# A brief visit with the theory of automata

The first concept for an analytical framework for a system of lexicography is that of the automaton that walks the source text. In walking we may see reflections of a tape machine. An equivalence is made between the traversal of a tape of instructions, and the traversal of a sequence of characters. Both affect the performance of the traversal by the automaton, while the character of purpose is very different.

First, we need to interrogate the commonality as more interesting than the difference in either one replicating the other by abstraction. The loss of the distinction of mechanical detail has substance, but is it significant to a theory of automata?

Second, we may postpone that question to examine alternative models of lexicographical processors under the same cloud of thought.

The stateless lexicograph maps each character code, for example ASCII, to a state class. For example, “a” is an "alphabet” and “:” is a “special”. The stateless classifier walks a source text to produce a target lexicograph most efficiently. Likewise, any stateless automaton traverses a process sequence in order.

At this point we may identify the intent of abstraction as the capture of commonality. The abstract automaton is inclusive of the common properties of any discreet, mechanical, stream processor. This is a frame of conception. The frame of analysis decorates the concept with mechanical detail sufficient to represent an object of study.

Therefore, the issues of commonality and comparison at the cloud were faulty by combination. A development proceeded the conception.

A stateful lexicograph may coalesce the represented state of subsequences, and may span the represented state of groups. For example, the subsequence “http://www/” may be coalesced to a “literal” despite having members from distinct classes, “alphabet” and “special”. Similarly, the group “(a < b)” may be spanned despite being a composition.

This automaton differs from an instruction processor in the decoration of its description, but in both cases we are concerned with issues we may term reachability and coverage. What problem spaces are entered and how are they covered? As designers and users we want to open problem spaces to awareness and knowledge, and to produce and use machines that cover those problem spaces uniformly and effectively. That is, we want solutions that close onto corresponding problems crisply.

This is the purpose of the theory of the computing machine, the theory of automata.

John Pritchard

https://twitter.com/syntelos
https://github.com/syntelos

Saturday, 1 September 2018

https://drive.google.com/file/d/1XXqxFSr0DlefZxbf1Pfn7wVsbLYnyTqq/view

# A first step with the theory of automata

With the conception of an analytical framework for the study of automata we need to examine iteration as differentiated from alternative mathematical concepts. We must examine the automaton. First, it must iterate in order to reflect the cycling of the machinery of digital information.

By condensing commonality from the differentiation of hard and soft automata, the automaton inherited iteration as intrinsic and implicit. Iteration is the process by which the object of study, the automaton, performs work. Practical automata convert electrical power to informational results by iteration.

This is important as the central feature of any automaton, and in distinction to the remainder of the universe of mathematical logic. Assuming that the universe of mathematical logic is rational, as would be prescribed by Russell and Whitehead, the universe of computation has an equivalent rationality that is founded on discrete objects of information and steps of iteration. The universe of mathematical logic has emerged from thought and mind as a collection of tools for the management of a particular class of thought. The universe of computational machinery is a tool for the management of information. The distinction concerns media, mind and machine. The mental medium is vastly more powerful than the mechanical medium, as in the comparison of internet and computer. Thought ranges over information with care, or carelessly. One may give little thought to the sights and sounds that bring no new information. Any mechanical proportion would represent a surplus of computational capacity over the examination of every sight and sound. The constraints on the features and devices of automata are severe and unusual, among the objects of the familiar world of information.

Rather than perform in the mind of the experience of performance, with the faculty of training and the facility of expression, the automaton performs in the unfamiliar sterility of its features and devices. The first of these are iteration and object.

With iteration and object we can describe the degenerate automaton that scans an object sequence by iteration, to no effect. With iteration, object, and subject, we can describe the stateless automaton that scans an object sequence to produce a result to a subject.

$$S_i \leftarrow_{\alpha} O_i$$

This automaton may be conceived in software or hardware, it represents any number of extremely simple devices. It performs a simple task at a particular frequency. The typical conception is a function from input to output, from a digital domain to a digital range.

John Pritchard

https://twitter.com/syntelos
https://github.com/syntelos

Monday, 3 September 2018

https://drive.google.com/open?id=13Yssv4DR9N_sB6rCsQm5tkUQXzbwKcYh

# A second step with the theory of automata

The complexity of the stateful automaton

$$S_j \leftarrow^{\sigma}_{\alpha} O_i$$

is bounded by time. The automaton performs as a sequence of events that occur with a constant frequency. An analysis of complexity is founded on the identification and characterization of the event, and is generally inclusive of both concrete and abstract conceptions of the mechanical iteration (step) event.

One kind of automaton may be afforded seconds or minutes, or longer, to complete, while another may complete multiple units of work per second. One may be software, and the other may be hardware etched onto a wafer of silicon, cut into a chip, and wired into a package.

The stateful automaton represents a large class of simple programs and devices that operate over input and output, object and subject, but are not also or additionally interactive.

John Pritchard

https://twitter.com/syntelos
https://github.com/syntelos

Monday, 3 September 2018

https://drive.google.com/open?id=1TTCuTtMONGMeA05SRRXrLD_Udin0S3gO