Is Constructivism Sufficient for Teaching? Experience of Machine Learning Says "Not Always"

Christian Servin, Olga Kosheleva, and Vladik Kreinovich

Abstract One of the main direction in modern pedagogy is *constructivism*, when instead of explicitly teaching general rules and algorithms, the instructor provides the students with a well-design sequence of examples, based on which the students can easily reconstruct the general rules. This direction has been very successful – and its success seems to be confirmed by spectacular successes of modern AI, successes based on a similar idea – that teaching computer examples from which the computer can reconstruct the rules is much more productive than explicitly teaching the rules. However, our experience of teaching complex rules and algorithms shows that sometimes, teaching rules first leads to better results, In this paper, we show that several recent machine learning results show a similar tendency – that for complex rules and algorithms, it is sometimes beneficial to explicitly teach computer the rules.

1 Constructivism in Education: What Is It, Its Successes, and Its Indirect Machine Learning Support

What is constructivism in education. Education needs to take into account that our memory is not perfect. If we are simply asked to remember a rule or an algorithm, there is a high probability that we will forget it when time comes to use this rule.

Christian Servin
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, Texas 79968, USA, e-mail: olgak@utep.edu

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

So, to make sure that students can still recover the rule when needed, a natural idea is to provide them not just with the rule itself, but with a lot of additional material from which the students can later derive the rule – even when forget the rule itself. It is also desirable to make sure that the students can reconstruct even if they forget some part of this additional information.

What kind of additional information is available? To answer this question, let us recall how general rules for solving a class of problems are formed. Usually, at first, we have solutions to individual problems. When many such solutions have been accumulated, people notice that there is a general pattern in all these individual solutions – and thus, a general rule is formulated. So, a natural idea is to teach not only the rule itself, but also examples that can lead to this rule.

This is usually done in teaching, but in many cases, the examples are given *after* the rule has been formulated. The problem with this approach is that when the students already know the rule, they will not pay much attention to specific examples – and they will most probably not understand how these examples generalize to a general rule. So, it looks like a better idea is to *first* teach the examples – selected in such a way that it will be reasonably easy to extract the rules from these examples – and only then teach the rule. Actually, an even better idea is *not* to teach the rule, but – since it is easy for the students to derive this rule – to let the students themselves form this rule based on the examples. This way, they will go through an experience of deriving the general rule. So, they will have some memories of what and how they did it, and these memories will help them derive the rule next time – when they have forgotten the rule but still remember some of the examples.

In this approach, the students *construct* the rules themselves, so this approach to education is known as *constructivism*; see, e.g., [11].

Comment. This term may be somewhat confusing to many mathematicians, since in mathematics, the same word *constructivism* is used to describe a direction of mathematical research when we limit ourselves only to objects and relations that can be actually constructed; see, e.g., [1, 2, 3, 4, 5, 6, 12, 13, 15, 22]. For example, instead of considering all possible real numbers, we only consider real numbers x for which there is an algorithm that, given a natural number n, computes a rational number r_n which is 2^{-n}-close to x.

Constructivism in education is different, but methodologically, it is a very similar idea: we limit ourselves to concepts and rules that can be constructed – this time, constructed from examples.

Successes of constructivism in education. Constructivism is, at present, one of the main directions in pedagogy. Many studies has shown that its use leads to more effective learning – when in the long term, students can recall and use a larger part of their knowledge.

Moreover, this direction encourage researchers and practitioners to come up with a natural way to derive the existing rules – even those that, at first, sound like weird tricks. These efforts has made several rules and algorithms more natural. This is similar to constructivism in mathematics: in mathematics, constructivism helped to come up with new algorithms for computing different important mathematical objects – and thus led to important practical applications.

Indirect machine learning support of constructivism in education. The main ideas behind constructivism in education seems to be strongly supported by the current spectacular successes in AI. In the past, most AI was focused on teaching rules to a computer system. This approach started with initial successes that encouraged optimistic predictions – that we will soon have chess-playing computers and thinking robots, computers translating from one natural language to another, talking to the users in natural languages, etc. Unfortunately, with rules-based AI, these predictions did not materialized.

Instead, all these goals were achieved when researchers decided, instead of teaching rules to a computer, to simply give the computer examples – from which the computer can reconstruct the general rules. This AI approach is known as *machine learning*. The current spectacular successes of the machine learning approach seem to indicate that a similar constructivism approach to human learning is more effective than teaching the rules to the students.

2 (Indirect) Doubts

All three authors are instructors, and we are using the constructivism approach as much as possible – and in many cases, we have seen that using a constructivism approach has improved students learning. However, there seems to be some problems – probably caused by overdoing it. These problems have occurred when one of us (VK) has been teaching two theory of computation classes: a graduate one and an undergraduate one (called *Automata*). Both classes involve several complex and not-easy-to-justify rules and algorithms.

These classes are not about memorization, they are about using the learned skills to come up with new applications of the algorithms and of the proof ideas. To avoid the need to memorize the material by heart, on each test, the instructor allows students to have a few-page "cheat sheets" that they can prepare beforehand. The instructor's advice is to use these cheat sheets to write down the needed rules and algorithms – and also examples of their applications. Many students do it, and most of these students perform very well on the corresponding tests.

However, many other students, in the spirit of constructivism, only place examples of their cheat sheets, thinking that they will be able, during the test, to reconstruct the rules from these examples. In general, the overall performance of these students is much worse than the performance of the students who explicitly write down the rules on their cheat sheets.

There was no such drastic difference in many other classes, where rules and algorithms are not as complex. Thus, the need to teach rules is probably a phenomenon explicitly related to the complexity of the material. All this has made us think that maybe in some cases, constructivism approach needs to be supplements with a more traditional teaching-rules-first approach.

How can we check whether this is a good idea?

3 New Machine Learning Results and What They Tell Us About Education

Let us look at what machine learning says. As we have mentioned, a large amount of support for constructivism approach comes from the current machine learning successes. So, we decided to look into whether in some cases, machine learning experience has shown a similar need for teaching rules in addition to providing examples.

And indeed, there are examples when also teaching rules helped machine learning. We immediately found several studies that show that in some cases, also teaching rules – and/or incorporating rule-based techniques like stacks into a neural network – helps machine learning. Interestingly, most of these examples were related to reasonably complex rules and algorithms – of the type that we teach in theoretical computer science classes; see, e.g., [7, 8, 9, 10, 14].

Another example is that the use of fuzzy rules helps machine learning; see, e.g., [16, 17, 18, 19, 20, 21] and numerous papers from this volume.

Conclusion. Both the experience of our own teaching and an experience of machine learning show that for complex topics, it is beneficial to sometimes supplement the constructivism examples-first approach with a more traditional rules-first approach.

Acknowledgments

This work was supported by the AT&T Fellowship in Information Technology, by the Institute for Risk and Reliability, Leibniz Universitaet Hannover, Germany, by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) Focus Program SPP 100+ 2388, Grant Nr. 501624329, by the European Union under the project ROBOPROX (No. CZ.02.01.01/00/22 008/0004590), by the Center of Excellence in Econometrics, Faculty of Economics, Chiang Mai University, Thailand, by the Ho Chi Minh City University of Banking, Vietnam, and by Thang Long University, Hanoi, Vietnam.

The authors are greatly thankful to Brian DuSell for valuable discussions.

References

1. O. Aberth, *Precise Numerical Analysis Using C++*, Academic Press, New York, 1998.
2. M. J. Beeson, *Foundations of constructive mathematics*, Springer-Verlag, N.Y., 1985.
3. E. Bishop, *Foundations of Constructive Analysis*, McGraw-Hill, 1967.

4. E. Bishop and D. S. Bridges, *Constructive Analysis*, Springer, N.Y., 1985.
5. D.S. Bridges, *Constructive Functional Analysis*, Pitman, London, 1979.
6. D. S. Bridges and S. L. Vita, *Techniques of Constructive Analysis*, Springer-Verlag, New York, 2006.
7. B. DuSell and D. Chiang, "Learning context-free languages with nondeterministic stack RNNs. In: R. Fernández and T. Linzen (eds.), *Proceedings of the 24th Conference on Computational Natural Language Learning CoNLL 2020, Online, November 19–20, 2020*, Association for Computational Linguistics, 2020, pp. 507–519, doi: 10.18653/v1/2020.conll-1.41, https://aclanthology.org/2020.conll-1.41
8. B. DuSell and D. Chiang, "Learning hierarchical structures with differentiable nondeterministic stacks", In: *Proceedings of the International Conference on Learning Representations ICLR 2022, Online, April 25–29, 2022*, https://openreview.net/forum?id=5LXw QplBiF
9. B. DuSell and D. Chiang, "The surprising computational power of nondeterministic stack RNNs", *Proceedings of the Eleventh International Conference on Learning Representations ICLR 2023, Kigali, Rwanda, May 1–5, 2023*, https://openreview.net/forum?id=o58JtGDs6y
10. B. DuSell and D. Chiang, "Stack attention: Improving the ability of transformers to model hierarchical patterns", *Proceedings of the Twelfth International Conference on Learning Representations, Vienna, Austria, May 7–11, 2024*, https://openreview.net/forum?id=XVhm3X8Fum
11. C. T. Fosnot (ed.), *Constructivism: Theory, Perspectives, and Practice*, Teachers College Press, New York, 2005.
12. V. Kreinovich, A. Lakeyev, J. Rohn, and P. Kahl, *Computational complexity and feasibility of data processing and interval computations*, Kluwer, Dordrecht, 1998.
13. B. A. Kushner, *Lectures on Constructive Mathematical Analysis*, Amer. Math. Soc., Providence, Rhode Island, 1984.
14. Y. Lakretz, Th. Desbordes, D. Hupkes, and S. Dehaene, "Can transformers process recursive nested constructions, like humans?" In: N. Calzolari, C.-R. Huang, H. Kim, J. Pustejovsky, L. Wanner, K.-S. Choi, P.-M. Ryu, H.-S. Chen, L. Donatelli, H. Ji, S. Kurohashi, P. Paggio, N. Xue, S. Kim, Y. Hahm, Z. He, T. Kyungil Lee, E. Santus, F. Bond, and S.-H. Na (eds.), *Proceedings of the 29th International Conference on Computational Linguistics COLING 2022, Gyeongju, Republic of Korea, October 12–17, 2022*, pp. 3226–3232, https://aclanthology.org/2022.coling-1.285
15. M. Pour-El and J. Richards, *Computability in Analysis and Physics*, Springer-Verlag, New York, 1989.
16. J. Viaña, K. Cohen, A. Ralescu, S. Ralescu, and V. Kreinovich, "Localized Learning: A Possible Alternative to Current Deep Learning Techniques", In: O. Castillo and P. Melin (eds.), *New Perspectives on Hybrid Intelligent System Design based on Fuzzy Logic, Neural Networks and Metaheuristics*, Springer, 2023, pp. 465–473.
17. J. Viaña, S. Ralescu, K. Cohen, A. Ralescu, and V. Kreinovich, "Extension to multidimensional problems of a fuzzy-based explainable and noise-resilient algorithm", In: M. Ceberio and V. Kreinovich (eds.), *Decision Making under Uncertainty and Constraints: A Why-Book*, Springer, Cham, Switzerland, 2023, pp. 289–296.
18. J. Viaña, S. Ralescu, V. Kreinovich, A. Ralescu, and K. Cohen, "Single Hidden Layer CEFYDRA: Cluster-first Explainable FuzzY-based Deep self-Reorganizing Algorithm", *Proceedings of the 2022 Annual Conference of North American Fuzzy Information Processing Society*, Halifax, Nova Scotia, Canada, May 31 – June 3, 2022, pp. 298–307.
19. J. Viaña, S. Ralescu, V. Kreinovich, A. Ralescu, and K. Cohen, "Multiple Hidden Layered CEFYDRA: Cluster-first Explainable FuzzY-based Deep self-Reorganizing Algorithm", *Proceedings of the 2022 Annual Conference of North American Fuzzy Information Processing Society*, Halifax, Nova Scotia, Canada, May 31 – June 3, 2022, pp. 308–322.
20. J. Viaña, S. Ralescu, V. Kreinovich, A. Ralescu, and K. Cohen, "Initialization and Plasticity of CEFYDRA: Cluster-first Explainable FuzzY-based Deep self-Reorganizing Algorithm", *Proceedings of the 2022 Annual Conference of North American Fuzzy Information Processing Society*, Halifax, Nova Scotia, Canada, May 31 – June 3, 2022, pp. 323–335.

21. J. Viaña, S. Ralescu, A. Ralescu, K. Cohen, and V. Kreinovich, "Explainable Fuzzy Cluster-based Regression Algorithm with Gradient Descent Learning", *Complex Engineering Systems*, 2022, Vol. 2, http://dx.doi.org/10.20517/ces.2022.14
22. K. Weihrauch, *Computable Analysis*, Springer Verlag, Berlin, 2000.