromorphic chips or quantum computing modeled by a Blum–Shub–Smale machine could overcome this obstacle. Hence, truly reliable AI might only be possible by going beyond classical digital computing platforms and augmenting it suitably by analog hardware. Considering the high amounts of energy required for training AI-based approaches on GPUs, also from the perspective of sustainability the consideration of novel innovative hardware such as neuromorphic computing will be a necessity in the future, which was also one reason for the CHIPS and Science Act in the US.

Summarizing, the lack of reliability of artificial intelligence is one of the current major obstacles worldwide which concerns any application of AI-based approaches. It can only be resolved by a deep mathematical understanding of the training and decision procedures of AI-based algorithms, which would allow, for instance, guarantees for success in terms of error bounds, similar to what is custom in many parts of computer science and engineering. In addition, automated certification of AI technology with respect to regulations such as the EU AI Act requires a formalization in the sense of mathematization of terms such as the "Right to Explanation" to avoid free interpretation depending on economic demands. Hence, reliability of AI is inextricably linked to a mathematical perspective.

References

[1] M. Belkin, D. Hsu, S. Ma, and S. Mandal. Reconciling modern machine learning practice and the bias-variance trade-off. Proc. Natl. Acad. Sci. **116** (2019), 15849–15854.

[2] J. Berner, P. Grohs, G. Kutyniok, and P. Petersen. The modern mathematics of deep learning. In: Mathematical Aspects of Deep Learning, Cambridge University Press, 2022.

[3] H. Boche, A. Fono and G. Kutyniok. Inverse problems are solvable on real number signal processing hardware, preprint, arxiv:2204.02066.

[4] H. Bölcskei, P. Grohs, G. Kutyniok, and P. Petersen. Optimal approximation with sparsely connected deep neural networks. SIAM J. Math. Data Sci. **1** (2019), 8–45.

[5] S. Bubeck and M. Sellke. A universal law of robustness via isoperimetry. J. ACM **70** (2023), 1–18.

[6] M, Colbrook, V. Antun, and A. C. Hansen. The difficulty of computing stable and accurate neural networks: On the barriers of deep learning and Smale's 18th problem. Proc. Natl. Acad. Sci. **119** (2022), e2107151119.

[7] G. Cybenko. Approximation by superpositions of a sigmoidal function, Math. Control Signal **2** (1989), 303–314.

[8] S. Kolek, D. Nguyen, R. Levie, J. Bruna, and G. Kutyniok. A rate-distortion framework for explaining black-box model decisions. In: xxAI - Beyond explainable Artificial Intelligence, Springer, 2022.

[9] W. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. B. Math. Biol., **5** (1943), 115–133.

[10] V. Papyan, X. Y. Han, and D. L. Donoho. Prevalence of neural collapse during the terminal phase of deep learning training. Proc. Natl. Acad. Sci. **117** (2020), 24652–24663.

Department of Mathematics, Ludwig-Maximilians-Universität München (LMU Munich), Germany
Email address: kutyniok@math.lmu.de