What Teachers Can Learn from Machine Learning

Christian Servin, Olga Kosheleva, and Vladik Kreinovich

Abstract Decades ago, machine learning was not as good as human learning, so many machine learning techniques were borrowed from how we humans learn – be it on the level of concepts or on the level of biological neurons, cells responsible for mental activities such as learning. Lately, however, machine learning techniques such as deep learning have started outperforming humans. It is therefore time to start borrowing the other way around, i.e., using machine learning experience to improve our human teaching and learning. In this paper, we describe several relevant ideas – and explain how some of these ideas are related to fuzzy logic and fuzzy techniques.

1 Introduction

In some application areas – e.g., in the control of many mechanical systems – we know the equations that describe how the state of the system will change under different controls. In such situations, selecting the best control becomes a precise optimization problem – this problem may be difficult to solve, but at least its formulation is precise.

In many other application areas, we do not have equations that would have enabled us to predict the results of different actions. Teaching is definitely one of

Christian Servin
Computer Science and Information Technology Systems Department
El Paso Community College (EPCC), 919 Hunter Dr., El Paso, TX 79915-1908, USA
e-mail: cservin1@epcc.edu

Olga Kosheleva
Department of Teacher Education, University of Texas at El Paso, 500 W. University
El Paso, TX 79968, USA, e-mail: olgak@utep.edu

Vladik Kreinovich
Department of Computer Science, University of Texas at El Paso, 500 W. University
El Paso, TX 79968, USA, e-mail: vladik@utep.edu

such situations. In such situations, we need to rely on the experience – people tried different actions in different circumstances and got different results. Some of this experience have been processed and summarized – formally or informally – by experts. The resulting expert knowledge – often formulated in fuzzy terms – is indeed very helpful, and can be even more helpful if properly applied; see, e.g., [5] and references therein (see also [6]).

However, the amount of experience related to teaching is limited. On the one hand, billions of people are being taught all the time, true, but, on the other hand, interesting conclusions can be made about the effect of different actions only when we try different actions, and not many such experiments are being performed. It is relatively easy to try different ways of controlling a mechanical system: in some cases, a new idea will lead to a better control, in other cases, it will lead to a worse control, no big harm done.

Education is different. You do not want to try untested ideas, ideas that may turn out to make situation worse, on real students. The situation is even worse than in medicine: there, at least, we can try on animals first, but teaching animals is too limited to be useful.

Because of all this, the amount of actual teaching experience that we can use to improve teaching is indeed limited. But there is another source of teaching and learning experience – numerous application of machine learning; see, e.g., [1, 3]. In machine learning, many different algorithms and ideas have been tried, and it is now reasonably clear what worked and what did not. So why not utilize this experience? Yes, machine learning is different from teaching a human being, but a model plane tested in a wind tunnel is also different from the actual plane – but we can still use the wind tunnel experience when designing the actual planes.

Let us see how we can use the experience of machine learning to improve our teaching.

2 First Idea: Take into Account that There Are Very Wrong and Somewhat Wrong Answers

Idea. In teaching – e.g., in teaching arithmetic or other simple mathematical skills – we usually distinguish between correct and wrong answers. The answer that $5 \times 5 = 55$ is as wrong as the answer that $5 \times 5 = 26$. Both answers will lead to 0 points. The overall grade on a homework or on a test is based on the number of examples in which the answer is correct.

This may sound reasonable – since this is what most teachers have been doing for millennia, but this is not what the current state-of-the-art machine learning algorithms – like deep learning [3] – are recommending. At the initial stages of learning, we may not get any correct answers, but the algorithms distinguish between answers which are closer to the correct ones and answers which are further away from the correct answers – in other words, between answers which are somewhat wrong and answers which are very wrong. Crudely speaking, answers which are only some-

what wrong still get some points (i.e., in precise terms, contribute to the value of the corresponding objective function), while answers which are very wrong contribute much less. This idea has shown to work; see, e.g., [4, 9].

The big question is how to gauge the difference, i.e., in fuzzy terms, what membership function to select for the corresponding "degree of wrongness" (or, alternatively, "degree of correctness"). In the past, in effect, the Euclidean distance between the correct and actual answers was used – i.e., equivalently, the sum of the squares of the differences. However, it turned out that other measures – such as Kullback–Leibler relative entropy [3] – lead to better learning. This is an area where human expertise may help.

Historical comment. Two of us (OK and VK) are originally from the former Soviet Union, so we are familiar with the differences between more and less serious mistakes. In one of Lenin's polemic papers that we had to study, he confesses that he also makes serious mistakes, sometimes even mistakes similar to claiming that 2 plus 2 is 5, but when his opponents make mistakes, their mistakes are often like claiming that 2 plus 2 is a candle.

Another Russian example is related to the distinction between serious errors and measurement errors – which are usually very small. In English, in both cases, the same word "error" is used, which sometimes confuses the general public. In Russian, these two concepts are described by two different words: "oshibka" for a serious error and Bible-motivated "pogreshnost'" (literally meaning "small (not so severe) sin") for measurement error.

3 Second Idea: Asking Why-Questions, Not Just Checking Where Answers are Correct

Idea. At present, as we have mentioned, the most efficient machine learning techniques are neural networks – in their current form of deep neural networks. Originally, the most widely used neural networks were 3-layer ones, in which there are only two processing layers, one of which is linear. For such networks, the difference between the value that we want the network to learn and the value that the trained-so-far network produces – this difference practically directly changes the state of the neurons. In such networks, if we want to know why a certain value was produced by the network, we can easily understand that by using the parameters of these neurons.

To be more precise, based on the inputs $x_1, \ldots, x_n$, we first compute the values $y_k = s_0\left(\sum_{i=1}^{n} w_{ki} \cdot x_i - w_{k0}\right)$, where $s_0(z)$ is a non-linear function called *activation function* and the parameters w_{ki} describes the state of each neuron. Once the value y_k are computed, we compute the final result $y = \sum_{k=1}^{K} W_k \cdot y_k - W_0$.

However, lately, it turned out that deep neural networks – which consist of many layers – lead to a more efficient learning. In such networks, if we try to trace why a given value was produced, we can easily trace it only to the previous layer – and,

of course, to explain why the corresponding values were produced on this layer, we need to go one more layer back, etc.

How can this be used for actual teaching? Traditional assessment, when we check the student's knowledge by making sure that they give us correct answers to different questions, is similar to the traditional neural network case. A natural analog of the deep learning would be not only to check the answer, but also to go deeper, to ask why the student obtained this answer – i.e., what thinking and what methods he/she used, then ask what was the motivation behind these methods, etc. This is definitely more time-consuming – similarly to the fact that deep learning requires much more computational resources than the traditional neural networks [3] – but this will hopefully lead to better learning. This way, we will know, e.g., why the student answered that 5 times 5 is 55: did he/she make an honest arithmetic mistake or is this student under a false impression that – similar to the notation *ab* for multiplication – to compute a product, we need to place the multiplied numbers together.

Comment. This idea also fits well with our Russian experience – this time, with the experience of oral exams for math department classes. Once you successfully prove a theorem – which usually means using other previously studied theorems – the professor who takes this exam would often ask to prove these auxiliary theorems, then to prove the theorems used in their proof, etc. – all the way to the basics (or, sometimes, to a gap in the student's knowledge).

Relation to explainability. So far, we have described what we can borrow from successes of deep learning, but we can also learn from problems of deep learning, one of which is the lack of explainability. This problem is similar to what we have when assessing student's knowledge: we see their results – which are not always correct – but we do not understand why. If we explicitly ask the why-questions, we will understand the reason for these mistakes – and this will help teach the correct way to students.

4 Third Idea: Let Us Be Positive

Idea. Traditional neural networks used the so-called sigmoid activation function

$$s_0(z) = \frac{1}{1+\exp(-z)},$$

borrowed, by the way, from biological neurons. In deep learning, it turned out that learning becomes much better if we use the *rectified linear* activation function:

$$s_0(z) = \max(0, z).$$

What is the difference?

- with the sigmoid activation function, we take into account both negative ($z < 0$) and positive ($z > 0$) inputs;
- in contrast, if we use the rectified linear activation function, we completely ignore negative signals and only take positive ones into account.

How can we use this idea in teaching? Naturally – ignore negative feeling and emotions and concentrate only on positive ones. This idea is in perfect accordance with the psychologists' ideas of the power of positive thinking; see, e.g., [2, 7].

Comment. While at present, the function $s_0(z) = \max(0, z)$ is largely associated with deep learning, this function was known and used much much earlier: this function describes a *diode*, an electronic device used to transform the highly oscillating amplitude-modulated signal into the original lower-frequency signal – e.g., corresponding to radio-transmitted speech. For teaching, the idea to avoid sometimes observed fast-mood-oscillation is also good: such mood oscillations create too much stress and thus, hinder the students' learning.

5 Fourth Idea: Making Learning More Robust

One of the main challenges of deep learning is that it is often not robust: a minor change to a picture of a cat – a change invisible to a human eye – can cause the neural network to confidently classify this picture as a dog. To avoid such mistakes, a natural idea is to train the network not only on the original examples, but also on modified versions of these examples.

A similar idea can be helpful in teaching. For example, in school math, students often learn the "rules" describing when to use what arithmetic operation: e.g., that "all" means addition. In general, it does, but what happens sometimes is that they uncritically use this rule all the time, even when "all" means something else. To avoid such false uses, it is important, in addition to standard word problems, to train students on modified versions of these problems, where the formulation of the problem is reworded in different ways.

6 Fifth Idea: Averaging – This Is Not What You Think

Yes another idea of deep learning is "averaging". It means that instead of using the whole neural network to learn, we divide it into subnetworks; each of them learns the same – or even different – patterns, and then these different learning results are combined ("averaged").

A natural application to teaching is that instead of the teacher teaching the same material to the whole class, students are divided into groups, groups learn by themselves – with the teacher's help – and then all get together and learn from each other. Learning is never absolutely even, so for each topic, one of the groups learned better

– thus all the groups have something to teach to each other, again with the teacher's help.

7 What Else?

So far, we have listed several features of deep learning for which it is clear how we can use them in teaching. There are, of course, many other features and challenges, and how to use them is not yet clear. For example, one of the big challenges of deep learning is a possible bias. Indeed, a neural network simply learns from real-life examples. Since some real-life examples are biased – e.g., there is still bias against women in some areas, bias against people of certain race or ethnicity in some places – the neural network will (and did), unfortunately, learn some of this bias. It is not absolutely clear how to avoid this bias, although many ideas have been proposed and are researched – see, e.g., [8] for one of the possible directions. It is also not clear how to apply these ideas in teaching – and there definitely is bias.

For example, in some Computer Science departments there is a certain bias against students who come from community colleges. For some not-yet-perfect community colleges, this may be a justified concern, but when extrapolated to all community colleges, including ones that provide good-quality education, this becomes a bias. There is sometimes bias against students who are not native speakers of English – these students may know math well and are very skilled in programming, but many of them have trouble with word problems, especially problems that deal with American realia like baseball or details of the American complicated election system. It would be great to see how we can use anti-bias techniques – which are currently actively developed in AI – against bias in teaching.

And, of course, all the above qualitative ideas need to be developed into more quantitative solutions.

Acknowledgments

This work was supported in part by the National Science Foundation grants 1623190 (A Model of Change for Preparing a New Generation for Professional Practice in Computer Science), and HRD-1834620 and HRD-2034030 (CAHSI Includes). It was also supported by the program of the development of the Scientific-Educational Mathematical Center of Volga Federal District No. 075-02-2020-1478.

The authors are thankful to anonymous referees for valuable suggestions.

References

1. C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, New York, 2006.
2. E. Chang, *Optimism & Pessimism: Implications for Theory, Research, and Practice*, American Psychological Association, Washington, DC, 2001.
3. I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, Cambridge, Massachusetts, 2016.
4. A. Jansen, *Rough Draft Math: Revising to Learn*, Stenhouse Publishers, Portsmouth, New Hampshire, 2020.
5. O. Kosheleva and K. Villaverde, *How Interval and Fuzzy Techniques Can Improve Teaching*. Springer, Cham, Switzerland, 2018.
6. V. Kreinovich, "A review of [5]", *Journal of Intelligent and Fuzzy Systems*, to appear.
7. M. E. P. Seligman, *Learned Optimism: How to Change Your Mind and Your Life*, Vintage, New York, 2006.
8. C. Servin and V. Kreinovich, "Is there a contradiction between statistics and fairness: from intelligent control to explainable AI", *Proceedings of the Annual Conference of the North American Fuzzy Information Processing Society NAFIPS'2020*, Redmond, Washington, August 20–22, 2020.
9. C. Servin, O. Kosheleva, and V. Kreinovich, "A review of [4]", *Journal of Intelligent and Fuzzy Systems*, to appear.