Understanding BERT Transformer: Attention isn't all you need

A parsing/composition framework for understanding Transformers

Synapse

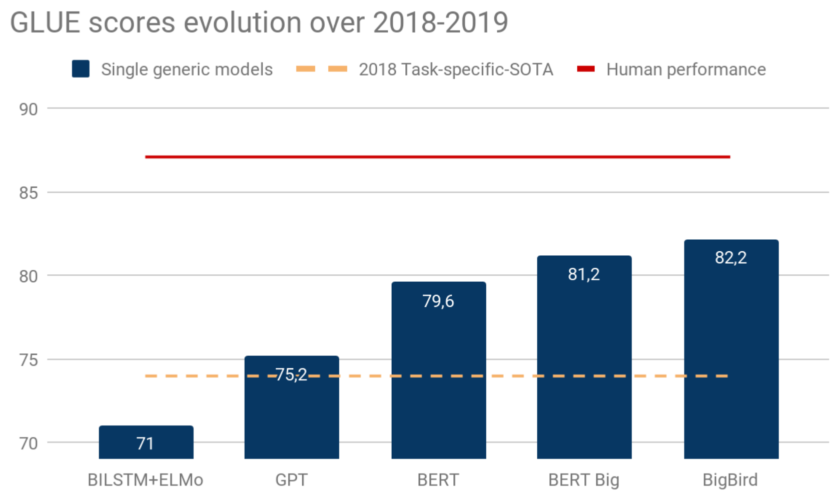
Damien Sileo Follow

Feb 26 · 9 min read ★

Why BERT matters

BERT is a recent natural language processing model that has shown groundbreaking results in many tasks such as question answering, natural language inference and paraphrase detection. Since it is openly available, it has become popular in the research community.

The following graph shows the evolution of scores for GLUE benchmark—the average of scores in various NLP evaluation tasks.



While it's not clear that all GLUE tasks are very meaningful, generic models based on an encoder named *Transformer* (Open-GPT, BERT and BigBird), closed the gap between task-dedicated models and human performance and within less than a year.

However, as Yoav Goldberg notes it, we don't fully undestand how the Transformer encodes sentences.

> *[Transformers] in contrast to RNNs—relies purely on attention mechanisms, and does not have an explicit notion of word order beyond marking each word with its absolute-position embedding. This reliance on attention may lead one to expect decreased performance on syntax-sensitive tasks compared to RNN (LSTM) models that do model word order directly, and explicitly track states across the sentence.*

Several articles delve into the technicalities of BERT. Here, we will try to deliver some new insights and hypotheses that could explain BERT's strong capabilities.

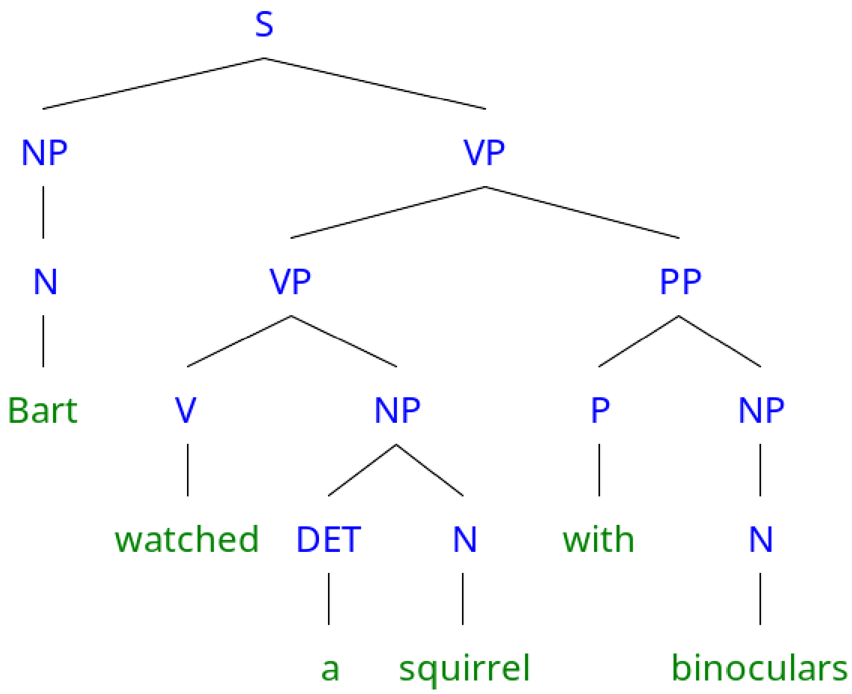
A framework for language understanding: parsing/composition

The way humans are able to understand language has been a long-standing philosophical question. In the 20th century, two complementary principles shed light on this problem:

- The Compositionality principle states that the meaning of word compounds is derived from the meaning of the individual words, and the manner in which those words are combined. According to this principle, the meaning of the noun phrase *"carnivorous* *plants*", can be derived from the meaning of "*carnivorous*" and the meaning of "*plant*" through a process named composition. [Szabó 2017]

- The other principle is the hierarchical structure of language. It states that through analysis, sentences can be broken down into simple structures such as clauses. Clauses can be broken down into verb phrases and noun phrases and so on.

Parsing hierarchical structures and deriving meaning from their components recursively until sentence level is reached is an appealing recipe for language understanding. Consider the sentence "*Bart watched a squirrel with binoculars*". A good parsing component could yield the following parse tree:

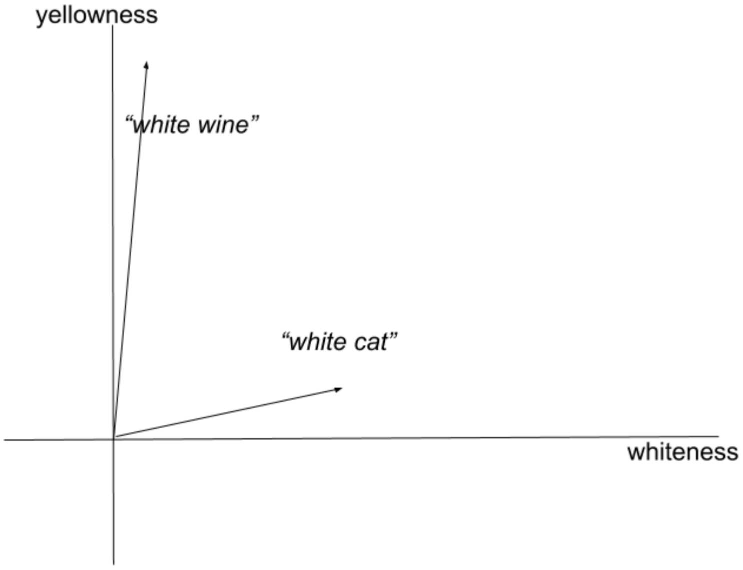


A constituency-based parse tree of the sentence "Bart watched a squirrel with binoculars"

The meaning of the sentence could be derived from successive compositions (composing "*a*" and "*squirrel*", "*watched*" with "*a squirrel*", "*watched a squirrel*" and "*with binoculars*") until the sentence meaning is obtained.
Vector spaces (as in word embeddings) can be used to represent words, phrases, and other constituents. Composition could be framed as a function f which would compose ("*a*","*squirrel*") into a meaningful vector representation of "*a squirrel*" = f("*a*","*squirrel*"). [Baroni 2014]

However, composition and parsing are both hard tasks, and they need one another.

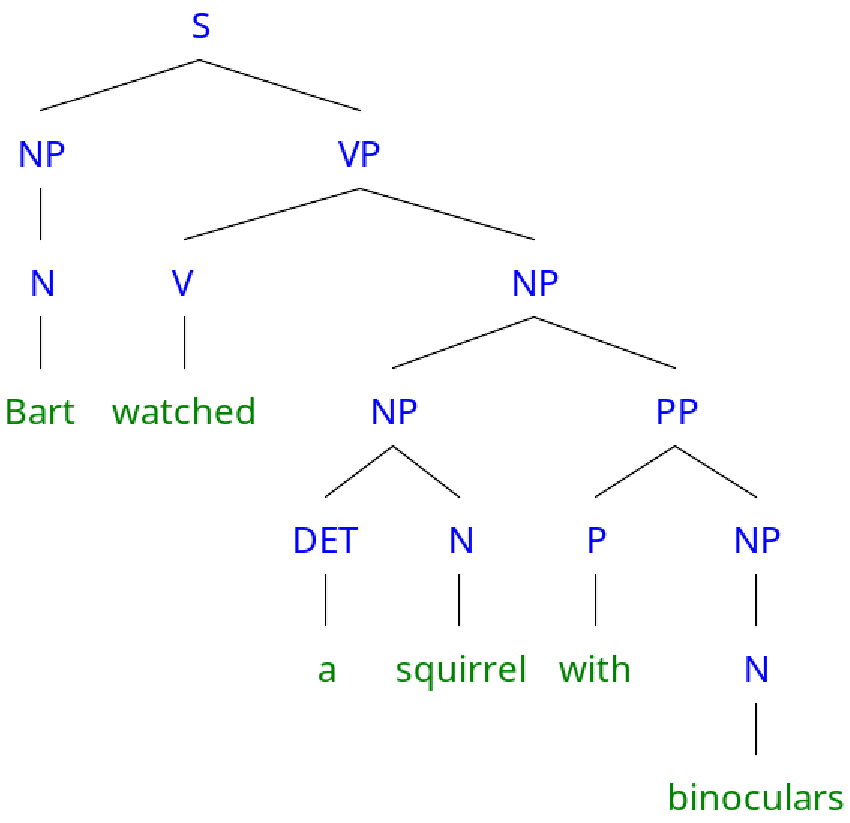
Obviously, composition relies on the result of parsing to determine what ought to be composed. But even with the right inputs, composition is a difficult problem. For example, the meaning of adjectives changes depending on the word they characterize: the color of *"white wine"* is actually yellow-ish while a white cat is actually rather white. This phenomenon is known as co-composition. [Pustejovsky 2017]



Representations of "white wine" and "white cat" in a two-dimensional semantic space (with color dimensions)

A broader context can also be necessary for composition. For instance, the way the words in *"green light"* should be composed depends on the situation. A green light can denote an authorization or an actual green light. The meaning of some idiomatic expressions requires a form of memorization rather than composition per se. Thus, performing those compositions in the vector space requires powerful nonlinear functions like a deep neural network (that can also memorize [Arpit 2017]).

Conversely, the parsing operation arguably needs composition in order to work in some cases. Consider the following parse tree of the same previous sentence "*Bart watched a squirrel with binoculars*".



Another constituency-based parse tree of the sentence "Bart watched a squirrel with binoculars"

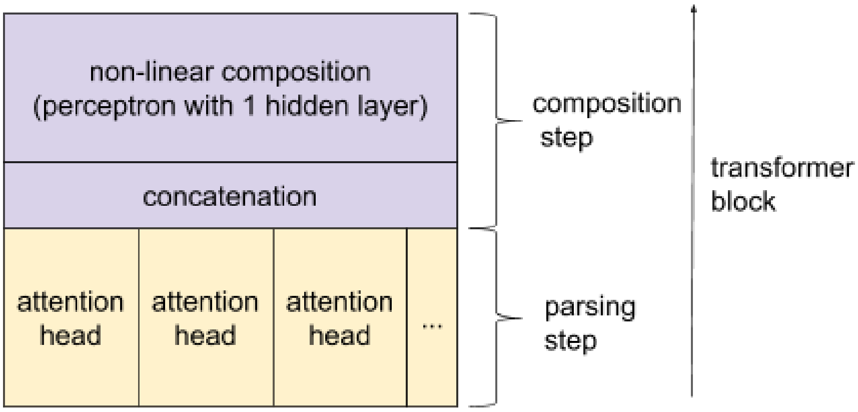
While it is syntactically valid, this parse leads to an odd interpretation of the sentence where Bart watches (with his bare eyes) a squirrel holding binoculars. However, some form of composition must be used in order to figure out that a squirrel holding binoculars is an unlikely event.
More generally, many disambiguations and integrations of background knowledge have to go on before the appropriate structures are derived. But this derivation might also be achieved with some forms of parsing and composition.

Several models have tried to put the combination of parsing and composition in practice [Socher 2013], however they relied on a restrictive setup with manually annotated standard parse trees, and have been outperformed by much simpler models.

How BERT implements parsing/composition

We hypothesize that Transformers rely heavily on these two operations, in an innovative way: since composition needs parsing, and parsing needs composition, Transformers use an iterative process, with successive parsing and composition steps , in order to solve the

interdependence problem. Indeed, Transformers are made of several stacked layers (also called blocks). Each block consists of an attention layer followed by a non-linear function applied at each token.
We will try to highlight the link between those components and the parsing/composition framework.

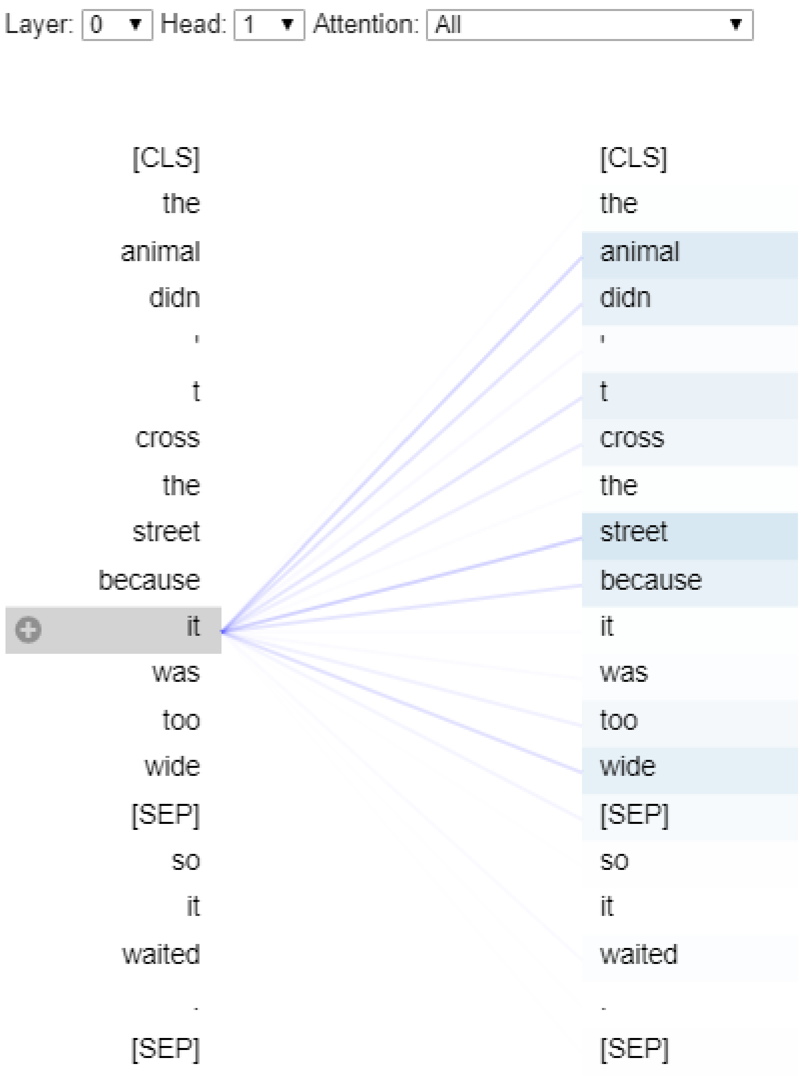


A transformer block, seen as successive parsing and composition steps

Attention as a parsing step

In BERT, an attention mechanism lets each token from the input sequence (e.g. sentences made of word or subwords tokens) focus on any other token.

For illustration purposes, we use the visualization tool from this article to delve into the attention heads and test our hypothesis on the pre-trained BERT base uncased model. In the following illustration of an attention head, the word *"it"* attends to every other token and seems to focus on *"street"* and *"animal"*.



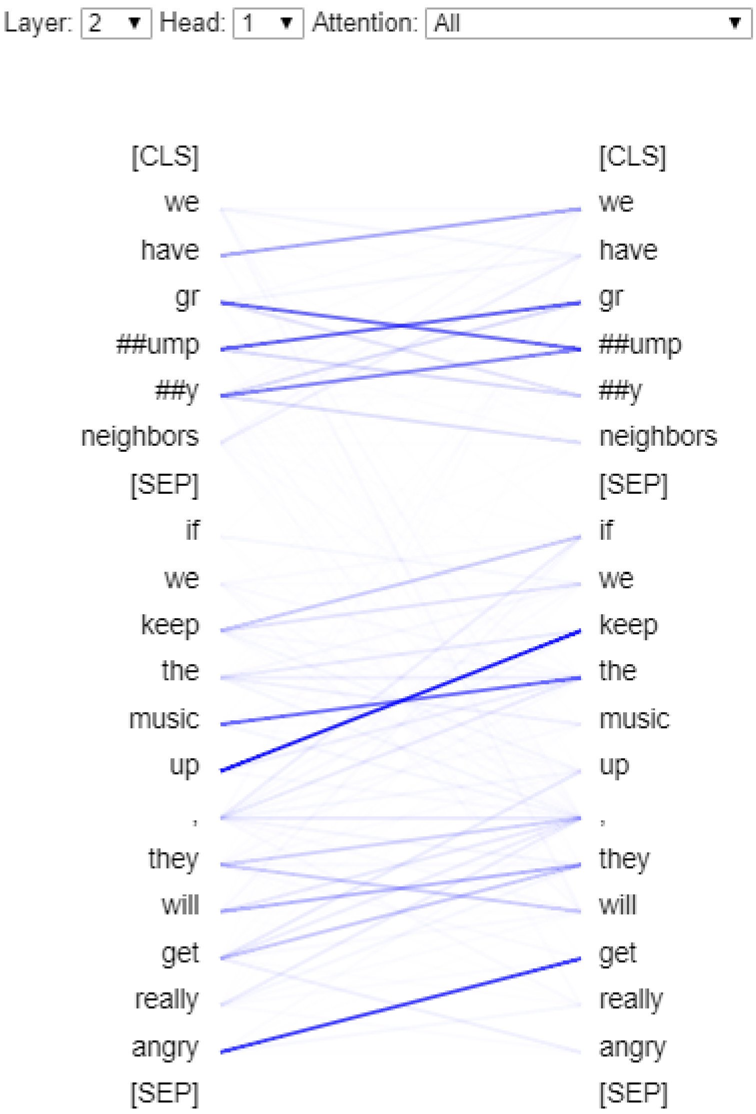
Layer: 0 Head: 1 Attention: All

Visualization of attention values on layer 0 head #1, for the token "it".

BERT uses 12 separate attention mechanism for each layer. Therefore, at each layer, each token can focus on 12 distinct aspects of other tokens. Since Transformers use many distinct attention heads (12*12=144 for the base BERT model), each head can focus on a different kind of constituent combinations.

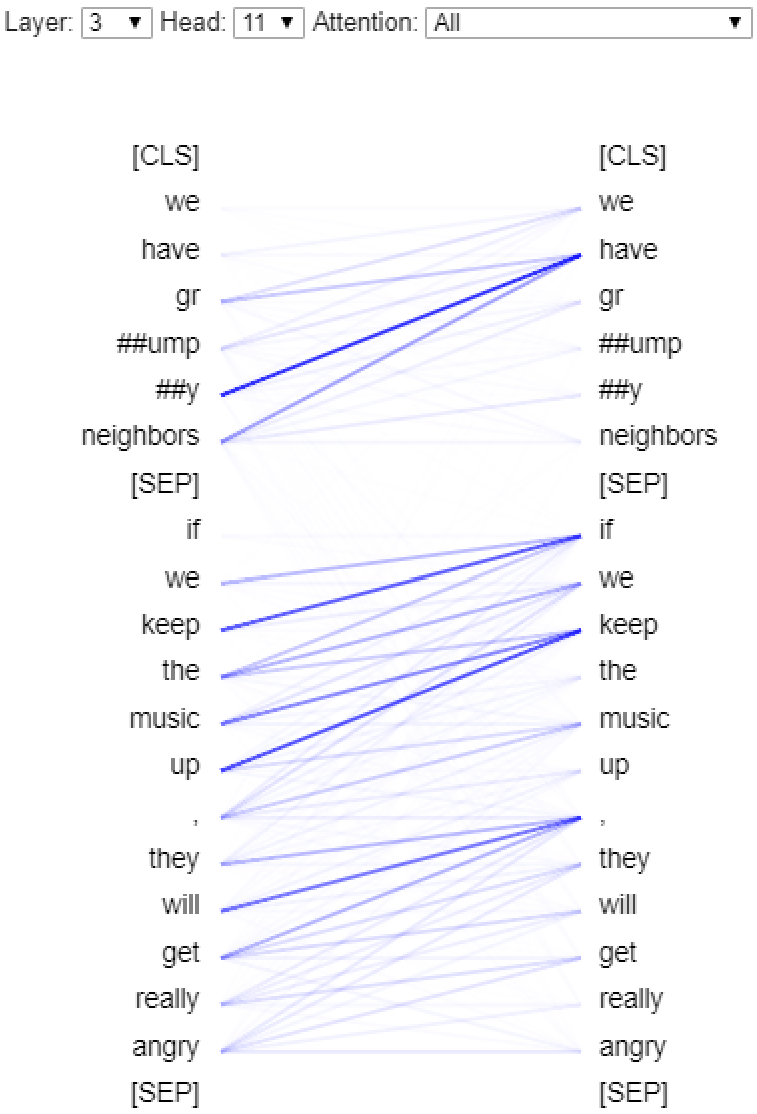
We ignored the values of attention related to the *"[CLS]"* and *"[SEP]"* token. We tried using several sentences, and it's hard not to overinterpret results, so you can feel free to test our hypothesis on this colab notebook with different sentences. Please note that in the figures, the left sequence attends to the right sequence.

In the second layer, attention head #1 seems to form constituents based on relatedness.



Visualization of attention values on layer 2 head #1; which seems to pair related tokens

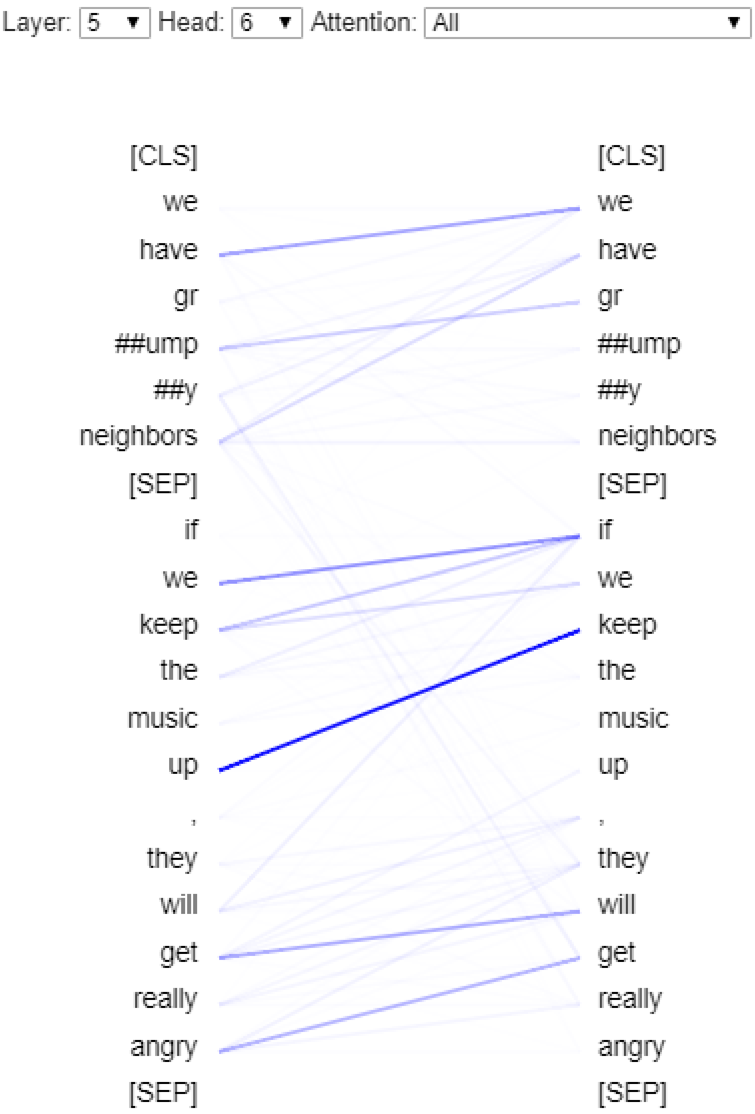
More interestingly, in the third layer, head #9 seem to show higher level constituents : some tokens attend to the same central words (if, keep, have)



Layer: 3 ▼ Head: 11 ▼ Attention: All ▼

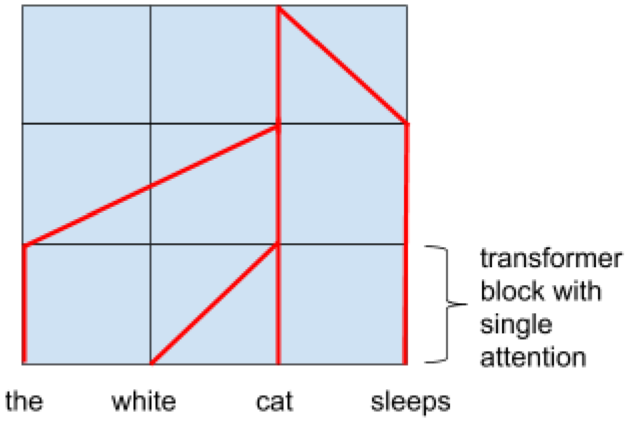
Visualization of attention values on layer 3 head #11; where some tokens seem to attend to specific central words (e.g. have, keep)

In the fifth layer, the matching performed by the attention head #6 seem to focus on specific combinations, notably involving verbs. The special tokens like *[SEP]* seem to be used to indicate the absence of matching. This could allow attention heads to detect specific structures where a composition would be appropriate. Such consistent structures could be fed to the composition function.



Visualization of attention values on layer 5 head #6; where combinations seem to be more focused (we, have), (if, we), (keep, up) (get, angry)

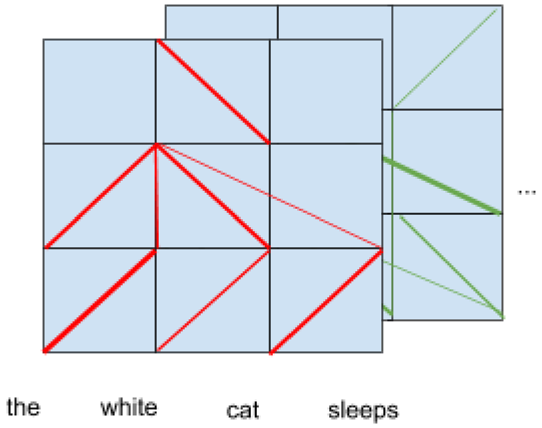
Arbitrary trees could have been represented with successive layers of shallow parsing, as shown in the next figure:



How several layers of attention can represent tree structures

We didn't find such clear cut tree structures upon examination of BERT attention heads, but still, it is possible for Transformers to represent them.
We note that, since the encoding is performed simultaneously on all layers, it is hard to correctly interpret what BERT is doing. The analysis of a given layer only makes sense with respect to the next and previous layers. The parsing is also distributed across attention heads.

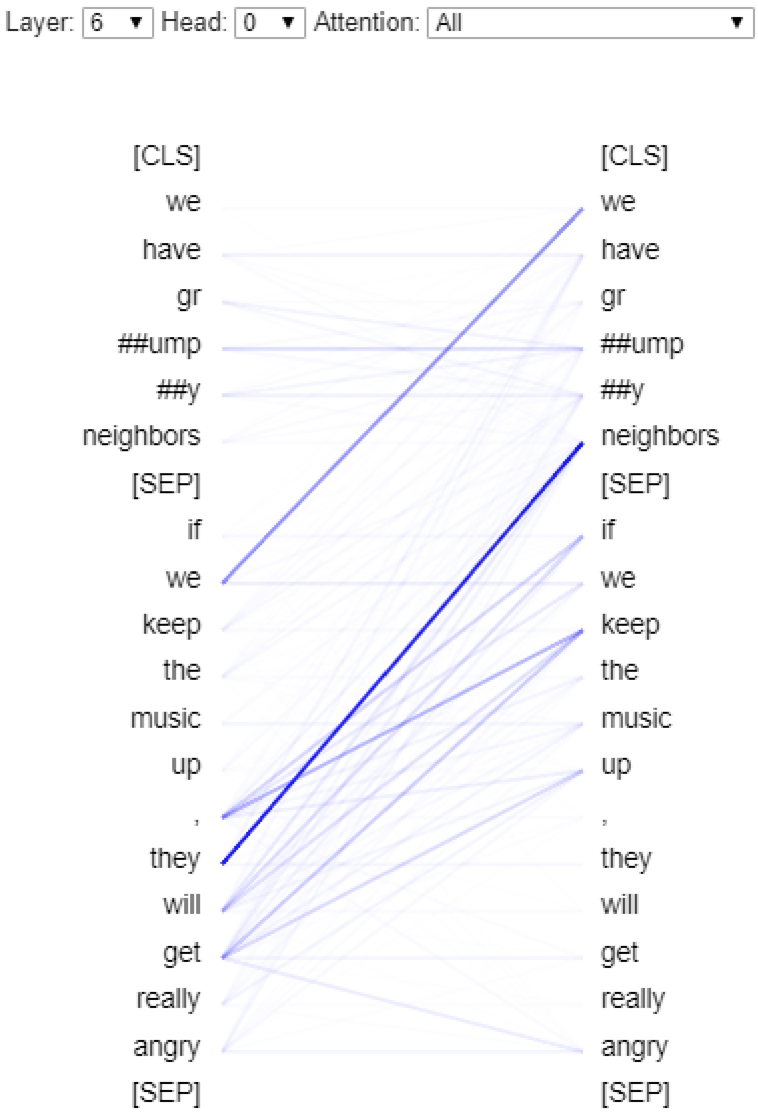
The following illustration shows what BERT attention more realistically looks like for two attention heads:



A more realistic look of attention values in BERT

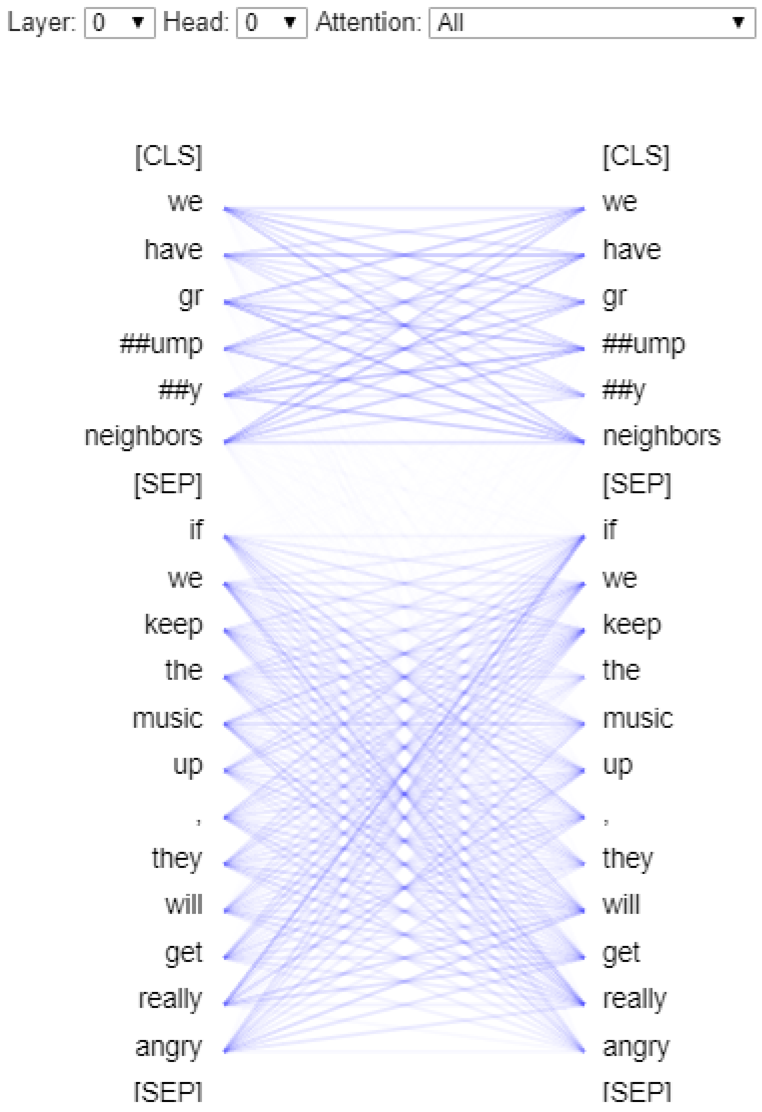
However, as we have seen previously, the parse tree is a high level representation, and it might build upon more complex "rhizomatic" [Deleuze 1987] structures . For instance, we might need to find out what pronouns refer to in order to encode the input (coreference resolution). In other cases, a global context can also be required for disambiguation.

Surprisingly, we discovered an attention head (layer 6 head #0) that seems to actually perform coreference resolution. And, as noted here, some attention heads seem to feed a global context to each word (layer 0 head #0).



Layer: 6 ▼ Head: 0 ▼ Attention: All ▼

Coreference resolution occurring in head #0 of layer 6



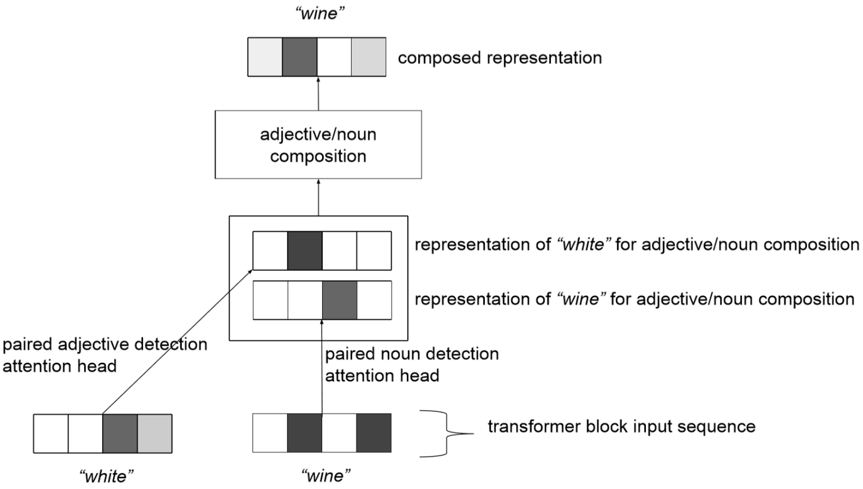
Layer: 0 Head: 0 Attention: All

Each word attends all other words in a sentence. This might allow a rough contextualization of each word.

The composition phase

In each layer, the outputs of all attention heads are concatenated and fed to a neural network that can represent complex nonlinear functions (needed for an expressive composition).

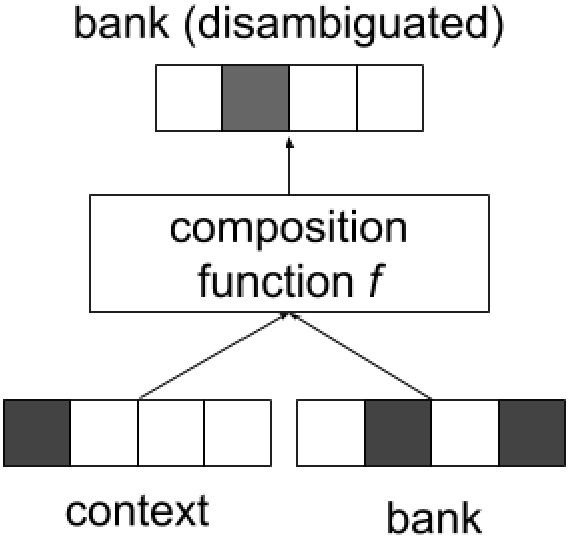
Relying on structured input from the attention heads, this neural network could perform various compositions. In the previously shown layer 5, the attention head #6 could guide the model to perform the following compositions: (*we*, *have*), (*if*, *we*), (*keep*, *up*) (*get*, *angry*). The model could combine them non-linearly and return a composed representation. Thus, the many attention heads can be used as tools that pave the way for composition.


“wine”
composed representation
adjective/noun composition
representation of “white” for adjective/noun composition
representation of “wine” for adjective/noun composition
paired adjective detection attention head
paired noun detection attention head
transformer block input sequence
“white”
“wine”

How attentions heads could be used to pave the way for specific composition such as adjective/nouns

While we didn’t find attention heads focusing on more consistent combinations such as adjective/nouns, there might be some common grounds between verb/adverb composition and other compositions that the model leverages.

There are many possible relevant compositions (words-subwords, adjective-noun, verb-preposition, clause-clause). To go even further, we can see disambiguation as a composition of an ambiguous word (e.g. bank) with its relevant context words (e.g. river or cashier). Integration of background common sense knowledge related to concept given a context could also be performed during a composition phase. Such disambiguation could also occur at other levels (e.g. sentence-level, clause-level).



Disambiguation as a composition

Besides, the composition might also be involved in word order reasoning. It has been argued that the positional encoding might not be sufficient to encode order of the words correctly. However, the positional encoding is designed to encode coarse, fine, and possibly *exact* positions of each token. (The positional encoding is a vector that is averaged with the input embedding in order to yield a position-aware representation of each token in the input sequence). Therefore, based on two positional encodings, the non-linear composition could theoretically perform some relational reasoning based on word relative positions.

We hypothesize that the composition phase also does the heavy lifting in BERT natural language understanding: Attention isn't all you need.

Wrapping up

We proposed an insight on the inductive bias of Transformers. However, we have to keep in mind that our interpretation might be optimistic regarding the capabilities of Transformer. As a reminder, LSTM were shown to be able to deal implicitly with tree structures [Bowman 2015] and composition [Tai 2015]. But LSTM has some limitations, some being due to vanishing gradient [Hochreiter 1998]. Thus, further work is needed to shed light on the limitations of the transformer.

References

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding [Devlin 2018]
Attention Is All You Need [Vaswani 2017]
Assessing BERT's Syntactic Abilities [Goldberg 2019]
Compositionality [Szabó 2017]
Frege in Space [Baroi, 2014]
Co-compositionality in Grammar [Pustejovsky 2017]
A Closer Look at Memorization in Deep Networks [Arpit 2017]
Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank [Socher 2013]
Improved Semantic Representations From Tree-Structured Long Short-Term Memory Networks [Tai 2015]
Tree-structured composition in neural networks without tree-structured architectures [Bowman 2015]
A Thousand Plateaus [Deleuze 1987]
The vanishing gradient problem during learning recurrent neural nets and problem solutions [Hochreiter 1998]

Deconstructing BERT, Part 2: Visualizing the Inner Workings of Attention [Vig 2019]