# Meeting 5: Meaning, Poetry, and Word Vectors

CS198: The Poetry of Computer Science, the Computer Science of Poetry
Philosophy of Computation at Berkeley
pocab.org

October 30, 2017

## 1 Syntax and Semantics

In semiotics[1], semantics is roughly defined as the relationship of signs to what they signify; syntax is roughly defined as the formal or structural relations between signs. Later in his years, Searle rephrased his Chinese Room argument using such constructs.

- Mental contents have semantics. (Thought has meaning.)
- Computers perform purely syntactic operations. (All that computers do is mindlessly shift symbols around according to some arbitrary rules.)
- Semantics is not reducible to syntax. (You cannot deduce the meaning of words just by looking at how the words are put next to each other.)

∴ Computers cannot have mental contents. (Computers can't think.)

Each of Searle's three points are plausible, and if they're all true, the conclusion is inevitable. But of course we all want to disagree with Searle, and so today we'll go after the third, somewhat obscure-sounding point. Is semantics really not reducible to syntax? As in, is there something about syntax that lets us deduce at least some, if not all, of semantics?

The tension between syntax and semantics is a proxy war for a discussion we've had over and over this semester: the tension between the objective and the subjective. Roughly, syntax, as formal rules and structural relations, is objective, and semantics, as the meaning of symbols, is subjective. If one believes there is a clear-cut distinction between the subjective and the objective, a so-called "metaphysical gap" between them, then Searle's argument is irrefutable.

At least one easy objection can be raised: a sign, *by itself*, has meaning, *without* what it signifies. Newspaper articles are like liquids: you can put one in any form and it retain its meaning. Poems are like solids: if you mess with the form of a poem, you mess with its meaning. One may even say that the meaning of poetry resides not in the signs, not in the signified, but between the signs and the signified. Hugo has a few things to say about this idea.

## 2 The Triggering Town

"Generally, in English, multisyllabic words have a way of softening the impact of language. With multisyllabic words we can show compassion, tenderness, and tranquility. With multisyllabic words we become more civilized. In the first four lines of the poem, seven of the twenty-six words, slightly better than one out of four, are two syllable words. This is a fairly high count unless you are in politics. The snake is sleepy. He presents no threat to the speaker. His dwelling is that of a harmless creature, a gopher. It's almost as if the snake were a derelict, an orphan, a vagabond who sleeps wherever he can."

[1] The study of meaning-making, the study of signs, the study of communication

- Project idea: collect a set of transcripts from Donald Trump and Hilary Clinton. Count the mean number of syllables in each.
- Project idea: find a correlation between the number of syllables in a word, and the meaning of the word represented a a vector.[2]
- Consider these arguments:

  (1) Given a poem, we can deduce some meaning in it, without ever reading it, simply by counting the mean number of syllables in the poem.

  (2) Given a piece of code, we can deduce some meaning in it, without ever reading it, simply by counting the mean number of syllables in the code.

  Q. Why is (1) more plausible than (2)?

"In the news article the relation of the words to the subject (triggering subject since there is no other unless you can provide it) is a strong one. The relation of the words to the writer is so weak that for our purposes it isn't worth consideration. Since the majority of your reading has been newspapers, you are used to seeing language function this way. When you write a poem these relations must reverse themselves. That is, the relation of the words to the subject must weaken and the relation of the words to the writer (you) must take on strength."

"Never worry about the reader, what the reader can understand. When you are writing, glance over your shoulder, and you'll find there is no reader. Just you and the page. Feel lonely? Good. Assuming you can write clear English sentences, give up all worry about communication. If you want to communicate, use the telephone."

- What does Hugo mean by "Never worry about the reader"? Does he literally mean that the reader doesn't matter? In a related note, what does Hugo mean by "communication"?

"Assumptions lie behind the work of all writers. The writer is unaware of most of them, and many of them are weird. Often the weirder the better. Words love the ridiculous areas of our minds. But silly or solid, assumptions are necessary elements in a successful base of writing operations. It is important that a poet not question his or her assumptions, at least not in the middle of composition. Finish the poem first, then worry, if you have to, about being right or sane."

- Consider the following argument:

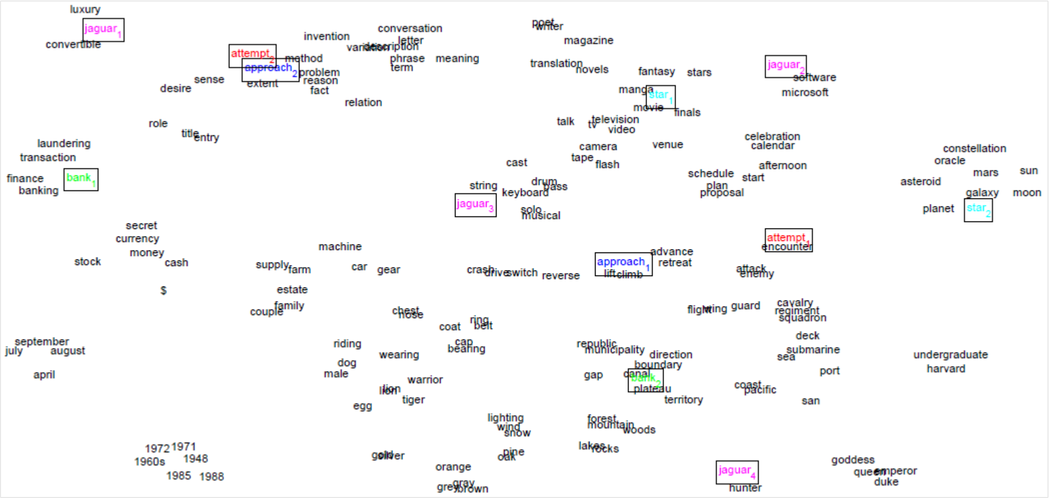
  > According to the Church-Turing thesis, everything that is physically computable is computable by a Turing machine. The process of writing poetry is a physical computation performed by the brain. Therefore, if one assumes the Church-Turing thesis, one assumes there is a Turing machine for writing poetry.

- Does this assumption take out the "magic" of poetry? Why or why not? Consider the sentence, "there is a Turing machine". What does the word "is" mean in this sentence? Where, exactly, *is* this Turing machine? If we can't point to it, and say, "it is here", what do we mean by *is*?
- Consider the following argument:

  > Assumptions are like axioms. Gödel's Incompleteness Theorem tells us that a formal system cannot prove if its axioms are true or false. In the same way, one must not question one's assumptions while writing poetry. The process of generating poems based on assumptions is the same as the process of generating mathematical theorems based on axioms.

[2] We'll get to vector representations of words later in today's discussion.

# 3 Word Vectors



Word vectors are the bread and butter of modern natural langauge processing algorithms. If you've used Siri or Cortana or whatever, you've dealth with word vectors. A word vector is a mathematical vector, the one you learn in linear algebra, representing the "meaning" of a word. A vector is a force, a push; one can imagine each word being a "push" in some semantic direction. Words that have similar meaning "push" in similar directions with similar powers. Words that have no meaningful relation whatsoever push in orthogonal directions.

But how are these meanings learned? Word vectors rest on a philosophical dictum made by the late linguist John Firth: "You shall know a word by the company it keeps". The basic idea is to take a giant corpus[3] and count how many times some word occurs in proximity to all other words. You can imagine a square table with the rows and columns being all the unique words in the corpus, and each entry in the table filled in with the number of times *word in that row* occurs in close proximity to *word in that column*. After you're done, each column will have a word at the top and a bunch of numbers below. Take those numbers and use it as the vector of that word.[4]

There's an alternative formulation, which in practice is almost the same: dump the corpus into a neural net by giving it a "window" of words, such as "the dog is $x$ at the moon", and training it to predict $x$. Take the weights of the neural net as the word vectors. For this reason, word vectors are sometimes called neural embeddings of words, or word embeddings.

Word vectors can encode a suprisingly large amount of semantic meaning. Given a set of word vectors, you can ask what the closest word(s) to some word is. You can also ask it to do analogy tasks, such as,

$$queen : king = x : man^5$$

Last I checked, word vectors can solve SAT analogy tasks with over 70% accuracy.

Multimodal word vectors tackle another philosophical problem: the "symbol-grounding" problem. This question asks how meaning can be learned by purely looking at patterns of syntax, without being "grounded" in the real world; consider a baby growing up in a dark, gray, dusty library, never going outside, reading tens of thousands of books for about twenty years. Would this baby really know what "Sun" means, without having ever seen a sun? Multimodal word vectors seek to solve this problem by appending to word vectors sensory data, so that a vector for "apple", for example, has a component that corresponds to what an apple looks like, a component that corresponds to what an apple smells like, a component that corresponds to what an apple sounds like, etc.

[3] A fancy term for "a bunch of text"

[4] Actually, we've skipped one step. A big corpus will usually have tens of thousands of unique words. So the table will be huge, tens of thousands of entries across and down. It will be too big to deal with. So we make the table smaller using a sort of compression technique, which people call singular value decomposition. Then the table can be manageable, maybe a hundred entries across and down. *Then* we can take those hundred numbers in a column as the vector for a word.

[5] Yes, word vectors can be sexist: this is a growing problem, exposed in "Man is to Computer Programmer as Woman is to Homemaker?": https://arxiv.org/pdf/1607.06520.pdf