What's so Funny About
Scripts, Plans, Goals, and Understanding?
Another Look at Schank &Abelson's
Programs and What It Means to Understand

William York

What is understanding? That is, what does it really mean to understand something? Are there different degrees of understanding? What faculties are necessary for genuine understanding to take place, either in a computer or a human brain?

These are some of the questions raised by the work of Roger Schank and Robert Abelson in their book *Scripts, Plans, Goals and Understanding* (1977). In it, the authors present a detailed model of natural language understanding, which they implement using a series of computer programs designed to read and answer questions about short fictional stories, news items, and other brief texts. One of the main issues they confront is how people are able to draw inferences, deal with implicit information, and form connections between events or actions when those connections are not explicitly mentioned. For example, if you are told that John went to a restaurant, ordered chicken, and left a large tip, you could easily answer questions about whether he ate the chicken or whether he felt satisfied with his meal. On the surface, these questions seem trivial, but a brief consideration of the mental mechanisms involved—and of how those mechanisms might be modeled in a computer program—reveals them to be anything but.

For Schank and Abelson, such inference-making abilities are an essential aspect of understanding. Likewise, the ability to answer questions like the ones about John's experience in the restaurant is considered a reliable indicator that understanding has taken place. They make a bold, if somewhat qualified, claim along these lines at one point in their book, announcing, “We have built, at Yale, a computer program called SAM ('Script Applier Mechanism') that understands simple stories about script-based situations" (42). They make similar claims about other programs devised in their lab, including PAM ("Plan Applier Mechanism"), and they illustrate their successes with many detailed examples.

But how valid are these claims? That is, in what sense can programs such as SAM be said to understand what they read? That is a complicated question, which depends in part on what is meant by *understanding*. While not entirely consistent, their usage of *understanding* focuses on the ability to "connect the dots" in a given situation, and a program that is capable of doing so when given a story to read is in turn considered to understand that story.

In evaluating their claims, however, it is important to examine not only the *type* of understanding, but also the *level* or *depth* of that understanding. For example, a student solving a math problem can demonstrate a surface-level understanding by following a set procedure that allows her to reliably "get the right answer," or she can have a deeper understanding by knowing why the answer makes sense in terms of the underlying concepts involved. Schank and Abelson's programs are fairly skilled at answering basic questions about basic stories, so they can reasonably be said to understand these stories on at least a surface-level. However, because they are built on a foundation of preprogrammed concepts and relations between actions—and thus do not actively form their own concepts or interact with the world—it is not plausible that they could have a genuine, deep understanding of what these stories are really about. Another way to put this is that, while the programs are capable of connecting the dots in a given story, they have no way of knowing what the dots are or what they mean in relation to the "real world."

In this paper, I try to argue that although Schank and Abelson have made remarkable achievements within the framework they set for themselves, they deprive themselves of a sturdy framework to begin with by glossing over critical processes such as concept formation, sensory perception, and the role of experience in building "knowledge structures" about the world. Unlike John Searle did in his infamous "Chinese Room" argument, I do not claim there is anything special that distinguishes biological “wetware” from computer software in terms of causal powers or consisting of the “right stuff” (or “wrong stuff”). Rather, the gist of my paper (which is in no way original) can be summarized by modifying one of Schank and Abelson's quotes—which originally states that "[t]here is no way to develop adequate computer 'understanding' without providing the computer with extensive knowledge of the particular world with which it must deal"—to instead read, "There is no way to develop adequate computer 'understanding' *by* providing the computer with extensive knowledge” of that particular world.