Grammars, Programs and the Chinese Room

DRAFT 1.0—6.3.2006; for comments

Cem Bozsahin
Middle East Technical University
bozsahin@metu.edu.tr

Searle (1980) in his Chinese Room thought experiment sets out to show that a purely formalist account of mind is not possible. The particular claim he was arguing against is strong AI, which he identified with the slogan "the computer is not merely a tool in the study of mind; rather the appropriately programmed computer really is a mind in the sense that computers given the right programs can be literally said to *understand* and have cognitive states." This view according to Searle is bound to fail in its aspirations because the kind of computation it envisages is formal, i.e. operates over symbols with no content, whereas mind sets up relations between intentional states and the world via causal powers of the brain.

In the same article (and subsequently in 1990), Searle addresses possible objections to his claim, many of which were companions to his target article, which mainly stem from the assumptions of the experiment and what we understand from understanding. Those that question the experimental setup are concerned with what is embodied in the Chinese room: an extended notion of body that includes not just Searle-in-the-box but the entire room, a robot with perceptual and motor skills, a brain simulator, a parallel rather than serial architecture, or a combination of the above. All these objections are constitutional.

It is interesting that the debate continued between Searle, philosophers, psychologists and practitioners of artificial intelligence, with almost no argument from linguistics (but cf. Carleton 1984). I offer one in this paper from philosophy of linguistics, to question whether Chinese room as imagined by Searle is possible. My argument is not constitutional; it is about what Searle considers computational, and about linguistic conception of the same notion.

First a summary of Searle's argument, from a more recent self statement (Searle, 2001): Imagine a native speaker of English, who has no knowledge of Chinese, locked in a room full of boxes of Chinese symbols (a database) together with a book of instructions in English (the program), which (s)he can interpret, for manipulating the symbols. More Chinese symbols are sent in to the room (questions), which the person in the room correctly answers in Chinese symbols by following the instructions for matching the database symbols and symbols in questions. The person passes the Turing (1950) test in understanding Chinese, for the outsiders cannot peek inside the room to see that the answers are simply look-ups. Yet (s)he does not understand a word of Chinese. The program and the database add no understanding of Chinese to the person, though (s)he already knows how to interpret symbols in one language, namely English. By extension, computers cannot understand Chinese (or any human language) by purely formal manipulation of symbols.

The linguistic aspect of the experiment I think is as follows: What is Chinese in the Chinese room is the database, and fragments of the program that contains Chinese symbols and their abstractions (the program is in English, but it is about Chinese symbol correspondences). The program cannot be of infinite size (otherwise it wouldn't be a program), therefore the correspondences in the program cannot be phrase-to-phrase matching, for we know that there are (countably) infinitely many Chinese expressions. Hence the program must contain finitely characterizable symbols and their program-internal abstractions, such as calling a group of symbols a certain kind of variable, and certain combinations of variables to be other variables and so on, in other words, a generative grammar of Chinese.

In the thought experiment we *must* assume that the program contains a generative grammar because we can "suppose also that the programmers get so good at writing the programs that from the external point of view—that is, from the point of view of somebody outside the room in which I am locked—my answers to the questions are indistinguishable from those of *native* Chinese speakers." (Searle, 1980, my emphasis).

Let us now turn to the boxes of Chinese symbols. They would minimally contain Chinese vocabulary, and perhaps more, such as a large inventory of expressions based on symbols in the program. This too must be finite to fit into the room. We thus have a generative linguistic system of grammar and a lexicon housed in the room.

I claim that the experimental setup is inconsistent because of the forced assumption of having a generative grammar, and not being able to use it for semantic interpretation in the room; all generative grammars—in any generative linguistic theory—are interpretable because their product, a structural description (Chomsky, 1965, 1995), is there solely to provide full array of phonetic, semantic and syntactic interpretation.

What, then, is the problem with computation in Searle's program? In linguistic sense, the program is not doing computation at all, because computation is what links the expression (the phonological form) and the meaning (the logical form) at the interfaces to perceptual and conceptual systems of cognition. It is not computation in computing science sense either, for computing there too is to link programs (the form) with executable code (the meaning), at the interfaces of the machine to programmer's expressions and intended tasks, the latter of which cannot be determined by the computational system.

An uninterpretable program has no semantics, whereas a program that *does* nothing has one, with perhaps free interpretation in the programmer's world. Thus Searle's criticism of formal symbol manipulation as basis of understanding may be directed towards possible reductionism (of e.g. ignoring semantics) in current practice, but it is not an intrinsic problem of computation.

One might argue that semantics as conceived above is not really semantics because it is not situated in the external world, but this is precisely the point in linguistics and computing: language-internal semantics is only a gateway to the conceptual system, then to the world, where meaning cannot be determined by language. Language provides a semantic representation over which external (anchored) meanings can be enumerated. That is, understanding is an interface problem of connecting internal and external meanings. If this is the case, then a computational system can in principle be made to face the same conditions as the child for understanding the connections between sounds and meanings.

There is already some progress in the way of breaking the "semantic divide" of a child's acquisition of language and a machine's learning of human language. Zettlemoyer and Collins (2005) report an experiment in statistical learning of generative grammars, in which the training data (for the machine) are sound-meaning pairs, and in which syntax is a hidden variable, that is, there is no external access to the internal states of a program such as Searle's. Therefore, the input to the room must be sound-meaning pairs in order for computation to take place inside the room.

Led this way, the system learns a fully interpretable grammar, of course with errors and approximations, but with the possibility of correcting them by exposure to further data.

The results are too preliminary to be conclusive, but they point out principled directions for discerning methodological and intrinsic problems of computing. I conclude that (i) Searle's experimental setup is linguistically inadequate, and (ii) it can be made consistent with bona fide computation, in which case the unduly pessimistic belief that a computational system cannot be made to face the same conditions for understanding as humans is not warranted.

References

Carleton, L. R. (1984). Programs, language understanding, and Searle. *Synthese 59*(2), 219–230.

Chomsky, N. (1965). *Aspects of the Theory of Syntax*. Cambridge, MA: MIT Press.

Chomsky, N. (1995). *The Minimalist Program*. Cambridge, Mass.: MIT Press.

Searle, J. R. (1980). Minds, brains and programs. *The Behavioral and Brain Sciences 3*, 417–424.

Searle, J. R. (1990). Is the brain's mind a computer program? *Scientific American 262*(1), 26–31.

Searle, J. R. (2001). Chinese Room argument. In R. A. Wilson and F. C. Keil (Eds.), *The MIT Encyclopedia of the Cognitive Sciences*, pp. 115–116. MIT Press.

Turing, A. M. (1950). Computing machinery and intelligence. *Mind 59*(236), 433–460.

Zettlemoyer, L. S. and M. Collins (2005). Learning to map sentences to logical form: Structured classification with Probabilistic Categorial Grammars. In *Proc. of the 21st Conf. on Uncertainty in Artificial Intelligence*, Edinburgh.