

AI: SLY TERMINOLOGY

Mykola Rabchevskiy

Imagine a technology designed to automate actions that a person can perform, the process of using which consists of three steps:

*1. Specialists prepare a set of files called "**training data set**".*

AGI engineering is a reader-supported publication. To receive new posts and support my work, consider becoming a free or paid subscriber.

*2. A computer system developed by specialists in a process called "**training**" or "**machine learning**" uses these prepared files as input, creating output files called "**model**".*

3. The second computer system, having read the model's files, performs proper actions, receiving input data (images, texts) and generating corresponding results (texts, images).

Readers who are familiar with modern achievements in the field of AI will recognize **deep learning** or its analog in the description.

Now imagine another technology that also aims to automate human actions, the process of using which also consists of three steps:

*1. Specialists prepare a set of files called "**source code**."*

*2. A computer system developed by experts in a process called "**compilation**" uses these prepared files as input, producing output files called "**executable**."*

3. The second computer system, having read the executable files, performs proper actions, receiving input data (images, texts) and generating corresponding results (texts, images).

Readers familiar with programming will quickly recognize this process as a standard programming technique.

The similarity between the two described technologies is quite apparent, but the terms used to describe them are distinct. What caused it? The main reason is that the developers of these technologies had different goals, different education, and different experience in the computer field.

The goal of AI developers was to make a system capable of learning and then using the acquired knowledge/skills to perform tasks previously performed by humans. Compiler developers aimed to automate the actions of "ancient" programmers who were forced to

manually translate formulas and calculation schemes into machine code. The implemented principles were similar, but the terminology became completely different. Which is natural in this situation.

This difference in terminology - again in a natural way - makes it challenging to detect existing similarities. And at the same time, it makes it difficult to notice the "gap" in machine learning/deep learning between the declared goal of creating artificial intelligence and what is actually created.

In the case of developing a compiler, it is hardly possible to call the compilation process "**learning**" in one form or another. We are talking about the **transformation of existing knowledge** (algorithm) **from one form of representation** (human-readable text) **into another**, taking into account the characteristics of the computer that will perform the calculation. This transformation is performed according to a **rigidly defined algorithm developed by programmers. Nothing similar to what is usually understood as a learning process can be found here.**

The detailed analysis concludes that **the same is true for ANN-based AI systems**. The method of preparing the data in step 1 is different, but it is also **performed by a human**. Like the compilation algorithm, the **model-building algorithm is developed by a human**. The transformation of input data into output data during operation is carried out according to the **immutable rules**.

Other similarities can also be found: the second step requires relatively high computational power and takes a long time, while the third step is usually quite efficient.

Naturally, the question arises: **should the compiler be considered a kind of artificial intelligence system, or should it be accepted that the process of forming a neural network is not learning in the generally accepted sense of the word, i.e. there is the same "intelligence" level in such a system as in a modern compiler?**

AGI engineering is a reader-supported publication. To receive new posts and support my work, consider becoming a free or paid subscriber.