

Creating Intelligence is a Problem of Philosophy, and Programming, Part 1

By James Lewis

June 2, 2021

Creating human-like intelligence seems possible given the power of computers today, but first we must answer the question: What is intelligence? Twentieth century philosophy provides answers to this question, and to the question of how to create intelligence. The work of the following philosophers is particularly important:

Jean Paul Sartre originated the phrase "existence precedes essence" in his 1945 lecture "Existentialism Is a Humanism." Sartre, borrowing from **Heidegger**, thought that a person starts life without an identity or value and will only acquire these things through living. The same is true for intelligence we create. It cannot begin "life" with an essence, a purpose, a value, whether that value is commercial or otherwise. The software must find its own value.

Ludwig Wittgenstein wrote in his "Tractatus Logico-Philosophicus" that all thought is based on representation. ("4.015 The possibility of all similes, of all the imagery of our language, rests on the logic of representation.") Although he refuted much of the Tractatus later in his career, he remained committed to representation. In "Philosophical Investigations" when discussing how we look inwardly to our thoughts he wrote: "When we look into ourselves as we do philosophy, we often get to see just such a picture. A full-blown pictorial representation of our grammar. Not facts; but as it were illustrated turns of speech." (PI: 295). Intelligent software needs to work with representations of reality: words, sentences, "grammar", and the relationships between these things ("pictures" as Wittgenstein called them.)

I find that while Wittgenstein informs us of the possibility, the **Roland Barthes** gives a programmer the means to create intelligence. In his many books Barthes examines various signatures of intelligence from literature to photography to fashion. Barthes' literary theories provided me with a starting point to develop the database my software uses to represent reality.

Jacques Derrida may be the most problematic twentieth century philosopher. His writing is extremely dense, even opaque. Fortunately, like Derrida I'm a fan of James Joyce, particularly his novel "Ulysses", so I have an in. Derrida was invited to deliver the opening address at the Ninth International James Joyce Symposium in Frankfurt in 1984, a great honor, but one he found challenging as Joyce had boasted about Ulysses, "I've

put in so many enigmas and puzzles that it will keep the professors busy for centuries arguing over what I meant." Joyce, it seems, knew all about Derrida's lecture, and was having a good laugh at Derrida's expense.

But what does Joyce's mockery tell us? Joyce laid traps for the scholars, traps that lock truth-seekers in endless loops; no matter how many times they go around they find themselves no nearer the truth. Derrida realized he must avoid these traps.

So, Derrida focused on coincidences in his own life and coincidences in *Ulysses* and approaches truth by looking for sense in these coincidences, showing us the all too human tendency to find sense in things that likely have no sense. One could say much of what we think we know is imagined.

Creating a robot with a pre-recorded script is rather pointless, like creating an endless loop similar to one of Joyce's traps. When we talk with our robot friend, yes, the robot has a responsibility to make sense of what we say, but we must also make sense of what the robot says, that is the challenge.

In Part 2, I'll discuss the tools programmers have for turning philosophical ideas into machine code.