Judging Machines. Philosophical Aspects of Deep Learning

Arno Schubbach

Accepted for publication in “Synthese. An International Journal for Epistemology, Methodology and Philosophy of Science” / February 22, 2019.

1 Introduction

The recent and astounding boom of artificial intelligence reached the general public at least since AlphaGo triumphed over one of the strongest Go players, Lee Sedol, in 2016. Today, it seems almost inevitable to stumble upon some eye-catching news about artificial intelligence on a daily basis. Success messages by researchers or sensational announcements by start-up companies are promising to improve medical diagnostics, to prevent crimes through predictive analytics or to convict offenders with the help of facial recognition, while yet others aspire to automate translations or journalist writing, assist the car drivers or even substitute them, and so on and so forth. It seems to be a common presumption that artificial intelligence, together with big data, makes one of the most important technologies shaping our future. Therefore, it comes as no surprise that the political representatives see their nation states competing for a key technology and try to outdo each other with research programs. Of course, these prospects produce just as much fear and anxiety as they nurture hopes and utopian desires.

In short, our situation can be summed up as follows: „Every day we read that digital computers play chess, translate languages, recognize patterns, and will soon be able to take over our jobs.“ (Dreyfus 1992, p. 79) Yet, this description was originally published in 1972 by Hubert Dreyfus, one of the most prominent critics of the first heyday of artificial intelligence research in the 1960 and 1970s that has never lived up to its promises. So, it would be easy to discredit the enormous attention given to this technique as a media hype and a well-known strategy of scientists to raise money for their research and of start-ups to attract investors. However, such criticism seems quite hasty in view of the astonishing technical developments and their very rapid spread in many fields of application within the last years. A new assessment of this research, its applications and possible consequences presupposes that we have appropriate concepts at our disposal. Yet, as Luciano Floridi has recently indicated, we lack these concepts:

> the current conceptual toolbox is not fitted to address new ICT-related [information and communication technology-related] challenges and leads to negative projections about the future: we fear and reject what we fail to make sense of and give meaning to. (Floridi 2015, p. 7)

Whether the challenges will appear less frightening once we find the adequate conceptual tools giving meaning to applications of artificial intelligence remains to be seen. First of all, we need to find the conceptual tools to describe, conceive and assess especially the methods and applications originating in artificial intelligence research as well as their profound impact on society, economy and science.

Against this rather broad background, the following contribution will suggest conceptual tools for epistemologically reflecting on the newer and successful approaches of artificial intelligence research, particularly in the area of deep learning. These conceptual tools are supposed to grasp how deep learning networks change our interaction with and our understanding of computers. In view of the increasing application of this technology and its potentially far-reaching consequences, it seems urgent to develop conceptual tools for their philosophical reflection beyond the machine learning community.

The deep learning approach began to develop in the 1940s, but proved to be successful only in recent years.[1] In the second section, I sketch the characteristics of deep learning by a short comparison of AlphaGo's triumph over Lee Sedol in Go with Deep Blue's win over Garri Kasparov in chess twenty years earlier.[2] In the third section, I will specify how deep learning breaks with the formalization constitutive of algorithmic approaches to computing and discuss one crucial consequence: The results of a deep learning network are no more formal than the process in which they are obtained. Neither do we know the hows and whys of the outputs nor can we assume that a programmer has made sure that the outputs are computed correctly. This lack of transparency raises questions about the ways in which we can explain or justify the outputs of a deep learning network. These questions will be discussed in the fourth section with reference to the research literature in computer

[1] For a short sketch of the history of this approach cf. Goodfellow, Bengio & Courville (2016, pp. 12-26).

[2] For the historical background of these two contests cf. Heßler (2017, p. 1-33). In the second section, I will develop a systematic comparison of Deep Blue and AlphaGo instead of writing a history of artificial intelligence research.

sciences. The challenge of explaining or justifying the way DLNs operate points to a change in our understanding of computers and their role in human practice: The output of a deep learning network is not to be explained as the result of a mechanical and verifiable rule-based calculation. Rather, it takes on the character of a human judgment that we have to trust. By a short excursion into Immanuel Kant's philosophical analysis of judgments, the fifth section will delve into the analogy of a deep learning network's outputs and an expert's judgments. Thereby I characterize the form of justification adequate for the outputs of deep learning networks: We can understand deep learning networks as judging machines by specifying the type of judgments and the form of justification they are capable of.

2 AlphaGo vs. Deep Blue, deep learning vs. 'brute force' approaches

In 1996, IBM's Deep Blue was an advanced piece of parallel hardware and was designed on the basis of a 'brute force' approach: The program computed and evaluated many, many possible moves and included furthermore a database of openings and endgames provided by chess masters.[3] In 2016, the approach of AlphaGo – developed by the start-up Deep Mind acquired by Google – was a totally different one. It is based on 'deep neural networks', that is, networks of single processors called 'neurons'. These networks are structured in layers, an input layer, many hidden layers, and an output layer. By processing a training set of data, in this case a huge number of Go experts' moves, and comparing the outputs with the desired results, the weights of the links between the so-called 'neurons' are adjusted. Thereby, such networks are able to learn from the training data.[4]

IBM's Deep Blue and Google's AlphaGo were both very advanced, but also very different machines. Each can be seen against a different historical background of analogies between computing machines and human intelligence, i.e., on the basis of either the supposed common formal character of algorithmic programs and logical thinking or the alleged common processing of a network of artificial or biological type.[5] Yet, I do not want to

[3] For more technical details cf. the developer's paper Campbell, Hoane & Hsu (2002, pp. 57-83); for the historical background cf. Ensmenger (2011, pp. 5-30, esp. 10-17).

[4] Surely, this description is highly simplified. For a more detailed description cf. Silver et al. (2016, pp. 484-489).

[5] The assumption of a formal and logical character of thinking as well as computation was the basis of artificial intelligence research for a long time (Floridi 1999, pp. 132-134 and 146-148) and made chess one of their

discuss these analogies and the related speculative question of whether artificial machines allow insights into human intelligence by imitating it.[6] Instead of focusing on this "psychological question", I would like to address the "engineering question of artificial intelligence" and the epistemological significance of the different approaches (Collins 1990, p. 14). Hence, the present contribution considers the different technical paradigms of artificial intelligence research and focuses on the epistemological consequences of deep neural networks for our cooperation with computers.

Framed from this perspective, when IBM's Deep Blue was labelled 'deep', it was not in the technical sense of deep learning: "Deep Blue, as it stands today, is not a 'learning system.'"[7] Deep Blue's functionality was defined by formalized algorithms implemented in hardware and programs in the imperative programming language C.[8] Consequently, its operations were predetermined by the developers and their code: They could expect to understand the weaknesses and strengths of their machine because they knew the hardware and program they had implemented. As one of the developers of Deep Blue, Feng-hsiung Hsu, put it: "I could tell precisely what hardware evaluation features were at play in each game." (Hsu 2002, p. 200) A similar understanding of the decisions of a 'learning system' like AlphaGo, on the other hand, would be rather surprising. AlphaGo was not called but rather qualified as 'deep', in the technical sense that its functionality was first and foremost based on 'deep learning' – a term coined only around 2006[9] –: Deep learning systems operate in dependence of the architecture of the network and the weights of its links being themselves the result of adapting to data by training. Consequently, AlphaGo's functionality is not based on a formal algorithm, but on its architecture and its adjustment to data by learning.

---

pivotal paradigms (Heßler 2017, pp. 6-9). For further cultural and historical reasons for the crucial role of chess cf. Ensmenger (2011, pp. 17-21).

[6] The analogy of the formal and logical character of thinking and computation was the object of the – let's say – classical critique of artificial intelligence research, cf. Dreyfus (1992, pp. 67-79 and 155-188) or Searle (1984, 28-56). The analogy of biological and artificial networks is critically discussed in Floridi (1999, pp. 169-175). Goodfellow, Bengio & Courville (2016. P. 16) draw the conclusion: "one should not view deep learning as an attempt to simulate the brain. Modern deep learning draws inspiration from many fields".

[7] Cf. https://de.wikipedia.org/wiki/Deep_Blue. The name 'Deep Blue' goes back to the computer 'Deep Thought' in Douglas Adams' *The Hitchiker's Guide to the Galaxy* and IBM's nick name 'Big Blue', cf. Hsu (2002, pp. 69 and 126sq.). If there is any technical reason for calling it deep, it is the fact that it could perform 'deep searches' within the tree of possible moves and consequential moves, cf. ib. (p. 197).

[8] For a preliminary definition of algorithms cf. Floridi (1999, p. 47).

[9] I follow here the well-informed guess of Schmidhuber (2015, p. 96).

This outline provides a first idea of what a deep learning network (DLN) is as well as of the conditions for the recent successes and the current boom of artificial intelligence. Newer approaches are not based on formalization of the problems addressed by them and thereby break with the traditional premise of 'good old fashioned artificial intelligence'. On the one hand, 'brute force' approaches presupposed that it is possible to formalize what and how to compute, which is in principle, e.g., in the case of chess or Go, rather simple. Subsequently, the crucial challenge was to develop efficient algorithms and to have sufficient computing power for the necessary computations, whose complexity is much lower in the case of chess than in the case of Go. Yet, this approach is not suitable for a lot of cases in which it is unclear what and how to compute, which again explains why such approaches failed in fields like machine vision or natural language processing. On the other hand, artificial intelligence research was dominated by the 'knowledge base approach': Developers of expert systems tried to formalize the relevant knowledge for a specific task by formalizing concepts, their semantic relations and rules of inference.[10] However, it was not only a time-consuming task to implement the necessary knowledge but also the researchers often simply reached the limits of what can be formalized. Consequently, the critical debate about artificial intelligence focused on the premise of formalization by arguing for the necessary and narrow limits of the formalization of human practice (as the philosophers Hubert Dreyfus and John Searle) or by addressing the social conditions for formalizations of specific practices like arithmetic (as the social theoretician of knowledge Harry Collins).[11] The limits of formalization became apparent in theoretical debates as well as in artificial intelligence research. Finally, this approach was considered a dead end and people began to speak of the 'AI winter'.

By breaking with the assumption of formalization, deep learning networks overcame at least some of the limits of the older approaches.[12] This methodological innovation as well as its astounding success and broad deployment makes the lack of conceptual tools to conceive this new approach and to assess its possible consequences evident. Furthermore, this lack becomes obvious in the common speculative debates on artificial intelligence

[10] For an introduction to this approach cf. Floridi (1999, pp. 196-207). This is the state of the art discussed in one of the most interesting philosophical approaches to artificial intelligence research, Donald Gillies' *Artificial Intelligence and Scientific Method* (1996).

[11] Cf. in addition to the already cited passages of Dreyfus' and Searle's texts Dreyfus (1992, pp. ix-xxx) and Collins (1990, pp. 3-58).

[12] Cf. Schmidhuber (2015, p. 97), with reference to Deep Blue's contest with Kasparov in 1997 and the pattern recognition of small children and computers then and in 2011.

marked by the recurrence of old cultural phantasies and dystopian fears that are much more linked to science fiction than to the actual technical developments. With regard to the latter, the overcoming of human kind by intelligent machines seems to be highly speculative because all available working machines are far removed from man's general and adaptable intellectual abilities: They are special purpose machines designed for special tasks, like playing chess *or* Go, and can compete with humans only in the particular tasks they are made for.[13] So, the topos 'man against machine' may have some sense, albeit the very limited one of a chess or Go contest, but it certainly has nothing to do with fantastic ideas about machines that turn against their creator.

Perhaps, however, these dystopian fears intuitively express aspects of some technical changes that indeed should be taken seriously by philosophers. Undoubtedly, AlphaGo is not an autonomous machine, yet in a certain sense it does function more autonomously than the computers we have been used to. One of the developers of Deep Blue, Hsu, asserts:

> The 'man versus machine' angle apparently sells well for chess books, but it does not capture the true essence of the contest. The contest was really between men in two different roles: man as a performer and man as a toolmaker. (Hsu 2002, p. ix)[14]

This description expresses the fact that the functionality of a machine is in principle predetermined by the design of the hardware and the programs written in an imperative language. In contrast, the functionality of a DLN is not defined in advance, but is established through training, which gives us the means to explain why AlphaGo is described as operating in a 'more autonomous way'. For these reasons, its developers could hardly relate AlphaGo's decisions to the functions and procedures that they had implemented or failed to implement. While AlphaGo was very successful in playing Go, its developers could not explain why it placed its checkers like it did. The same holds for Go experts. Not only was AlphaGo trained by many historic Go games played by humans but also it was in a second phase trained by playing against instances of itself, thereby developing its own strategies

[13] That artificial intelligence research made progress in developing special purpose machines, is only a criticism if we presuppose that its primary aim is to imitate intelligent human abilities and that this aim was lost by focusing on special purposes, cf. paradigmatically Dreyfus (1992, p. 27). Instead, the present article is focusing on the different approaches of these special purpose machines and their epistemological consequences.

[14] Against the backdrop of philosophy of technology, we could dispute if it is adequate to conceive of such a complex machine as Deep Blue as a tool. At this point, however, this discussion leads astray.

that were unknown to the human Go tradition and therefore surprised and bewildered its experts.[15] The relation between the developer, as well as the experts and the machine changed, so that Lee Sedol hardly played against 'man as a toolmaker' or even as a teacher,[16] but rather against a self-learning machine.

This observations should make it clear that deep learning profoundly affects our understanding of computers and computation and requires a revision of the philosophical reflection about them.[17] For this purpose, I will set aside the well-known speculative debates for some time, and instead take into account the actual state of research and applications for developing adequate conceptual tools that succeed in capturing the effect of deep learning on our cooperation with computers and our understanding of computation.[18]

3 The lack of transparency of deep learning networks

The new approach of deep learning networks changed what programmers of artificial intelligence applications actually do. Whereas they formerly had to find ways to formalize the problem in such a way that solutions could be calculated or to provide a knowledge base that enabled the computer to deduce answers, they now have to set up and train a network. Dreyfus observed this difference already in the 1990s:

---

[15] Cade Metz, The Sadness and the Beauty of Watching Google's AI Play Go, in: Wired, March, 3, 2016, https://www.wired.com/2016/03/sadness-beauty-watching-googles-ai-play-go/ [last access 18 June 2018]. Technically speaking, AlphaGo was trained using a combination of supervised and unsupervised learning (i.e., reinforcement), cf. Silver (2016, pp. 484-486). While supervised learning requires the definition of the desired behavior of a DLN by a target value for every element of the training data, unsupervised learning aims to identify patterns within the data without such specifications.

[16] In a further step, Silver et al. (2017, pp. 354-359) developed AlphaGo Zero exclusively based on unsupervised learning so that it dispenses with the requirement of human expertise: "AlphaGo becomes its own teacher". So, it discovered "not only fundamental elements of human Go knowledge, but also non-standard strategies beyond the scope of traditional Go knowledge." (ib., p. 357)

[17] Dreyfus already saw the philosophical importance of 'neural networks': „neural networks raise deep philosophical questions. It seems that they undermine the fundamental rationalist assumption that one must have abstracted a theory of a domain in order to behave intelligently in that domain." (Dreyfus 1992, p. xxxiii) But in difference to the present paper, Dreyfus discusses artificial intelligence research primarily in view of the aim to replicate general and adaptive human intelligence and criticizes it on the basis of his "phenomenology of human intelligent action" (ib., lisq.). For this purpose, he readapts his criticism of expert systems to machine learning, cf. ib. (pp. xxxiii-xlvi); but this readaptation seems to be less accurate, also because research on DLNs has made enormous progress since then.

[18] In this paper I focus on our cooperation with computers; the consequences for our understanding of computing I plan to unfold in a second paper. There, I would like to detail the question of how deep learning introduces a new paradigm of computing and a new conception of 'representation' by computer 'models'.

> Indeed, the most striking difference between neural-network modeling and GOFAI [good old fashioned artificial intelligence] is that the neural-network modeler provides not rules relating features of the domain but a history of training input-output pairs, and the network organizes itself by adjusting its many parameters so as to map inputs into outputs, that is, situations into responses. (Dreyfus 1992, p. xv)

Yet, setting up a deep neural network is not a trivial job.[19] Usually, a programmer uses one of the popular software libraries that enables them to set up a DLN without much effort. However, she has to define the general architecture of the network adequate for the task at hand and the desired performance.[20] Furthermore, she has to specify activation functions for the single processors ('neurons') determining their output in relation to their inputs. In addition, she has to choose the starting values of the weights, that is, the numbers associated to each link determining its importance as input for the next single processor. Then, the network has to be trained with the help of a training set of data: By feeding data into the input layer, processing it through the single processors and communicating their outputs along the links and in dependence of their weights to the next processors, the output layer finally produces a result. This result, again, can be compared to the desired result included in the training data, in order to calculate the error and to adjust the weights of the links.[21] By this, a "learning through weight changes" (Schmidhuber 2015, p. 87) is implemented. Each of these different steps implies specific methodological problems und engineering challenges: choosing the architecture adequate for the task is not trivial; the training of the network, however, is even more challenging.[22]

---

[19] At different levels of difficulty, there are many good introductions to deep learning available that offer first insights into the implementation of a learning network. Rashid (2016), offers an easy-to-read introduction; Grant Sanderson's video tutorial www.youtube.com/playlist?list=PLZHQObOWTQDNU6R1_67000Dx_ZCJB-3pi takes a similar approach and can be recommended as a very entertaining introduction; Buduma (2017) is more demanding as well as comprehensive and works with Google's TensorFlow; Graupe (2016, esp. 111sqq.) puts an interesting emphasis on applications of DLNs. The most encompassing overview is provided by Goodfellow, Bengio & Courville (2012).

[20] The most basic difference concerns cyclic (recurrent) or non-cyclic (feedforward) networks. Convolutional networks are a special type of feedforward networks known for their astonishing performance in many important applications, cf. the overview of LeCun, Bengio & Hinton (2015, pp. 439sq.).

[21] In the case of supervised learning backpropagation is the most important algorithm for adjusting the weights of the links, cf. Schmidhuber (2015, p. 91). For the basic idea of this algorithm and for an overview of the historical development of the research into it, cf. ib., (pp. 89-94). In the case of unsupervised learning reinforcement takes its place, cf. ib. (pp. 100-103).

[22] Cf. Buduma (2017, pp. 27-37) for a short overview over the most important problems of training and the crafts of optimizing the training.

The break with the formalization of procedural algorithms or semantic knowledge relevant to the task at hand made possible the success of DLNs in many areas such as pattern recognition, object detection, image segmentation, natural language processing, advertising, or finance. Yet, the theoretical and practical consequences of this break with formalization have hardly been discussed sufficiently: The outputs of learning networks are not based on well-defined procedures or explicit criteria any more than their processing. Although we do get an output, we do neither know how this output was computed nor why it is this output and no other. Therefore, DLNs are regularly called 'black boxes': "Despite widespread adoption, machine learning models remain mostly black boxes."[23] This metaphorical expression indicates a lack of transparency in DLNs' mode of operation as well as the need for a more precise conceptual analysis. Most of the technical systems we use are 'black boxes' to us. None of us understands in detail the operations of his or her computer, as the design of the hardware is too complicated and too many lines of code make up a program. But this lack of transparency is due to the sheer complexity of technical systems in general: Only a few of us will understand how an old FM radio works. However, the lack of transparency peculiar to DLNs is of a different kind. This specific lack of transparency can be captured more precisely by comparing the mode of operation of a DLN with that of an algorithmic machine.

In order to implement the desired behavior in an algorithmic machine, the programmer anticipates every possible input and predetermines an adequate response to it. Thereby, she defines a formal frame within which the operations of the computer are determined. Consequently, the behavior of the machine is anticipated throughout the process of its programming, although this anticipation is in fact and due to the complexity of technical systems impossible, as the many bugs and security problems of our software prove on a daily basis. In contrast, the approach of machine learning is based on the insight that "it can be far easier to train a system by showing it examples of desired input-output behavior than to program it manually by anticipating the desired response for all possible inputs." (Jordan & Mitchell 2015, p. 255) Thus, the behavior of a DLN is neither anticipated by the programmer nor is it possible to anticipate it, because a DLN does not operate within a predefined formal framework. Instead, the DLN determines its own functionality by adapting its structure (i.e., the weight of the links) to the training set of data. Therefore, the

[23] Ribeiro, Singh & Guestrin (2016, abstract). For a more prominent text cf. Castelvecchi (2016).

operations and the behavior of a DLN cannot in principle be anticipated during its set-up, since they are only determined by, say, the 'history' or 'experience' of the DLN made through its training.[24] The programmer has to check if the DLN works as it is supposed to by using parts of the training data not for training, but for testing the trained model. Thereby, the error rate of the DLN can be calculated. However, this does not imply that we understand how the DLN works and why it delivers whatever outputs.[25]

Consequently, we have to draw a difference between the in-fact-impossibility to anticipate the operations of an algorithmic machine due to the complexity of the technical system and the in-principle-impossibility in the case of DLNs due to the fact that their functionality is established by learning. Because of this fundamental difference, the two lacks of transparency cannot be addressed in the same way. In the case of a classical computer program, we normally may not know how it works, but we may trust that the programmers have done a good job, have chosen adequate algorithms and implemented them correctly. If not, we could, at least in principle, try to trace the error by looking into the source code, learning the applied algorithms and discussing their implementation. With this aim in mind, 'algorithm watch groups' pursue the political demand that algorithms of great social importance be disclosed.[26] Nevertheless, this approach falls short in the case of a DLN that established its functionality via learning. A DLN, we have learned, depends on a training set of data as well as the network structure, the algorithms for adaptation and the specific training process. The result is a network whose processing is characterized by a huge matrix of weights of links that is hardly intelligible: It does not provide any insight into how and why certain inputs lead to certain outputs, neither to the programmer nor to the user of the DLN. Hence, opening the 'black box' of a DLN does not and cannot immediately produce the transparency required to understand how a DLN works.

This principle lack of transparency peculiar to DLNs entails a whole range of practical and theoretical challenges that can be addressed from different perspectives. One approach

[24] That a computing machine can be determined by its history is also highlighted by Wegner (1998), with reference to the more general conception of 'interactive machines'.

[25] Therefore, Zeiler & Fergus (2014, p. 818) observe for the most successful type of networks for image recognition and similar tasks, so called large convolutional network models, the following: "there is still little insight into the internal operation and behavior of these complex models, or how they achieve such good performance. From a scientific standpoint, this is deeply unsatisfactory. Without clear understanding of how and why they work, the development of better models is reduced to trial-and-error."

[26] Cf. https://algorithmwatch.org/ or https://netzpolitik.org/2018/new-york-city-plant-arbeitsgruppe-zur-ueberpruefung-von-staedtischen-algorithmen/ [last access 29 June 2018] on a municipal law in New York City with a similar aim.

common in the machine learning community is called 'statistical learning theory'. There, a mathematical theory is used to grasp the "inductive learning" from data typical for machine learning (Harman & Kulkarni 2007, pp. 19-21 and 36-44). For this purpose, it does not focus on the specific approaches of machine learning and the respective computing processes.[27] Rather, statistical learning theory tries to formalize the properties of the learning process from an external perspective, i.e. to measure, evaluate and improve the approximation process to an unknown "background probability distribution" (ib., pp. 33-36).[28]

In contrast, a second approach aims at gaining insight into the specific computing processes of DLNs that would help to explain how they achieve their often astonishing performance, e.g., how they can recognize objects in pictures. On this account, it is not sufficient to 'look into the black box' where we will only find thousands or millions of weights. Rather, we are in need of more advanced methods to reconstruct the learned functionality. A third approach addresses the lack of transparency of DLNs by providing users with additional information to understand and interpret the output of a DLN. The latter two approaches are particularly to be found in the deep learning community and often called 'interpretable' or 'explainable artificial intelligence'.[29] As these key words make clear, this approach, unlike the first, focuses on the understanding of the functionality of a DLN by experts or users. Thus, the question of the explanation or justification of a DLN's results is framed by our relation to a DLN and its computing processes. Yet, it is this relation that is most important if we want to understand how the technique of deep learning and its growing adoption possibly changes our understanding of computers and their role in human practices. Thus, in the following, I will draw on deep learning research that attempts to grasp

[27] The technique that is mostly used in the context of statistical learning theory is the approach of 'support vector machines' (SVM), cf. Kulkarni & Harman (2011, pp. 172-186) whereof deep learning is considered to be "a special case", cf. Harman & Kulkarni (2007, pp. 78-87, esp. 87). Thus, it is presupposed that DLNs – as SVMs – implement "rules that assign a classification to every possible set of features" (ib., p. 89). More precisely: "Such networks [feedforward neural networks] encode principles in their connection weights and there is no reason to expect people to have access to the relevant principles." (ib. p. 92) Against the backdrop of my argumentation, the assumption of encoded principles to which we have no access seems questionable.

[28] For an exemplary and interesting study on the training process of DLNs based on statistical mechanics cf. Martin & Mahoney (2017).

[29] The well-known research funding organization of the US-American military, DARPA, has launched a special funding program for this research field in 2016: "The goal of Explainable Artificial Intelligence (XAI) is to create a suite of new or modified machine learning techniques that produce explainable models that, when combined with effective explanation techniques, enable end users to understand, appropriately trust, and effectively manage the emerging generation of Artificial Intelligence (AI) systems." (https://www.darpa.mil/attachments/DARPA-BAA-16-53.pdf, 5 [last access 18 June 2018]). Yet, the challenge to explain or justify results has a longer history in machine learning as well as in adjacent fields, as Biran & Cotton (2017) show.

the computing processes of DLNs by discussing how their results can be explained or by justifying these results by supplemental information.

4 Explanations or justifications

Against the backdrop of deep learning research, it seems natural to react to the lack of transparency common to DLNs by inspecting their computation processes and thereby explaining their functionality. So, it is possible in the case of very simple networks to process a 'backwards query', i.e., to use the network in the other direction: By processing an output, we get an input representing the 'features' that are discriminatory for this output. In the case of a simple network identifying hand-written numbers for example, we process a single number and get a blurry picture exhibiting the visual structure 'defining' this number in hand-writing (Rashid 2016, pp. 178-182). In the case of more complex networks, however, 'backwards queries' are impossible. Thus, more advanced methods have to be developed in order to visualize the features that help understanding, say, a network for classifying images. As these networks usually are of an immense size, the state of the art of such 'feature visualizations' shows the features of different layers of the network and thus different levels of abstraction.[30] However, by the resulting image series we gain hardly any intelligible or formalizable knowledge about how the network processes its inputs.

Philosophically, it is crucial to discuss how exactly such technical responses to the lack of transparency of DLNs can help us understanding their functionality. The research literature itself introduces the difference between the explanation and the justification of the results of a DLN and links it to the difference between algorithmic computing and deep learning: In the case of an algorithmic computation, the justification of the result can be identified with the explanation of the computing process because the rule-based procedure of computation justifies the correctness of its results; in the case of a DLN, the justification of

[30] Cf. the well noticed paper Zeiler & Fergus (2014, pp. 818-825). They combine a convolutional network – the type of network that is the most important for a lot of applications – with a further deconvolutional network in order to visualize the features of relevance for the functionality of the different hidden layers, cf. ib. (p. 824). The very nice digital publication Olah, Mordvinstev & Schubert (2017) hints at the limits of this approach: "By itself, feature visualization will never give a completely satisfactory understanding. We see it as one of the fundamental building blocks that, combined with additional tools, will empower humans to understand these systems." (ib., without pagination, conclusion) Cf. also Mordvintsev, Olah & Tyka (2015) and Mahendran & Vedaldi (2016).

a result cannot recur to the computing process in the same way, as Or Biran and Kathleen McKeown argue:

> In contrast to rule-based systems, justifying the predictions of ML [machine learning] is not a straightforward task; it is no longer the case that *explaining* how a prediction was reached automatically *justifies* it to the user. Due to the complex, quantitative and unintuitive nature of most models, it is unreasonable to expect that users who are not ML experts, even if they are experts in the domain of the prediction, will understand how the model works, regardless of how transparently it is presented. (Biran & McKeown 2017, p. 1461)

Biran and McKeown assume that understanding the DLN and its output is only the problem of 'users who are not experts'. Yet, they introduce a principal difference: An *explanation* refers to the factual computation and legitimates the results by its rule-based procedure; in contrast, a *justification* is supposed to provide an understanding why this output is adequate independently of how it was computed. Biran and McKeown are very clear about this difference between explanation and justification: "Explanation answers the question 'how did the system arrive at the prediction?' while justification aims to address a different question, 'why should we believe the prediction is correct?'" (2014, without pagination, introduction) In the case of an algorithmic machine, explanation corresponds to justification because the rule-based computation justifies the correctness of its result. In the case of a DLN, however, explanation and justification have to be separated.[31]

The difference between explanation and justification illustrates that this whole discussion is closely connected to our relation to computers and its change through the application of deep learning. Although it is not consistently observed in the research literature, it is crucial to assess the distinct ways through which different technical approaches address the DLNs' lack of transparency. The above-mentioned approach of 'feature visualization' inspects a DLN's internal functionality in order to make explicit, as far as possible, the features it is 'looking for' and thus to *explain* how the outputs are computed. A different approach aims at *justifying* the output of a DLN by supplementing it with

[31] Justification in this sense is not to be equated with mathematical criteria or measures of the performance of learning machines. Different varieties of such measures are discussed in an interesting paper by Corfield (2010).

additional information intended to strengthen the user's confidence. For example, Lisa Anne Hendricks et al. combine a DLN for image recognition with a DLN for image captioning in order to produce "visual explanations" in natural language that do not only describe the pictures but highlight the discriminative features 'justifying' the output of the image recognition network (2016, p. 3sq.). However, these 'visual explanations' do not effectively explain how the images were classified but merely justify the result of this classification in the sense of Biran and McKeown.[32] They are not primarily supposed to increase our understanding of how the outputs were obtained, but to create trust in the decision of the DLN. For that purpose, we do not have to refer to the "features" the DLN operates with, but offer the users of the DLN "interpretable data representations" that facilitate their understanding, as Marco Tulio Ribeiro, Sameer Singh and Carlos Guestrin make clear: "*interpretable* explanations need to use a representation that is understandable to humans, regardless of the actual features used by the model [i.e., the DLN]." (2016, without pagination, section 3.1) In their paper "'Why Should I Trust You?'", they consequently focus on the question how to increase user trust by providing justifications of the output.

The difference between justifications and explanations is of considerable philosophical importance because it helps to further characterize the lack of transparency peculiar to DLNs and its epistemological consequences. In the case of an algorithmic machine, we normally identify explanation and justification because we assume the justification to be referring to the algorithm, which at the same time explains how the result in fact was computed. We can assume this because the algorithmic approach implies the "coupling between the programmable algorithmic procedure and the computational process of which it is a specification".[33] In the case of DLNs, we instead have to mark out the difference between explanation and justification, because the computation of the outputs does not justify the functionality of the DLN. An explanation would have to relate to the factual process of computation of the output, i.e., the processing of the input along the links and in dependence of their weights step for step until the output layer is reached. Algorithmically, this process is basically a multiplication of huge matrices and vectors whose

[32] Cf. Hendricks et al. (2016, pp. 3 and 5): "In contrast to systems […] which aim to explain the underlying mechanism behind a decision, Biran et al. [2014] concentrate on why a prediction is justifiable to the user. Such systems are advantageous because they do not rely on user familiarity with the design of an intelligent system in order to provide useful informations."

[33] Floridi (1999, p. 35), with reference to the universal Turing machine as the standard model of algorithmic processing.

numbers are given by the weights of the links or the input. Yet, such an explanation does not 'explain' very much and does not justify the output, because it does not provide any idea of the functionality of the DLN. The reason is that the algorithm used to calculate the output of different DLNs is every time the same, it is only parameterized by different numbers, i.e., the weights. Consequently, the specific functionality of a DLN is not specified by the algorithm used to calculate the outputs, but by the latter's parameters:

> The computation performed by the network in transforming the input pattern of activity to the output pattern depends on the set of connection strengths; these weights are usually regarded as encoding the system's knowledge. In this sense, the connection strengths play the role of the program in a conventional computer. (Smolensky 1988, p. 1)[34]

Consequently, we can explain the computation of the outputs by a matrix multiplication algorithm, but this explanation does not genuinely explain the functionality of the DLN, nor is it to be understood as its justification, since it does not provide any insights into the functionality of the network and it gives no intelligible reasons for the output. Alternatively, if we try to justify the outputs of the DLN, as Biran and McKeown or Hendricks et al. suggest, we have to be aware that the added justification will be largely independent from the actual computation of the DLN. Especially, it cannot refer to the functionality of the network in the form of a rule-based procedure that would allow to easily embed the input/output-relation into inferential structures, as in the case of an algorithm. In contrast, the weights of the links that in fact determine the DLN's functionality are hardly intelligible. Therefore, a justification of the output does not justify in an argumentative or even logical sense. Instead, it primarily aims at strengthening the user's confidence in the DLN where its lack of transparency appears impenetrable. For sure, this aim is of great interest for computer scientists and of great importance for the deployment of DLNs.

To sum up, the lack of transparency peculiar to DLNs translates into the absence of a justification that could claim an argumentative or logical value and be seamlessly embedded

[34] This means that the functionality of the DLN can be computed or simulated by a classical algorithmic machine, but its functionality is not defined in the form of an algorithm. Therefore, deep learning is to be distinguished from the algorithmic paradigm detailed by the universal Turing machine and to be understood as an own paradigm of computing. This argument I plan to unfold with reference to the philosophy of computing in a further paper.

in inferential contexts. This finding poses a particular challenge for the scientific application of DLNs. Science often may be a 'dirty practice', yet it finally aims at justified knowledge. Thus, every enthusiasm about a possible "partial automation of every element of scientific method, from hypothesis generation to model construction to decisive experimentation" (Mjolsness & DeCoste 2001, p. 2051) should be met with a more differentiated consideration. The use of DLNs and their effects on scientific practice must be precisely assessed. Whether DLNs are used to predict a particular tissue-dependent expression of genes (Leung et al. 2014) or the biological activity of substances on the basis of their physico-chemical properties and structure in pharmaceutical drug research (Ma et al. 2015); whether they are part of a complex scheme of detecting specific particles in the proton collisions of the LHC at CERN (Ciodaro et al. 2012), we must be aware that they perform these tasks without us exactly knowing why and how they find their results. To be sure, they usually replace stochastic models and their probability statements, yet we often understand these models more precisely as the functionality of DLNs. Furthermore, such data is usually not neutral and imbued with the way how we evaluated it. So, the DLNs can become instruments that embody theories and knowledge of yesterday while we keep using these instruments on a daily basis without thinking too much, as they automatically deliver the results for the continuation of our work. However, if we get used to DLNs without knowing and reflecting on how they work, we will possibly start to erroneously trust their outputs. Hence, any DLN could become an *asylum ignorantiae* in the midst of scientific practice.

From a philosophical point of view, however, the absence of justification in the sense of an explanation of the functionality of a DLN also raises the question of whether the outputs and the functionality of a DLN do not require another form of justification that is not based on the premise of a rule-based procedure and an algorithmic approach to computing. This question seems reasonable because a DLN and its outputs can be tentatively compared to a human expert and her judgments. The DLN resembles an experienced human being whose train of thought we cannot know in detail and whose judgment we nevertheless trust. Possibly, the doctor 'explains' her findings, but these 'explanations' primarily justify her judgment instead of explaining her conclusion on the basis of explicit rules and criteria. Certainly, she can try to make explicit the criteria and reasons for her findings, but she will hardly succeed in attempting to explain her judgment in reference to well-defined rules. As in the case of a DLN, the justifications react to the lack of transparency inherent in the

judgment of the human expert, but does not resolve this lack. Yet the case of expertise hints at a new form of justification: The expert's judgment can be justified by his experience or authority. Possibly, this form could also be adequate for justifying the functionality of a DLN that is established by its training, i.e., the experience and history of its processing. Accordingly, the DLN seems to put into effect the power to judge on the basis of experience – and not only the power to calculate on the basis of procedural rules. The learning machine would be not a calculating, but a judging machine whose verdicts we have to trust if we want to work with it.

Of course, this description suggests only a heuristic analogy. It is not meant to ascribe some psychological ability or intelligent behavior to computers or to anthropomorphize them. Rather, this analogy illustrates how we conceive of computers when interacting and working with them. Hitherto, the role computers play in practice was discussed mainly on the basis of our understanding them as algorithmic or calculating machines. Especially, the well-known approach of Harry Collins and Martin Kusch assumes that the functionality of computers consists in executing formal rules and procedures. This raises the question how this type of functionality can be embedded in a human practice which cannot be reduced to rule-based procedures.[35] So, they highlight the essential difference between human actions and mechanical computation and at the same time describe aspects of the mechanization of human actions that allow for a cooperation of humans and computers and for the latter's inclusion into human practice: "The two realms can interact where humans intentionally act like machines" (Collins & Kusch 1998, p. 196; cf. ib. pp. 1-3, 55-73, and 113-136). Starting with the above understanding of DLNs in contrast to algorithmic machines, the analogy of the functionality of learning machines and the judgment of human experts suggests another and new role of computers within human practice: The cooperation becomes less based on mechanization and formalization, but on learning, experience, and history which determines the DLN's decisions. In this respect, its functionality seems comparable to the judgment of another experienced human being. In the context of scientific or epistemic practices, however, this analogy is primarily worthy of discussion insofar as it can be developed further and thereby provide us with a better understanding of a form of justification that would fit

[35] Collins & Kusch (1998), p. 50, themselves occasionally concede possible limits of their approach, which refers almost exclusively to "good old artificial intelligence". Their criticism primarily covers "the research program of artificial intelligence (at least, the program that preceded neural nets and so forth)". 20 years ago, this limitation may not have been too severe, as there were good reasons to be skeptical about the performance of DLNs, cf. Collins and Kusch (ib., p. 129sq.).

to the specific mode of a DNL's operation. With this aim in mind, I want to explore Immanuel Kant's theory of judgment.

5 Kant on forms of judgment and justification

The theory of judgment is not an isolated part of Immanuel Kant's work. Rather, it permeates his entire philosophy and is one of its constitutive concepts. Whether we focus on his epistemology, his ethics or his aesthetics, it is almost impossible to characterize the Kantian approach without discussing his understanding of judgments in the respective fields. In the following, I will only outline these aspects of judgments that are helpful to develop a better understanding of the functionality of DLNs.

In general, Kant's theory of knowledge introduced in his epoch-making *Critique of Pure Reason* (1781/1998) can be understood as a theory of judgment, as according to him we know only such objects that are objects of an objectively valid judgment. With this intricate formulation, I want to hint at some basic features of Kant's epistemological approach. Firstly, objects of knowledge must not be confused with things that are given and exist independently of the process of knowledge. Rather, they are to be understood as the results or the correlates of knowledge. In this sense the *objects* of knowledge are in a strict sense objects *of knowledge*. Secondly, knowledge takes on the form of judgments that are not to be understood as referring to the relations between concepts, as the pre-Kantian tradition would have it, but as processes bringing together the two main sources of knowledge, i.e., intuitions and concepts. Thirdly, the validity of a judgment is based on its procedures because it depends on the concepts that are to be understood as general rules of synthesis which are applied to singular intuitions as its materials to be synthesized. In sum, the objects of knowledge are correlating to the processes of knowledge that take on the form of judgments whose validity again depends on the rules of judgment.

Further, the question is where the rules come from and it is this question exactly that allows us to use this analysis of judgments as heuristics for our understanding of the different paradigms of computation.[36] The aim of Kant's *Critique of Pure Reason* is to show that our understanding is the "faculty of rules". The faculty of rules does not apply any

[36] In the following I will draw partly on an interpretation of Kant elaborated in Schubbach (2016, 147sqq.).

arbitrary rule but necessarily follows a small set of rules inherent to it, i.e., the categories (Kant 1781/1998, p. 242). These rules are supposed to be constitutive of any object of knowledge and any form of knowledge must conform to them. The *Critique of Pure Reason* therefore sometimes conveys the impression that the theory of knowledge has thus been completed. Yet, this impression lives from the assumption that a theory of knowledge exclusively deals with the most general rules of understanding encompassing every form of knowledge.

However, at the latest in the *Critique of Judgment* (1790/2000) it becomes obvious that this is not sufficient for a theory of knowledge. Kant had realized that specific forms of knowledge – like physics, chemistry, or biology – furthermore require more specific rules adequate for different types of objects (moving physical bodies, chemical reactions of substances, living organisms and taxonomy of life forms). These rules cannot be necessary, because we apply them only within specific regions of empirical objects, and we have to learn about them from these very objects, that is, we firstly have to discover them and, secondly, have to validate them in the process of forming empirical knowledge. Hence, the *Critique of Judgment* introduces a second form of judgments, rules and objects in which the rules are no longer defined in advance and applied to any sensual data, but have to be extracted from the data and to stand the test of time.

Finally, the *Critique of Judgment* introduces us to another, third form of judgment, i.e., the aesthetic judgment that takes its norms from individual perceptions of individual works of art without being able to translate them into general rules.[37] We do not possess such rules for judging works of art, but we judge them nevertheless, based on the norms we learned by works of art and now transfer to others.[38] Thus, Kant's analysis of judgments ultimately leads to the opposite end of the relationship between the object and rule of the judgment: Had he first tried to explain judgments by their most general and presupposed rules, irrespective of the concrete object, he finally arrives at judgments which are not capable of being explained by well-defined rules and can at best only refer to authoritative objects. Following Hannah Ginsborg's interpretation, this last analysis also reveals the core of the judgment with regard to its epistemological implications:

[37] Kant (1790/2000, p. 121): "Rather, as a necessity that is thought in an aesthetic judgment, it can only be called **exemplary**, i.e., a necessity of the assent of **all** to a judgment that is regarded as an example of a universal rule that one cannot produce." [Emphases in original]

[38] Kant (1790/2000, p. 186): "since there can also be original nonsense, its [the genius'] products must at the same time be models, i.e., **exemplary**" [Emphases in original].

> What we need, to be able to claim in good faith that our capacity to judge is legitimate, is an independent sense that we can judge without relying on proofs and justifications: and it is this sense that aesthetic experience provides. (Ginsborg 1999, p. 218)

Against the odds, these three forms of judgments offer a helpful heuristic for understanding the difference between an algorithmic and a learning machine as discussed in the preceding sections. For this purpose, we take Kant's theory of judgments carried out by consciousness and transfer it to computers, not in order to speculate about the presumable pseudo-psychological states of computers, but to specify the role of rules and of their relation to (sense) data in different approaches to computing. With this said, it is possible to conceive of the classic algorithmic approach by comparing it to the first form of judgment according to our rough sketch of Kant's analysis. Algorithms define the rules of processing in advance of their actual execution, as the categories do for the understanding. This implies that the objects and the features that can be represented are also predetermined by these rules, be that the rules of an algorithm or of our understanding. Correspondingly, the data to which these rules are applied are nothing else than "raw material" – as Kant describes "sensible sensations" in relation to understanding and Floridi labels data in relation to algorithmic processing[39] –: Nothing has to be learned from these data, the rules of processing remain independent of them. Thereby, this first form of judgment has narrow limits: It can only relate to data and objects on the basis of its own rules and therefore can represent only these features that are predefined by these rules independent of specific data and objects.

A learning machine, on the contrary, is better understood by comparing it to the second or even third form of judgments. Deep learning breaks with the formalization of predefined algorithmic rules in order to implement a "learning from example" (Buduma 2017, p. 4), which also marks a crucial point in Kant's second and third form of judgment. Particularly, we can draw an important parallel in this respect, namely that 'learning from example' is not restricted to finding out if an object has this or that feature out of a predefined set of possibilities. Instead, this type of learning consists in "representation

[39] Kant (1791/1998, p. 127): "Experience is without doubt the first product that our understanding brings forth as it works on the raw material of sensible sensations." Floridi (1999, p. 229), with reference to the conception of the algorithm insofar as it was already outlined by the mechanical calculators in the prehistory of computing.

learning" (LeCun, Bengio & Hinton 2015, p. 436), that is, in learning the adequate conditions for representing specific types of objects and their special features. This is essential for judgments in the sense of Kant's second form of judgments (in respect to moving bodies, reactions of substances, living organisms) as well as for deep learning that adapts the network to the structures hidden in the data and thereby develops ways of representing their essential features (the distinguishing features of written letters, the features of specific objects within a picture, the features of a specific painting style, etc.). Accordingly, the results of a DLN seem to be comparable to Kant's empirical judgments as they first and foremost are based on learning the frames of representation from the data and objects they relate to.

On closer reflection, however, the comparison of the learning machine and Kant's second form of judgment appears premature. In the case of Kant's empirical judgment, the frames of representation learned from examples are nothing else than the conceptual conditions, thus the explicit rules for processing the data and representing the objects. Consequently, the outputs of the DLN's operations can only be compared to Kant's empirical judgment if its functionality can, at least in retrospect, be translated into explicit rules and criteria. If this is not possible, then the outputs appear to be closer to the aesthetic judgment: The aesthetic judgment is also based on 'learning from example', but in contrast to the empirical judgment it cannot make explicit the norms learned from exemplary works of art. Following Kant, these norms cannot be translated into general rules, because the aesthetic judgment can and must not be based on such rules. This is also true for the DLN, if the argument of the preceding section of this paper is correct. The outputs of a DLN are calculated on the basis of rather simple rules, yet this does not imply that the functionality simulated in this way can be translated into the algorithmic form of procedural rules and explicit criteria. Kant's analysis of judgment helps us to explain why such a translation seems impossible: The different forms of judgment progressively reduce the role of formal procedures while strengthening the role of the objects and their inherent structures. It is this development that explores the possibility of judgments which are not justified by their presupposed formal rules but by their inherent involvement with their objects. Similarly, DLNs engage in 'representation learning' and 'learning from examples' that distinguishes them from the algorithmic approach on the same basis, i.e., by an inherent involvement with data effected by the adaptation of the network and the weights of its links during training.

Therefore, a DLN can be conceptualized as a kind of judging machine that takes judgments 'without relying on proofs and justifications', if we presuppose that this justification could only consist in an unbroken chain or seamless net of inferences.[40] Nevertheless, Kant's understanding of the aesthetic judgment hints at a different form of justification of judgments through their involvement with data or objects. It is this form of justification that is more appropriate for the functionality of DLNs, as this functionality is established through learning and experience.

6 Conclusion

Kant's analysis of different forms of judgment yields some valuable insights into how we can understand the functionality of DLNs within human practice by analogy to an expert's judgment. Firstly, the lack of transparency is not only typical for a DLN, but also for the human expertise and its form of judgment: The outputs of a DLN as well as the decisions taken by humans are mostly not transparent, as they are rarely based on procedural rules and explicit criteria. Consequently, the lack of transparency inherent in the functionality of DLNs is not a totally new problem of epistemic practices. Nevertheless, it is new that the outputs of computers participate in a similar kind of lack of transparency as human judgment because they are not the result of an algorithm that would, at least in principle, explain and justify the why and how of this output. Secondly, Kant's analysis of judgments provides us with a framework to assess the lack of transparency common to this type of judgments. This lack is not only a shortcoming in comparison to strictly rule-based judgments but rather a positive characteristic of a different type of judgments: They get involved with their specific objects or data in order to extract their structures and adapt to them. For this purpose, they cannot rely on presupposed rules. Thirdly, Kant's analysis

[40] For Kant, a judgment without objective validity based on the rules of understanding remains a kind of philosophical curiosity. Therefore, he introduces a new form of so called 'intersubjective validity' adequate for the aesthetic judgment. This intersubjective validity is based not on common rules of processing as the objective validity of knowledge judgments, but expresses the common reaction to a sensory stimulus that is to be explained by the common constitution and faculties of human beings, cf. Kant (2000, p. 170). This argument results out of a rather simple and disputable reading, but sheds an interesting light on the question to what extent we can understand and comprehend the results of DLNs. Kant's argument seems not to go any further, since the processing of DLNs and human judgement do not operate on a common basis, and self-learning machines develop their own mode of operation. The moves of AlphaGo that appeared totally foreign to human Go experts shortly discussed in the second section of this paper seem to confirm this thought.

suggests the argument that it is a different type of task and situation requiring such judgments being entangled with objects and data. If deep learning is to be considered as an own paradigm of computing, this still does not imply that it is the kind of new paradigm that is going to replace the old one. Rather, deep learning should be understood as a paradigm linked to specific tasks in situations that defy judgments based on procedural rules and explicit criteria.

Finally, Kant's analysis of judgments suggests a new form of justification that seems appropriate to judging machines. Thus, it seems to be impossible *to justify these outputs* as in the case of an algorithmic processing *by explaining how they are computed*. As we saw in the fourth section, that is why researchers address this lack of explanation by adding a complementary 'justification' to the output. Yet, this 'justification' does not *justify in a strict, argumentative or even logical sense*, rather it aims at building trust on the user's side and thereby at facilitating the widespread use of DLNs. In contrast, Kant's analysis of judgment points toward a different form of justification, one that is not based on rules, but still necessary and essential where we do not and cannot know rules. In this situation, judgments can be *justified by experience, by familiarity or involvement with objects and data*. It is this form of justification which the expert could claim for herself and which we could also concede to a DLN. For sure, this form of justification does not seamlessly and completely integrate into a chain or network of logical inferences as an algorithmic program could possibly promise. On the contrary, it intercepts them. That is why every judgment linked to such a form of justification does not exclude its critical discussion – in opposition to judgments justified by formal, logical or deductive rules. Rather, it makes such a discussion possible and requires critical scrutiny.

Consequently, introducing this form of justification by involvement with objects and data and conceding it to the outputs of DLNs does not amount to a wholesale justification of this technique. The approach of deep learning in general can neither be positively nor negatively assessed. Rather, the deployment of DLNs entails a differentiated assessment and requires a critical discussion of consequences in each case and context. The decisions taken in offices or laboratories are only rarely justified by strict rules and explicit criteria. Often, they primarily are justified by experience and involvement with the data, objects, or persons, but also often biased by prejudices. Hence, the deployment of DLNs does not introduce a totally new lack of transparency, but rather another one. The question is how

these lacks of transparency are reflected in practice or can even be harnessed. If the judging machine would assist the decision, it could be possible to compensate for one lack of transparency by another and create opportunities for critical reflection. If the judging machine were to substitute for human decisions, then the risk of an automated decision process eliminating the room for critically reflecting on the decisions and their effects emerges. Therefore, the deployment of DLNs should be critically monitored where important and far-reaching decisions are made.[41]

Such a monitoring should pay particular attention to the data used to train networks. If the functionality of the network is largely and directly determined by the data without being mediated by common sense or the prejudices of human expertise and if we concede that its judgments can be justified by the involvement with data, the fact that data are rarely neutral or unproblematic, but often biased or prone to systematical error is of crucial importance. The strength of deep learning is to learn from examples, to adapt to data and to get involved with them. However, this strength can turn into weakness if the data is of dubious nature. So in 2016, Microsoft was forced to switch off its Twitter bot @TayandYou within 24 hours: In fact, it effectively learned from other Twitter users, but soon it was taught how to behave like a racist and sexist.[42]

[41] This is not only the case in medical diagnosis or similar applications, but also in credit scoring or evaluation of job applications. For a powerful polemic against the use of mathematical methods and their impact on society, cf. O'Neil (2016).

[42] Cf. https://www.theguardian.com/technology/2016/mar/24/tay-microsofts-ai-chatbot-gets-a-crash-course-in-racism-from-twitter?CMP=twt_a-technology_b-gdntech [last access 6 July 2018].

References

Biran, Or & Cotton, Courtenay (2017). Explanation and Justification in Machine Learning: A Survey. XAI Workshop at IJCAI 2017, Melbourne, Australia. http://www.cs.columbia.edu/~orb/papers/xai_survey_paper_2017.pdf. Accessed 25 June 2018.

Biran, Or & McKeown, Kathleen (2014). Justification Narratives for Individual Classifications. AutoML Workshop at ICML 2014, Beijing, China. http://www.cs.columbia.edu/~orb/papers/justification_automl_2014.pdf. Accessed 21 June 2018.

Biran, Or & McKeown, Kathleen (2017). Human-Centric Justification of Machine Learning Predictions. In: Carles Sierra (ed.), Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence. Main Track (pp. 1461-1467). https://www.ijcai.org/proceedings/2017/0202.pdf. Accessed 25 June 2018.

Buduma, Nikhil (2017). Fundamentals of Deep Learning: Designing Next-Generation Machine Intelligence Algorithms. With contributions by Nicholas Locascio. Sebastopol, CA.

Campbell, Murray, Hoane, A. Joseph & Hsu, Feng-hsiung (2002). Deep Blue. Artificial Intelligence 134, 57-83.

Castelvecchi, Davide (2016). The Black Box of AI. In: Nature 538, 6 October, 20-23.

Ciodaro, T. et al. (2012). Online Particle Detection with Neural Networks Based on Topological Calorimetry Information. Journal of Physics. Conference Series 368, 012030.

Collins, H. M. (1990). Artificial Experts: Social Knowledge and Intelligent Machines. Cambridge, MA, and London.

Collins, Harry & Kusch, Martin (1998). The Shape of Actions: What Humans and Machines Can Do. Cambridge, MA, and London.

Corfield, David (2010). Varieties of Justification in Machine Learning. In: Minds & Machines 20, 291-301.

Dreyfus, Hubert L. (1992). What Computers Still Can't Do: A Critique of Artificial Reason. Cambridge, MA, and London.

Ensmenger, Nathan (2011). Is chess the drosophila of artificial intelligence? A social history of an algorithm. Social Studies of Science 42, 5-30.

Floridi, Luciano (1999). Philosophy and Computing: An introduction. London and New York.

Floridi, Luciano (ed.) (2015). The Onlife Manifesto: Being Human in a Hyperconnected Era. Cham 2015.

Gillies, Donald (1996). Artificial Intelligence and Scientific Method. Oxford.

Ginsborg, Hannah (1999). The Role of Taste in Kant's Theory of Cognition. New York and London.

Goodfellow, Ian, Bengio, Yoshua & Courville, Aaron (2016). Deep Learning. Cambridge, MA, and London.

Graupe, Daniel (2016). Deep Learning Neural Networks: Design and Case Studies. Singapore et al.

Harman, Gilbert & Kulkarni, Sanjeev (2007). Reliable Reasoning: Induction and Statistical Learning Theory. Cambridge, MA, and London.

Hendricks, Lisa Anne, et. al. (2016). Generating Visual Explanations. In: Bastian Leibe et al. (eds.). Computer vision – ECCV 2016. 14th European conference, Amsterdam, The Netherlands, October 11-14, 2016. Proceedings, Part IV (= Lecture Notes in Computer Science, 9908) (pp. 3-19). Cham.

Heßler, Martina (2017). Der Erfolg der „Dummheit": Deep Blues Sieg über den Schachweltmeister Garri Kasparov und der Streit über seine Bedeutung für die Künstliche Intelligenz-Forschung. In: N.T.M. 25, 1-33.

Hsu, Feng-Hsiung (2002). Behind Deep Blue. Princeton and Oxford.

Jordan, M. I. & Mitchell, T. M. (2015). Machine Learning: Trends, Perspectives, and Prospects. In: Science 349.6245, 255-260.

Kant, Immanuel (1781/1998). Critique of Pure Reason. Transl. and ed. by Paul Guyer and Allen W. Wood. Cambridge.

Kant, Immanuel (1790/2000). Critique of the Power of Judgment. Ed. by Paul Guyer, transl. by Paul Guyer and Eric Matthews. Cambridge.

Kulkarni, Sanjeev & Harman, Gilbert (2011). An Elementary Introduction to Statistical Learning Theory. Hoboken, NJ.

LeCun, Yann, Bengio, Yoshua & Hinton, Geoffrey (2015). Deep Learning. Nature 521, 28 May, 436-444.

Leung, Michael K. K., et al. (2014). Deep learning of the tissue-regulated splicing code. Bioinformatics 30, i121-i129.

Ma, Junshui, et al. (2015). Deep Neural Nets as a Method for Quantitative Structure—Activity Relationships. Journal of Chemical Information and Modeling 55, 263-274.

Mahendran, Aravindh & Vedaldi, Andrea (2016). Visualizing Deep Convolutional Neural Networks Using Natural Pre-Images. International Journal of Computer Vision 120.3, 233-255.

Martin, Charles H. & Mahoney, Michael W. (2017). Rethinking Generalization Requires Revisiting Old Ideas: Statistical Mechanics Approaches and Complex Learning Behavior, https://arxiv.org/pdf/1710.09553.pdf [last access 22 November 2018].

Mjolsness, Eric & DeCoste, Dennis (2001). Machine Learning for Science: State of the Art and Future Prospects. Science 293, 14 September, 2051-2054.

Mordvintsev, Alexander, Olah, Christopher & Tyka, Mike (2015). Inceptionism: Going Deeper into Neural Networks. https://ai.googleblog.com/2015/06/inceptionism-going-deeper-into-neural.html. Accessed 27 June 2018.

O'Neil, Cathy (2016). Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy. New York.

Olah, Chris, Mordvintsev, Alexander & Schubert, Ludwig (2017). Feature Visualization: How Neural Networks Build Up Their Understanding of Images. Distill, https://distill.pub/2017/feature-visualization/. Accessed 27 June 2018.

Rashid, Tariq (2016). Make Your Own Neural Network: A Gentle Journey through the Mathematics of Neural Networks. USA.

Ribeiro, Marco Tulio, Singh, Sameer & Guestrin, Carlos 2016. "Why Should I Trust You?" Explaining the Predictions of Any Classifier. KDD 2016, San Francisco, CA. https://arxiv.org/abs/1602.04938. Accessed 21 June 2018.

Schmidhuber, Jürgen (2015). Deep Learning in Neural Networks: An Overview. Neural Networks 61, 85-117.

Schubbach, Arno (2016). Die Genese des Symbolischen: Zu den Anfängen von Ernst Cassirers Kulturphilosophie. Hamburg.

Searle, John (1984). Minds, Brains and Science. Cambridge, MA.

Silver, David, et al. (2016). Mastering the Game of Go with Deep Neural Networks and Tree Search. Nature 529, 28 January, 484-489.

Silver, David, et al. (2017). Mastering the Game of Go without Human Knowledge. Nature 550, 19. October, 354-359.

Smolensky, Paul (1988). On the Proper Treatment of Connectionism. Behavioral and Brain Sciences 11, 1-74.

Wegner, Peter (1998). Interactive Foundations of Computing. Theoretical Computer Science 192, 315-351.

Zeiler, Matthew D. & Fergus, Rob (2014). Visualizing and Understanding Convolutional Networks. In: David Fleet et al. (eds.), Computer vision – ECCV 2014, 13th European Conference Zurich, Switzerland, September 6-12, 2014, Part I (= Lecture Notes in Computer Science, 8689) (pp. 818-833). Cham.