Do Attention Heads in BERT Track Syntactic Dependencies?

Anonymous

Abstract

We investigate the extent to which individual attention heads in BERT implicitly capture the syntactic dependency relations at each layer. Our analysis shows that the simple heuristic of taking the largest attention weight between words of BERT for different layers and heads can consistently identify more than 75% of Universal Dependencies relation types better than linguistically uninformed baselines on parsed English text, suggesting that some self-attention weights in BERT act as a proxy for syntactic structure, at least to a degree. Additionally, as previous work has shown that fine-tuning BERT on an intermediate data-rich supervised tasks improve its performance on the GLUE benchmark, we fine-tune BERT on the syntax-oriented Corpus of Linguistic Acceptability (CoLA) and the semantics-oriented Multi-Genre Natural Language Inference (MNLI) dataset to investigate whether fine-tuning affects the patterns of BERT’s self-attention. Although the fine-tuned BERT models show different performance from BERT on the downstream tasks such as the GLUE benchmark, we do not observe significant difference in the dependency relations extracted using our methods. Our results suggest that the attention heads of both BERT and fine-tuned BERT models do not learn to track dependency structure significantly better than our baselines and that attention does not reveal the significant syntactic knowledge that BERT is known to learn.

1 Introduction

Pretrained Transformer models like OpenAI GPT (Radford et al., 2018) and BERT (Devlin et al., 2019) have shown stellar performance on language understanding tasks. BERT and BERT-based models significantly improve the state-of-the-art on many tasks such as constituency parsing (Kitaev and Klein, 2018), question answering (Rajpurkar et al., 2016), and have attained top positions on the GLUE leaderboard (Wang et al., 2019). As BERT becomes a staple component of many NLP models, many researchers have attempted to analyze the linguistic knowledge that BERT has learned by analyzing the BERT model (Goldberg, 2018) or training probing classifiers on the contextualized embeddings of BERT (Tenney et al., 2019).

BERT, as a Transformer-based language model, computes the hidden representation at each layer for each token by attending to all the tokens in an input sentence. The attention heads of Transformer have been claimed to capture the syntactic structure of the sentences (Vaswani et al., 2017). Intuitively, for a given token, some specific tokens in the sentence would be more linguistically related to it than the others, and therefore the self-attention mechanism should be expected to allocate more weight to the linguistically related tokens in computing the hidden state of the given token. In this work, we aim to investigate the hypothesis that syntax is implicitly encoded by BERT’s self-attention heads. We use two relation extraction methods to extract dependency relations from all the self-attention heads of BERT. We analyze the resulting dependency relations to investigate whether the attention heads of BERT implicitly track syntactic dependencies significantly better than chance, and what type of dependency relations BERT learn.

We extract the dependency relations from the self-attention heads instead of the contextualized embeddings of BERT. In contrast to probing models, our dependency extraction methods require no further training. Our experiments suggest that the attention heads of BERT encode most dependency relation types with substantially higher accuracy than our baselines—a randomly initialized Transformer and relative positional baselines. Fine-
tuning BERT on the syntax-oriented CoLA does not appear to impact the accuracy of extracted dependency relations. However, when fine-tuned on the semantics-oriented MNLI dataset, there is a slight improvement in accuracy for longer-term clausal relations and a slight loss in accuracy for shorter-term relations. Overall, while BERT models obtain non-trivial accuracy for some dependency types such as nsubj, obj, nmod, aux, and conj, they do not substantially outperform the trivial right-branching trees in terms of undirected unlabeled attachment scores (UUAS). Therefore, although the attention heads of BERT reflect a small number of dependency relation types, it does not reflect the full extent of the significant amount of syntactic knowledge BERT is shown to learn by the previous probing work.

2 Related Work

There has been substantial work so far on extracting syntactic trees from the attention heads of Transformer-based neural machine translation (NMT) models. Mareček and Rosa (2018) aggregate the attention weights across the self-attention layers and heads to form a single attention weight matrix. Using this matrix, they propose a method to extract constituency and (undirected) dependency trees by recursively splitting and constructing the maximum spanning tree respectively. In contrast, Raganato and Tiedemann (2018) train Transformer-based machine translation model on different language-pairs and extract the dependency trees using the maximum spanning tree algorithm on the attention weights of the encoder for each layer and head individually. In work concurrent with ours, Voita et al. (2019) focus on finding confident attention heads of the Transformer encoder based on a heuristic of the concentration of attention weights on single tokens. They identify that these heads appear to serve three specific functions: attending to relative positions, syntactic relations, and rare words.

Prior work on the analysis of the contextualized embeddings of BERT has shown that BERT learns significant knowledge of syntax (Goldberg, 2018). Tenney et al. (2019) introduce a probing-style method for evaluating syntactic knowledge in BERT and show that BERT encodes syntax more than semantics. Hewitt and Manning (2019) train a structural probing model that maps the hidden representations of each token to an inner-product space that corresponds to syntax tree distance. They show that the learned spaces of strong models such as BERT and ELMo (Peters et al., 2018) are better able to reconstruct dependency trees compared to baselines that can encode features for training a parser but aren’t capable of parsing themselves.

3 Methods

3.1 Models

BERT (Devlin et al., 2019) is a Transformer-based masked language model pretrained on BooksCorpus (Zhu et al., 2015) and English Wikipedia that has attained stellar performance on a variety of downstream NLP tasks. We run our experiments on the pretrained cased and uncased versions of the BERT-large model, which is a Transformer model consisting of 24 self-attention layers with 16 heads each. For a given dataset, we feed each input sentence through BERT and capture the attention weights for each individual head and layer. Phang et al. (2018) report that they achieve performance gains on the GLUE benchmark by supplementing pre-trained BERT with data-rich supervised tasks such as the Multi-Genre Natural Language Inference dataset (MNLI; Williams et al., 2018). Therefore, we also run experiments on the uncased BERT-large model fine-tuned on the Corpus of Linguistic Acceptability (CoLA; Warstadt et al., 2018) and MNLI, to investigate the impact of fine-tuning on a syntax-related task (CoLA) and a semantic-related task (MNLI) on the structure of attention weights and resultant extracted dependency relations. We refer to these fine-tuned models as CoLA-BERT and MNLI-BERT. As a baseline, we apply the same relation extraction methods to the BERT-large model with randomly initialized weights (which we refer to as random BERT) as the previous work has shown that randomly initialized sentence encoders perform surprisingly well on a suite of NLP tasks (Zhang and Bowman, 2018; Wieting and Kiela, 2019).

3.2 Relation Extraction Methods

We aim to test the hypothesis that the attention heads of BERT learn syntactic relations implicitly, and that self-attention between two words encodes information about their dependency relation. We use two methods for extracting relations from the attention weights in BERT. Both methods oper-
ate on the weight matrix $W \in (0, 1)^{T \times T}$ for a
given head at a given layer, where $T$ is the num-
ber of tokens in the sequence, and the rows and
columns correspond to the attending and attended
tokens respectively (such that each row sums to
1). We exclude $[\text{CLS}]$ and $[\text{SEP}]$ tokens from
the attention matrices, which allows us to focus
on inter-word attention. Where the tokenization
of our parsed corpus does not match the BERT
tokenization, we merge the non-matching tokens
until they are mutually compatible, and sum the
attention weights for the corresponding columns
and rows. We then apply either of the two extrac-
tion methods to the attention matrix. To handle
the subtokens within the merged tokens, we set all
subtokens except for the first to depend on the first
subtoken. This approach is largely similar to that
in Hewitt and Manning (2019). We use the English
Parallel Universal Dependencies (PUD) treebank
from the CoNLL 2017 shared task (Zeman et al.,
2018) as gold standard for our evaluation.

**Maximum Attention Weights**  We assign a re-
lation $(w_i, w_j)$ between word $w_i$ and $w_j$ if $j = \arg\max_i W[i]$ for each row $i$ in attention matrix
$W$. Based on this simple method, we extract rela-
tions for all sentences in our evaluation datasets.
The relations extracted using this method need not
form a valid tree, or even be fully connected. The
resulting edge directions may or may not match
the canonical directions in a tree, so we evaluate
the resulting arcs as undirected.

**Maximum Spanning Tree**  To extract valid de-
pendency trees from the attention weights for a
given layer and head, we follow the approach
of Raganato and Tiedemann (2018) and treat the
matrix of attention weight tokens as a complete
weighted directed graph, with the edges point-
ing from the output token to each attended token.
As in Raganato and Tiedemann, we take the root
of the gold dependency tree as the starting node
and apply the Chu-Liu-Edmonds algorithm (Chu,
1965; Edmonds, 1967) to compute the maximum
spanning tree. The resulting tree is a valid undi-
rected dependency tree.

**Relative position baselines**  Many dependency
relations tend to occur in specific positions relative
to the parent word. For example, $\text{nsubj}$ mostly oc-
curs between a verb and the adjacent word before
verb. As an example, Figure 1 shows the distribu-
tion of relative positions for four major UD rela-
tions in our data. Following Voita et al. (2019),
we compute the most common positional offset
between a parent and child word for a given de-
pendency relation, and formulate a baseline based
on that most common relative positional offset.

### 4 Results

Figure 2 and Table 1 describes the accuracy for
$\text{nsubj}$, $\text{obj}$, $\text{advmod}$, and $\text{amod}$ and the 10 most
frequent relation types in the dataset using rela-
tions extracted based on the maximum attention
weight method. We also include $\text{advcl}$ and $\text{csubj}$ in
Table 1 as it shows the behavior of MNLI-BERT
that tends to track longer-term clausal dependen-
cies better than BERT and CoLA-BERT. Addition-
ally, Figure 3 shows the accuracy for $\text{nsubj}$,
$\text{obj}$, $\text{advmod}$, and $\text{amod}$ relations extracted based
on the maximum spanning tree algorithm. The
pre-trained and fine-tuned BERT models out-
perform random BERT substantially for all depen-
dency types. They also outperform the relative
position baselines for more than 75% of relation
types. They outperform all baselines by a large
margin for $\text{nsubj}$ and $\text{obj}$, but only slightly bet-
ter for $\text{advmod}$ and $\text{amod}$. These results suggest
that the self-attention weights in trained BERT
models implicitly encode certain dependency rela-
tions. Moreover, we do not observe very substan-
tial changes in accuracy by fine-tuning on CoLA
and MNLI. However, both BERT and CoLA-
BERT have similar or slightly better performance
than MNLI-BERT, except for clausal dependen-
cies such as $\text{advcl}$ (adverbial clause modifier) and
$csubj$ (clausal subject) where MNLI-BERT out-
performs BERT and CoLA-BERT by more than
5 absolute points in accuracy. This suggests that
semantic-oriented fine-tuning task encourages ef-
effective long-distance dependencies.

Figure 4 describes the maximum undirected un-
labeled attachment scores (UUAS) across each
Table 1: Highest accuracy for the most frequent dependency types (excluding nsubj, obj, advmod, and amod). We include advcl and csubj although they are not among the ten most frequent relation types as MNLI-BERT outperform other models for these dependency types. **Bold** marks the highest accuracy for each dependency type. *Italics* marks accuracies that outperform our trivial baselines.

<table>
<thead>
<tr>
<th>Model</th>
<th>advcl</th>
<th>csubj</th>
<th>case</th>
<th>det</th>
<th>obl</th>
<th>nmod</th>
<th>punct</th>
<th>aux</th>
<th>conj</th>
<th>cc</th>
<th>mark</th>
<th>compound</th>
</tr>
</thead>
<tbody>
<tr>
<td>BERT (cased)</td>
<td>29.4</td>
<td>55.6</td>
<td>83.8</td>
<td>93.2</td>
<td>31.0</td>
<td>51.7</td>
<td>40.2</td>
<td>80.0</td>
<td>45.0</td>
<td>73.5</td>
<td>72.1</td>
<td>80.6</td>
</tr>
<tr>
<td>BERT</td>
<td>26.6</td>
<td>48.1</td>
<td>88.8</td>
<td>96.3</td>
<td>32.1</td>
<td>66.7</td>
<td>40.5</td>
<td>81.0</td>
<td>48.9</td>
<td>67.4</td>
<td>72.1</td>
<td>82.8</td>
</tr>
<tr>
<td>CoLA-BERT</td>
<td>28.0</td>
<td>51.9</td>
<td>89.5</td>
<td>96.2</td>
<td>33.8</td>
<td>66.2</td>
<td>41.1</td>
<td>82.2</td>
<td>50.6</td>
<td>68.5</td>
<td>70.8</td>
<td>82.1</td>
</tr>
<tr>
<td>MNLI-BERT</td>
<td>34.5</td>
<td>63.0</td>
<td>87.9</td>
<td>95.3</td>
<td>32.4</td>
<td>63.3</td>
<td>41.3</td>
<td>78.6</td>
<td>50.5</td>
<td>65.5</td>
<td>68.5</td>
<td>80.6</td>
</tr>
<tr>
<td>Positional Baselines</td>
<td>10.23</td>
<td>25.9</td>
<td>38.7</td>
<td>56.7</td>
<td>24.0</td>
<td>35.4</td>
<td>18.6</td>
<td>55.5</td>
<td>27.8</td>
<td>43.4</td>
<td>53.7</td>
<td>82.7</td>
</tr>
<tr>
<td>Random-BERT</td>
<td>13.3</td>
<td>22.2</td>
<td>13.7</td>
<td>12.6</td>
<td>13.5</td>
<td>13.8</td>
<td>12.6</td>
<td>16.3</td>
<td>18.9</td>
<td>20.9</td>
<td>12.8</td>
<td>14.8</td>
</tr>
</tbody>
</table>

5 Conclusion

In this work, we investigate whether the attention heads of BERT exhibit the implicit syntax dependency by extracting and analyzing the dependency relations from the attention heads of BERT at all layers. We use two simple dependency relation extraction methods that require no additional training, and observe that there are attention heads of BERT that track more than 75% of the dependency types with higher accuracy than our baselines. However, the hypothesis that the attention heads of BERT track syntactic dependencies is not well-supported as the linguistically uninformed baselines outperform BERT on nearly 25% of the dependency types. Additionally, BERT’s performance in terms of UUAS is only slightly higher than that of the trivial right-branching trees, suggesting that the dependency syntax learned by the attention heads is trivial. Additionally, we observe that fine-tuning on the CoLA and MNLI does not affect the pattern of self-attention, although the fine-tuned models show different performance from BERT on the GLUE benchmark.
References


