Are Representations Built from the Ground Up? An Empirical Examination of Local Composition in Language Models

Emmy Liu and **Graham Neubig**
Language Technologies Institute
Carnegie Mellon University
{mengyan3, gneubig}@cs.cmu.edu

Abstract

Compositionality, the phenomenon where the meaning of a phrase can be derived from its constituent parts, is a hallmark of human language. At the same time, many phrases are *non-compositional*, carrying a meaning beyond that of each part in isolation. Representing both of these types of phrases is critical for language understanding, but it is an open question whether modern language models (LMs) learn to do so; in this work we examine this question. We first formulate a problem of predicting the LM-internal representations of longer phrases given those of their constituents. We find that the representation of a parent phrase can be predicted with some accuracy given an affine transformation of its children. While we would expect the predictive accuracy to correlate with human judgments of semantic compositionality, we find this is largely *not* the case, indicating that LMs may not accurately distinguish between compositional and non-compositional phrases. We perform a variety of analyses, shedding light on when different varieties of LMs do and do not generate compositional representations, and discuss implications for future modeling work.[1]

1 Introduction

Compositionality is argued to be a hallmark of linguistic generalization (Szabó, 2020). However, some phrases are non-compositional, and cannot be reconstructed from individual constituents (Dankers et al., 2022a). Intuitively, a phrase like "I own cats and dogs" is locally compositional, whereas "It's raining cats and dogs" is not. Therefore, any representation of language must be easily composable, but it must also correctly handle cases that deviate from compositional rules.

Both lack (Hupkes et al., 2020; Lake and Baroni, 2017) and excess (Dankers et al., 2022b) of compositionality have been cited as common sources of errors in NLP models, indicating that models may handle phrase composition in an unexpected way.

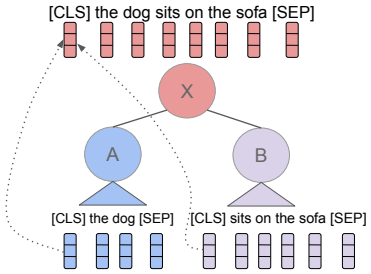

Figure 1: An illustration of the local composition prediction problem with [CLS] representations.

In general form, the compositionality principle is simply "the meaning of an expression is a function of the meanings of its parts and of the way they are syntactically combined" (Pelletier, 1994). However, this definition is underspecified (Partee, 1984). Recent efforts to evaluate the compositional abilities of neural networks have resulted in several testable definitions of compositionality (Hupkes et al., 2020).

Previous work on compositionality in natural language focuses largely on the definition of **substitutivity**, by focusing on changes to the constituents of a complex phrase and how they change its representation (Dankers et al., 2022a; Garcia et al., 2021; Yu and Ettinger, 2020). The definition we examine is **localism**: whether or not the representation of a complex phrase is derivable only from its local structure and the representations of its immediate "children" (Hupkes et al., 2020). A similar concept has been proposed separately to measure the compositionality of learned representations, which we use in this work (Andreas, 2019). We focus on localism because it is a more direct definition and does not rely on the collection of contrastive pairs of phrases. This allows us to examine a wider range of phrases of different types and lengths.

[1]Code and data available at https://github.com/nightingal3/lm-compositionality

In this paper, we ask whether reasonable compositional probes can predict an LM's representation of a phrase from its children in a syntax tree, and if so, which kinds of phrase are more or less compositional. We also ask whether this corresponds to human judgements of compositionality.

We first establish a method to examine local compositionality on phrases through probes that try to predict the representation of a parent given its children (section 2). We create two English-language datasets upon which to experiment: a large-scale dataset of 823K phrases mined from the Penn Treebank, and a new dataset of idioms and paired non-idiomatic phrases for which we elicit human compositionality judgements, which we call the **C**ompositionality of **H**uman-annotated **I**diomatic **P**hrases dataset (**CHIP**) (section 3).

For multiple models and phrase types, we find that phrase embeddings across models and representation types have a fairly predictable affine compositional structure based on embeddings of their constituents (section 4). We find that there are significant differences in compositionality across phrase types, and analyze these trends in detail, contributing to understanding how LMs represent phrases (section 5). Interestingly, we find that human judgments do not generally align well with the compositionality level of model representations (section 6). This implies there is still work to be done at the language modelling level to capture a proper level of compositionality in representations.

2 Methods and Experimental Details

2.1 Tree Reconstruction Error

We follow Andreas (2019) in defining deviance from compositionality as *tree reconstruction error*. Consider a phrase $x = [a][b]$, where a and b can be any length > 0. Assume we always have some way of knowing how x should be divided into a and b. Assume we also have some way of producing representations for x, a, and b, which we represent as a function r. Given representations $r(x)$, $r(a)$ and $r(b)$, we wish to find the function which most closely approximates how $r(x)$ is constructed from $r(a)$ and $r(b)$.

$$\hat{f} = \arg\min_{f \in \mathcal{F}} \frac{1}{|\mathcal{X}|} \sum_{x \in \mathcal{X}} \delta_{x,ab} \tag{1}$$

$$\delta_{x,ab} = d(r(x), f(r(a), r(b)) \tag{2}$$

Where $\mathcal{X}$ is the set of possible phrases in the language that can be decomposed into two parts, $\mathcal{F}$ is the set of functions under consideration, and d is a distance function. An example scenario is depicted in Figure 1.

For d, we use cosine distance as this is the most common function used to compare semantic vectors. The division of x into a and b is specified by syntactic structure (Chomsky, 1959). Namely, we use a phrase's annotated constituency structure and convert its constituency tree to a binary tree with the right-factored Chomsky Normal Form conversion included in NLTK (Bird and Loper, 2004).

2.2 Language Models

We study representations produced by a variety of widely used language models, specifically the `base-(uncased)` variants of Transformer-based models: **BERT**, **RoBERTa**, **DeBERTa**, and **GPT-2** (He et al., 2021; Liu et al., 2019; Devlin et al., 2019; Radford et al., 2019).

2.2.1 Representation extraction

Let $[x_0, ..., x_N]$ be a sequence of $N + 1$ input tokens, where x_0 is the `[CLS]` token if applicable, and x_N is the end token if applicable. Let $[h_0^{(i)}, ..., h_N^{(i)}]$ be the embeddings of the input tokens after the i-th layer.

For models with the `[CLS]` beginning of sequence token (BERT, RoBERTa, and DeBERTa), we extracted the embedding of the `[CLS]` token from the last layer, which we refer to as the **CLS** representation. For GPT-2, we extracted the last token, which serves a similar purpose. This corresponds to $h_0^{(12)}$ and $h_N^{(12)}$ respectively.

Alternately, we also averaged all embeddings from the last layer, including special tokens. We refer to this as the **AVG** representation.

$$\frac{1}{N+1} \sum_{i=0}^{N+1} h_i^{(12)} \tag{3}$$

2.3 Approximating a Composition Function

To use this definition, we need a composition function $\hat{f}$. We examine choices detailed in this section.

For parameterized probes, we follow the probing literature in training several probes to predict a property of the phrase given a representation of the phrase. However, in this case, we are not predicting a categorical attribute such as part of speech. Instead, the probes that we use aim to predict the

parent representation $r(x)$ based on the child representations $r(a)$ and $r(b)$. We call this an *approximative probe* to distinguish it from the usual use of the word probe.

2.3.1 Arithmetic Probes

In the simplest probes, the phrase representation $r(x)$ is computed by a single arithmetic operation on $r(a)$ and $r(b)$. We consider three arithmetic probes:[2]

$$\textbf{ADD}(r(a), r(b)) = r(a) + r(b) \quad (4)$$

$$\textbf{W1}(r(a), r(b)) = r(a) \quad (5)$$

$$\textbf{W2}(r(a), r(b) = r(b) \quad (6)$$

2.3.2 Learned Probes

We consider three types of learned probes. The linear probe expresses $r(x)$ as a linear combination of $r(a)$ and $r(b)$. The affine probe adds a bias term. The MLP probe is a simple feedforward neural network with 3 layers, using the ReLU activation.

$$\textbf{LIN}(r(a), r(b)) = \alpha_1 r(a) + \alpha_2 r(b) \quad (7)$$

$$\textbf{AFF}(r(a), r(b)) = \alpha_1 r(a) + \alpha_2 r(b) + \beta \quad (8)$$

$$\textbf{MLP}(r(a), r(b)) = W_3 h_2 \quad (9)$$

Where

$$h_1 = \sigma(W_1[r(a); r(b)])$$

$$h_2 = \sigma(W_2 h_1),$$

W_1 is (300×2), W_2 is (768×300), and W_3 is (1×768). We do not claim that this is the best MLP possible, but use it as a simple architecture to contrast with the linear models.

3 Data and Compositionality Judgments

3.1 Treebank

To collect a large set of phrases with syntactic structure annotations, we collected all unique subphrases (≥ 2 words) from WSJ and Brown sections of the Penn Treebank (v3) (Marcus et al., 1993). [3]

The final dataset consists of **823K** phrases after excluding null values and duplicates. We collected the length of the left child in words, the length of the right child in words, and the tree's production rule, which we refer to as *tree type*. There were 50260 tree types in total, but many of these are unique. Examples and phrase length distribution can be found in Appendix A, and Appendix B.

3.2 English Idioms and Matched Phrase Set

Previous datasets center around notable bigrams, some of which are compositional and some of which are non-compositional (Ramisch et al., 2016b; Reddy et al., 2011). However, there is a positive correlation between bigram frequency and human compositionality scores in these datasets, which means that it is unclear whether models are capturing compositionality or merely frequency effects if they correlate well with the human scores.

Because models are likely more sensitive to surface features of language than humans, we gathered a more controlled set of phrases to compare with human judgments.

Since non-compositional phrases are somewhat rare, we began with a set of seed idioms and bigrams from previous studies (Jhamtani et al., 2021; Ramisch et al., 2016b; Reddy et al., 2011). We used idioms because they are a common source of non-compositional phrases. Duplicates after lemmatization were removed.

For each idiom, we used Google Syntactic NGrams to find three phrases with an identical part of speech and dependency structure to that idiom, and frequency that was as close as possible relative to others in Syntactic Ngrams (Goldberg and Orwant, 2013).[4] For example, the idiom "sail under false colors" was matched with "distribute among poor parishioners". More examples can be found in Table 1. An author of this paper inspected the idioms and removed those that were syntactically analyzed incorrectly or offensive.

4 Approximating a Composition Function

4.1 Methods

To approximate the composition functions of models, we extract the **CLS** and **AVG** representations from each model on the Treebank dataset. We used 10-fold cross-validation and trained the learned probes on the 90% training set in each fold. The

[2] Initially, we considered the elementwise product $\textbf{PROD}(r(a), r(b)) = r(a) \odot r(b)$, but found that it was an extremely poor approximation.

[3] We converted the trees to Chomsky Normal Form with right-branching using NLTK (Bird and Loper, 2004). We note that not all subtrees are syntactically meaningful. However, we used this conversion to standardize the number of children and formatting. We exclude phrases with a null value for the left or right branch (Bies et al., 1995).

[4] The part of speech/dependency pattern for each idiom was taken to be the most common pattern for that phrase in the dataset