https://doi.org/10.48417/technolang.2022.02.02
Research article

# Instructing to and Instructing in: Two Paradigms of Instruction

Danka Radjenović (✉)
University of Koblenz-Landau, Fortstraße 7, 76829 Landau, Germany
radjenovic@uni-landau.de

## Abstract

In my contribution, I appropriate the distinction made in English between “instructing to” and “instructing in” in order to differentiate between the mode of instruction characteristic of technical processes – instructing to – which is more akin to order and command, and a mode of instruction closer to teaching – instructing in. Talk of instruction covers a spectrum of cases, with the technological paradigm of “instructing to” being on the one end of the spectrum, as opposed to the open-ended process of “instructing in” on the other end. More precisely, the former paradigm is that of an automaton, “a machine which performs a range of functions according to a predetermined set of coded instructions”, whereas the latter can be imagined as an “open-ended” process of instruction, such as language instruction (following Cavell’s take on Wittgensteinian scenes of instruction). While the model of instruction pertaining to technology is led by the goal of achieving automatisation, language instruction runs counter to the idea of language usage running in an automatic way – even though the process of instruction itself includes elements of drill and repetition. The goal of becoming a competent language user is in a way never achieved fully, since it is always possible to discover new ways of expressing the same things or even to discover new words and expressions. As the distinction elaborated in this contribution helps to show, it is thus not appropriate to talk of instructing a machine in singing, but it will be possible to instruct it to produce sounds that remind of singing. Taking the other direction, however, reveals that technological systems can instruct humans to behave in certain “automatic” ways, leaving it to education to instruct present and future generations in becoming competent users of different technologies.

**Keywords:** Instruct to; Instruct in; Automatisation; Teaching; Embeddedness; Wittgenstein; Cavell

**Citation:** Radjenović D. (2022). Instructing to and Instructing in: Two Paradigms of Instruction. *Technology and Language*, 2022, 3(2), 6-13. https://doi.org/10.48417/technolang.2022.02.02

УДК 008: 002
https://doi.org/10.48417/technolang.2022.02.02
Научная статья

# Инструкция к … и инструкция о … : Две парадигмы инструкции

Данка Радженович (✉)
Университет Кобленц-Ландау, Фортштрассе 7, 76829 Ландау, Германия
radjenovic@uni-landau.de

## Аннотация

В своей статье я использую различие, проведенное в английском языке между “instructing to” (“инструкция к …”) и “instructing in”, (“инструкция о…”), чтобы различать способ обучения, характерный для технических процессов – instructing to – который больше похож на приказ и команду, и способ обучения ближе к обучению – “instructing in, инструктирование. Разговор об обучении охватывает спектр случаев, причем технологическая парадигма “instructing to” находится на одном конце спектра, в отличие от открытого процесса “instructing in” на другом конце. Точнее, первая парадигма – это парадигма автомата, “машины, которая выполняет ряд функций в соответствии с заранее определенным набором закодированных инструкций”, тогда как вторую можно представить как “открытый” процесс обучения, такой как языковое обучение (в соответствии с подходом Кавелла к витгенштейновским сценам обучения). В то время как модель обучения, относящаяся к технологии, направлена на достижение цели автоматизации, обучение языку противоречит идее автоматического использования языка, даже если сам процесс обучения включает элементы тренировки и повторения. Цель стать компетентным пользователем языка никогда не достигается полностью, поскольку всегда можно открыть новые способы выражения одних и тех же вещей или даже открыть новые слова и выражения. Как помогает показать различие, разработанное в этом вкладе, неуместно говорить об обучении машины пению, но можно научить ее производить звуки, напоминающие пение. Однако с другой стороны, мы обнаруживаем, что технологические системы могут научить людей вести себя определенным “автоматическим” образом, оставляя образованию обучать нынешнее и будущие поколения тому, как стать компетентными пользователями различных технологий.

**Ключевые слова:** Инструкция; Автоматизация; Обучение; Встроенность, Витгенштейн; Кавелл

**Для цитирования:** Radjenović D. Instructing to and Instructing in: Two Paradigms of Instruction // Technology and Language. 2022. № 3(2). P. 6-13. https://doi.org/10.48417/technolang.2022.02.02

# INTRODUCTION

We are all familiar with the composition of a cooking recipe, which contains a set of ingredients and instructions on how to prepare a meal, using the ingredients listed. And every now and then we read user manuals, following the instructions contained in them. In some other cases we can ourselves figure as “instructors” – when teaching another person how to cook or how to operate a machine or vehicle, to name only a few examples. Instructions can be either verbal (expressed in sentences of natural languages or in speech acts), formal (sets of symbols in programming languages) or non-verbal (hand signs, gestures).

Despite its familiarity within the everyday life and its special prominence in the context of using technology, the topic of instructing and instruction has not as yet been thoroughly explored in the area of philosophy of technology. In order to contribute to opening a discussion about instructions, I would like to present some preliminary ideas about the ways in which we can approach this topic. In this paper I introduce and elaborate on two distinct paradigms of instruction: *instructing to* and *instructing in*. They differ in several aspects: as regards their procedures, the context of application and overall goals/purposes.

*Instructing to* is most prominently found in the area of programming, in the cases where a machine, application, device, or an entire system is instructed to behave in a certain way, performing tasks or solving problems. The way the instruction works is rather straightforward: there is a clearly defined task and distinct steps that need to be completed in order for the task to be fulfilled. The regularity and routinized processes are at the core of this kind of instruction, since its success largely depends on the exact execution of instructions, that should be formulated in an unambiguous way. The possibility of variation or deviation has to be previously integrated into the instructions. In the first section of this paper I will introduce several cases of *instructing to*, in the area of human-machine interaction, but also in the interaction between human agents. The applicability of the paradigm in the case of molecular biology will also be presented.

*Instructing in* can be – most generally speaking – found in the field of teaching, where a skill is to be mastered or knowledge is being transmitted. This is a rather open-ended kind of instruction, where we cannot definitively say when the last stage has been reached. In my paper *instructing in* will be elaborated in the second section, where I will focus on the example of teaching and learning a language, by looking into this process from the perspective of both the person teaching and the person being taught. Furthermore, it will be assumed that even though there is a goal that is to be reached when we engage in this kind of instruction, this goal can never be fully attained, as the point is not simply to complete a task, but either to become *good at something* or to gain specific insight or expertise, which is accompanied by certain independence or autonomy in exercising it, that can only be the result of a long-term training process.

The goal of introducing the distinction between *instructing to* and *instructing in* is to capture two distinct processes and their specific features, as well as to clarify whether this distinction is well-founded.

## THE PARADIGM OF *INSTRUCTING TO*

The first paradigm of instruction is mostly found in the context of technology, broadly speaking: both as a feature of human-machine interaction, as well as within the interaction between machines. In its purest form it is exemplified by an automaton – "a machine which performs the range of functions according to a predetermined set of coded instructions" (Rangra & Madhusudan, 2016). Automatic processes unfold in a predetermined way, where any variation or divergence is either also predetermined (hence part of the instructions) or otherwise indicates an interruption, error, or any kind of failure in the process. The underlying scheme of this kind of instruction is: command – execute – repeat. Typically, the goal of this first type of instruction is to enable the performance of different functions or a fulfilment of a task that a specific machine or device is designed to fulfil. The process of executing instructions, which are normally formulated as commands, is directed at fulfilling well-defined tasks. The success of the process depends both on the precision or exactness of instructions, as well as on well-defined tasks or functions that are supposed to be completed.

This is why *instructing to* is characteristic of computer programmes, or of programming generally. According to a common definition: "a computer program is a detailed plan or procedure for solving a problem with a computer; more specifically, an unambiguous, ordered sequence of computational instructions necessary to achieve such a solution." (Gregersen, 2021). What is important here is that the computer program gives orders to a computer processor, because it can be unambiguously translated into exact instructions in machine language. A group of such orders or commands for the central processing unit is called an *instruction set*. They enable the central processing unit to perform tasks. There are different kinds of instruction sets, some of which are more complex than others. One example of a single instruction can be a single *add* command: "A single instruction can initiate multiple actions by the computer, such as a single *add* command launching multiple memory access load and store instructions" (Kivan, 2022). Apart from that "instruction sets work with other important parts of a computer, such as compilers and interpreters. Those components translate high-level programming code into machine code that the processor can understand" (Kivan, 2022). What is apparent from the above definitions is that instruction sets have to be embedded into the entire makeup of the computer, in order to make possible the completion of certain tasks or functions. In order to be understood by the processor, programs have to be translated into instructions.

The importance of translation for *instructing to* becomes especially clear in the cases in which this paradigm had found its way into other fields, for example into molecular biology. When describing processes at the molecular level, it is common to say that DNA contains “instructions” for essential biological processes, such as protein synthesis. Analogous to the case of programming, the DNA code must be interpreted and translated, via mRNA and other intermediary steps, in order to be enacted in the cell. These features of *instructing to* show the process of translation to be its complementary process, at least in the two cases that were presented here: computer programmes and the DNA code.

The two mentioned cases of *instructing to* may suggest that this paradigm is only found in the context of programming – including the (metaphorical) application of programming to other fields. However, the case of *instructing to* is by no means limited to machine language or machine-to-machine communication. In the realm of interaction between human agents there are numerous cases where *instruction to* is instantiated. The example of partaking in traffic – either as a pedestrian, bicycle rider, car driver, or a user of any other means of transport – can serve to illustrate this case. In cases where the regulation by means of the system of traffic lights is not available, or for any other reason cannot be relied upon to regulate the traffic, there is a human agent – traffic policeman – regulating the flow of traffic at major busy crossroads, by using his arms and hands. The hand-signs that the traffic policeman is using are instructions – in the sense of *instructing to*. Such instructions are embedded into the broader context of traffic rules and driving tests, which makes it possible for participants in the traffic to understand the instructions given by the traffic policeman and to spontaneously act according to them in new situations. The act of translation, which was necessary in both previous cases, is here replaced by previous training – part of which consists in getting acquainted with the rules of the traffic system. We will see in the next section how this aspect of training features in the second paradigm of instruction, *instructing in*.

## THE PARADIGM OF *INSTRUCTING IN*

The second paradigm can be best introduced by looking at the process of teaching and learning. I have chosen the example of a child learning a language, thus becoming in time a competent speaker and being introduced into the community of language speakers.

In explicating the paradigm with the help of this example, I follow Stanley Cavell’s reading of Wittgensteinian *scenes of instruction*, which are prominent in the *Philosophical Investigations*.

In these scenes we always see an instructor/teacher and a pupil/student focusing on a certain task or theme that the student is being instructed in. Normally the teacher will show the student the first steps of the task – for example how to continue a series of natural numbers according to a certain rule. After a while the student will be required to go on with the series without teacher's assistance, thus demonstrating the ability to continue the series on his or on her own and thereby of having mastered the application of the rule generating the series. Wittgenstein is especially interested in all the ways in which this instruction process can "go wrong". These are discussed under the general heading of "rule-following" and cover a much broader spectrum of questions than those pertaining to instruction. When it comes to the role of instruction in these examples, one can say that part of the instruction process does consist in *instructing to* – the student is instructed to write one number after another, or (to take the example of instruction in languages) demonstrate the ability to formulate a sentence according to grammatical rules. What makes the examples so interesting is the following: every time the teacher and the student reach a certain point at which the student needs to go on without teacher's assistance. At that moment the student might need to make a sort of a "leap" from already familiar cases to completely new ones. Cavell has described this as "anxiousness…upon which instruction may founder: an awareness of the point at which the path of our communication depends upon your taking the next step, unaided by anything more from me save my belief in your readiness to take it. It is the mark of a good teacher in certain domains to know when to stop prompting, domains in which further knowledge is earned not through further drilling but through proper waiting. It is a different form of exercise. People are not equally good at this, certainly teachers are not equally good; but one can learn to be better" (Cavell, 1999). The crucial thing about *instructing in* is that it requires this "leap" to happen in order for it to be successful. In most cases this is nothing extraordinary and perhaps one can even say that it happens naturally. Still, it marks one of the central differences between *instructing to* and *instructing in*. Perhaps we can say that *instructing in*, when successful, allows the instructed party to leave the instruction behind. If someone can continue on their own, without being told what the next step is or how to conduct it, then there is no need to be instructed. The goal is to attain mastery of a practice, whether that practice is dancing, playing an instrument, building houses, or speaking a language.

The second major difference between *instructing to* and *instructing in* concerns the kind of embedding that is present in both cases. We have seen that *instruction to* depends for its workings either on translation, or on its embedding in a system of rules. This gives rise to the question: What kind of embedding is required for *instructing in* to take place? In order to give an answer to this, I will one more time refer to Cavell's reading of the scenes of instruction, in the case in which a child is learning its mother tongue: "Instead, then, of saying either that we *tell* beginners what words mean, or that we *teach* them what objects are, I will say: We initiate them, into the relevant forms of life held in language and gathered around the objects and persons of our world. For that to be possible, we must make ourselves exemplary and take responsibility for that assumption of authority; and the initiate must be able to follow us, in however rudimentary a way, *naturally* (look where our finger points, laugh at what we laugh at, comfort what we comfort, notice what we notice, find alike or remarkable or ordinary what we find alike or remarkable or ordinary, feel pain at what we feel pain at, enjoy the weather or the notion we enjoy, make the sounds we make); and he must *want* to follow us (care about our approval, like a smile better than a frown, a croon better than a croak, a pat better than a slap). ‘Teaching’ here would mean something like ‘showing them what we say and do’, and ‘accepting what they say and do as what we say and do’, etc.; and this will be more than we know, or can say” (Cavell, 1999). The kind of embedding that is depicted here encompasses the entire way of living in which a certain practice takes place. Cavell describes the first steps of being instructed in a language (this language being one’s mother tongue) as being initiated “into the relevant forms of life held in language and gathered around the objects and persons of our world”. This kind of embedding provides both the instructor and the person being instructed with the possibility to reach the stage (be it one or several stages) at which the teacher can stop the instruction (stop prompting, requesting), so as to allow the other to take the next step on their own. Only then can the instructing process fulfil its purpose.

## CONCLUSION

The two paradigms of instruction are indeed different paradigms. They cannot be “translated” into one another. If the goal of *instructing to* is to reach automatisation, the goal of *instructing in* is to become autonomous when engaging in a certain practice. These are very different goals. And even though *instructing in* includes *instructing to* at its various stages, it is still not possible to reduce *instructing in* to *instructing to*. At least for now, it is not possible to instruct a machine or a robot in singing; one can only instruct it to produce sounds similar to singing. It remains to be seen whether the developments in the field of machine learning and artificial intelligence in general can ever bring about the overcoming of this difference.

## ACKNOWLEDGMENT

I would like to thank the editors and guest-editors of this *Special Issue* for their insightful comments, corrections and suggestions. In addition to that, I would like to thank my colleague Simon Schüz for (proof)reading the manuscript and offering invaluable suggestions. Even if it was not possible for me in the present paper to consider all the points they raised, I look forward to elaborating on them in further work on this topic.

## REFERENCES

Cavell, S. (1999). *The Claim of Reason. Wittgenstein, Scepticism, Morality, and Tragedy*. Harvard University Press.

Gregersen, E. (2021). Computer Program. In *Encyclopedia Britannica*. https://www.britannica.com/technology/computer-program

Kivan, P. (2022). What is an instruction set? *TechTarget*. https://www.techtarget.com/whatis/definition/instruction-set

Rangra, R. & Madhusudan, A. K. Sharma (2016). Natural language parsing: Using finite state automata, In *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)* (pp. 456-463). IEEE.

Wittgenstein, L. (1997). *Philosophical Investigations* / *Philosophische Untersuchungen*. (G. E. M. Anscombe und R. Rhees eds., G.E.M Anscombe trans.). Blackwell. (Original published 1953)

## СВЕДЕНИЯ ОБ АВТОРЕ / INFORMATION ABOUT THE AUTHOR

Данка Радженович, radjenovic@uni-landau.de

Danka Radjenović, radjenovic@uni-landau.de

Статья поступила 5 июня 2022
одобрена после рецензирования 19 июня 2022
принята к публикации 20 июня 2022

Received: 5 June 2022 /
Revised: 19 June 2022
Accepted: 20 June 2022