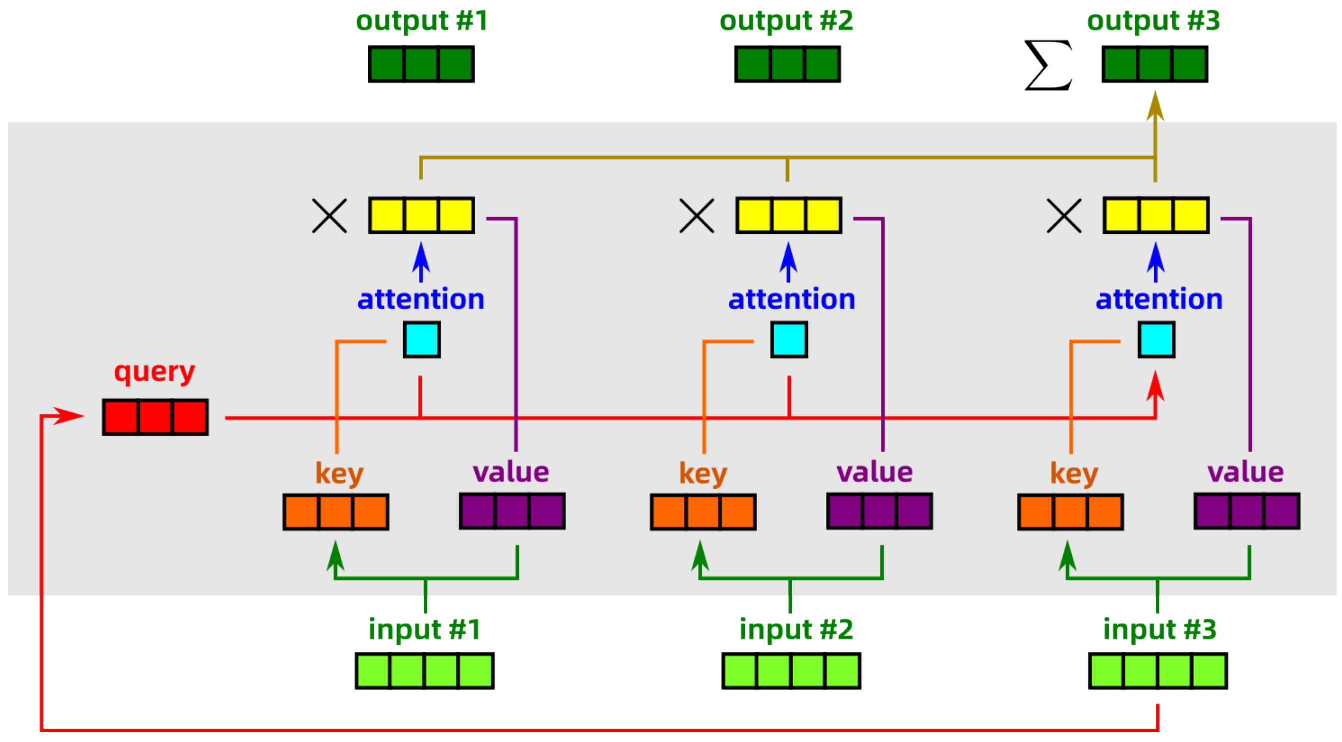
Transformer has logic structure

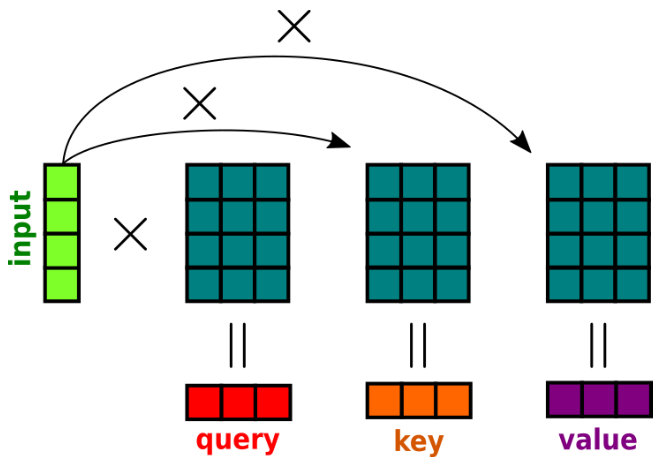
In this infographic I'd explain a major finding that is the culmination of many years of my research: the Transformer is a symbolic-logic machine.

For your convenience let's refresh on the Transformer's Self-Attention mechanism:



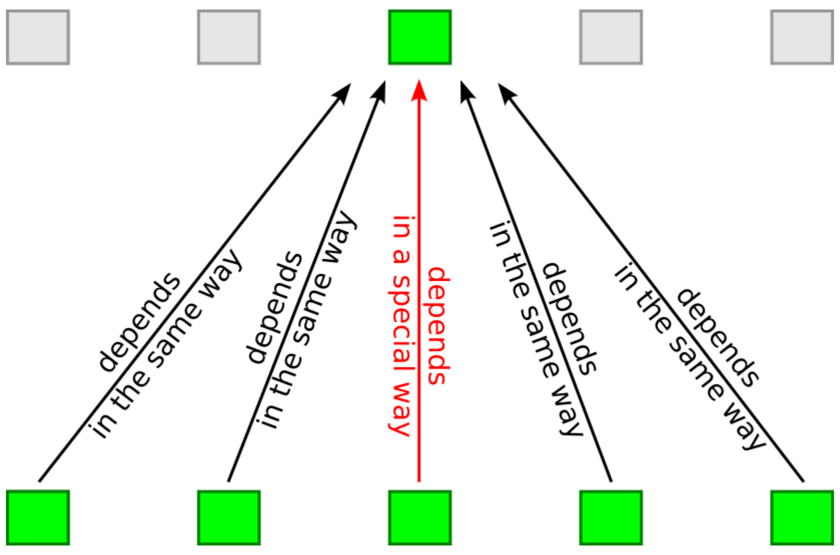
(1)

"Input" tokens are translated to Q, K, V (query, key, value)'s via matrix multiplication, which can be regarded as a kind of table look-up, or memory store:



(2)

From an abstract point of view, the Transformer has the following structure, which gives rise to its equivariance property (if input elements are swapped in a certain order, the output elements changes the same way):



(3)

The equivariance property corresponds to the exchangeability of logic propositions:

$$A \wedge B \quad \Leftrightarrow \quad B \wedge A \tag{4}$$

For example:

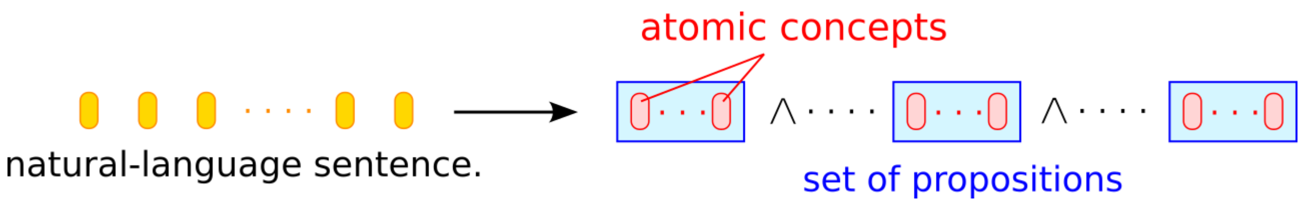
$$\text{it's raining} \wedge \text{I'm heart-broken} \quad \Leftrightarrow \quad \text{I'm heart-broken} \wedge \text{it's raining} \tag{5}$$

Propositions are made up of atomic concepts, but here, at the sub-propositional level, atoms cannot be permuted freely:

$$\text{I} \cdot \text{love} \cdot \text{you} \quad \neq \quad \text{you} \cdot \text{love} \cdot \text{me} \tag{6}$$

otherwise there would be no such things as heart-breaks.

Given a natural-language sentence, we'd like to convert or decompose it into a bunch of logic propositions:

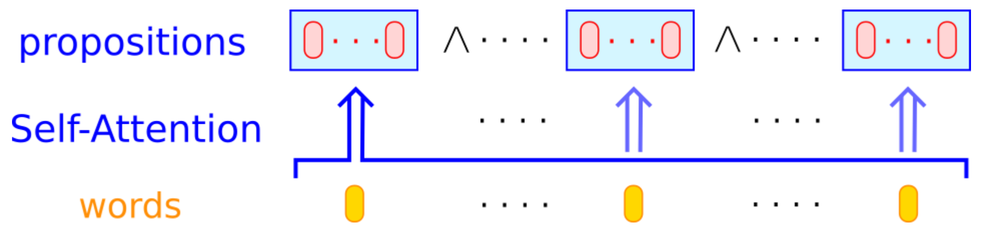


(7)

The structure on the right is a mental state of a logical AI system. It is composed of (exchangeable) propositions, which are in turn made up of atomic concepts. This 2-level structure is characteristic of all logical systems.

Surprisingly, I found out that the Transformer completely satisfies this 2-level logic structure.

On the first layer, a Transformer transforms each input word token into one proposition:



(8)

The crucial point here is that propositions are made up of atoms (⬭), which is achieved in the Transformer by adding vectors (that represent atomic concepts) together. Note also that the Transformer is equivariant, so we must add "positional encoding" to each word.