The Mathematics of Text Structure

Bob Coecke

Oxford University, Department of Computer Science

Cambridge Quantum Computing Ltd.

Abstract

In previous work we gave a mathematical foundation, referred to as DisCoCat, for how words interact in a sentence in order to produce the meaning of that sentence. To do so, we exploited the perfect structural match of grammar and categories of meaning spaces.

Here, we give a mathematical foundation, referred to as DisCoCirc, for how sentences interact in texts in order to produce the meaning of that text. We revisit DisCoCat: while in the latter all meanings are states (i.e. have no input), in DisCoCirc word meanings are types of which the state can evolve, and sentences are gates within a circuit which update the meaning of words. Like in DisCoCat, word meanings can live in a variety of spaces e.g. propositional, vectorial, or cognitive. The compositional structure are string diagrams representing information flows, and an entire text yields a single string diagram in which word meanings lift to the meaning of an entire text.

While the developments in this paper are independent of a physical embodiment (cf. classical vs. quantum computing), both the compositional formalism and suggested meaning model are highly quantum-inspired, and implementation on a quantum computer would come with a range of benefits.

From States to Processes

Consider the following example:

```
Alice is a dog.
Bob is a person.
Alice bites Bob.
```

Clearly, the meaning of the third sentence crucially depends on what we learn about the meaning of the nouns `Alice` and `Bob` in the first two sentences, turning `dog bites man` into `man bites dog` if Bob were to be a `dog` and `Alice` were to be a `person`. Also, before the 1st sentence is stated, `Alice` is just a meaningless name, and the same goes for `Bob` until the 2nd sentence is stated. So the meaning of `Alice` and `Bob` evolves as the text progresses, and it are the sentences that update our knowledge about `Alice` and `Bob`. What we propose is that the 3rd sentence, which would look like:
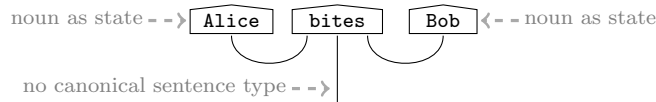

in DisCoCat, would instead be drawn like this:
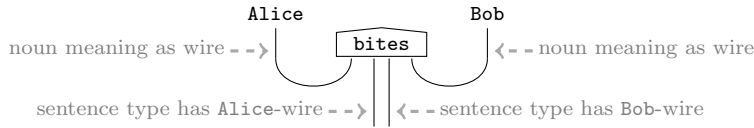
So in particular, the nouns `Alice` and `Bob` are now not states but wires (a.k.a types) and the sentence is an I/O-box with the nouns `Alice` and `Bob` both as inputs and as outputs. In this way, the sentence can act on the nouns and update their meanings. Hence:
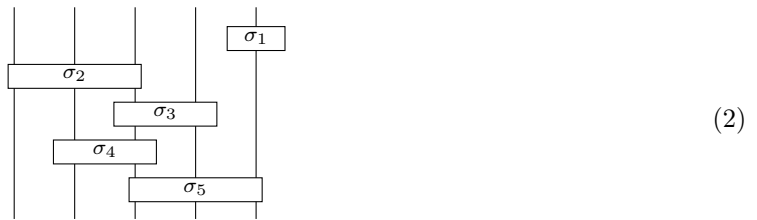
> *A sentence is not a state, but a process,*

that represents how words in it are updated by that sentence. The wire-representation of `Alice is a dog` becomes:
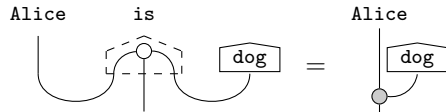
Altogether the composition of the three sentences now becomes:
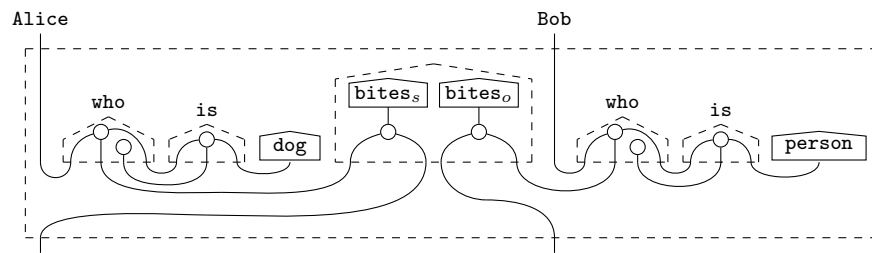

(1)
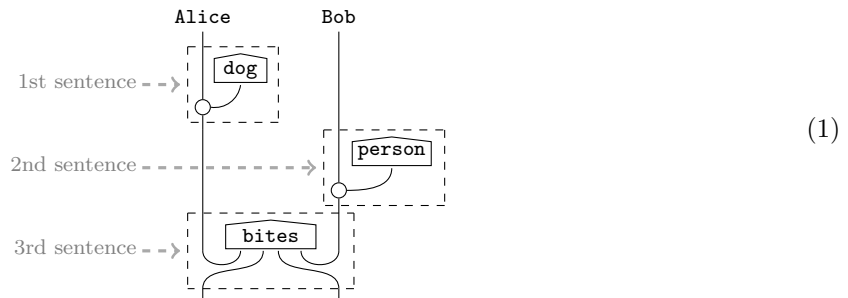
So in general, given text, we end up with a wire diagram that looks like this:

$$(2)$$

where the sentences themselves also have a wire diagram. In particular, it's a process, and this process alters our understanding of words in the text. This yields another slogan:

Text is a process that alters our understanding of words

The single sentence:

results in the wire-diagram (1), so diagrams can expose that different texts have the same meaning.

Another example

Harmonica (is brother of) Claudio.
Frank hangs Claudio.
Snaky (is in gang of) Frank.
Harmonica shoots Snaky.
Harmonica shoots Frank.

$\mapsto$