Under the Hood: Using Diagnostic Classifiers to Investigate and Improve how Language Models Track Agreement Information

Mario Giulianelli, Jack Harding, Florian Mohnert, Dieuwke Hupkes and Willem Zuidema

Presented by: Ziyad Alsaeed

Jan 24, 2020
Motivation

• Understand how neural language models represent syntactic features through diagnostic classifiers.

• Use insight found from using diagnostic classifiers to influence the LSTM models for processing difficult sentences.

• Eventually the goal is to increase the LSTM overall accuracy.
Language Short and Long Distance Relations.

- LSTM models perform well on short and especially long distance relations in a sentence or a discourse.
  - E.g. subject-verb agreement.

- It is not clear which component of published LSTM models are responsible for sorting or processing syntactic features or how are they represented.

- Different approaches are proposed to investigate the state of LSTM models such as visualizing the state space, performing ablation to the network or using the internal state of the networks for some auxiliary task.
Subject-Verb Agreement

• In English, the verb and the head of its syntactic subject must agree on their number (singular or plural).

• Example:
  • *The dog chases the cat.*
  • *The dog chase the cat.*

• K: minimal number of word appearing before the subject.

• L: *Context size*, which is the number of tokens between the subject head and the verb.

• M: minimal number of word after the verb.

• A: the position of the attractor relative to the subject.
  • *Attractors*: number of intervening nouns between subject head and verb.
Datasets

• Dataset introduced by Gulordava et al. (2018):
  • Contains 410 sentences.
  • Each sentence has at least three tokens occurring between the subject head and the verb.
  • Every sentence is annotated with correct and incorrect verb forms, the position of the subject head and the verb, and the number of agreement attractors.

• Wikipedia corpus by Linzen et al. (2016):
  • Contains ~1.5 million annotated sentences.
  • Allows for obtaining a subset of sentences with specific attributes (e.g. number of word before subject and after verb or the number of attractors).
The average of estimates of the 10 economists polled puts the dollar around 1.820 marks.
Diagnostic Classifiers

• The key idea of DC is to test whether an LSTM’s intermediate representations contains information about a particular phenomenon.
  • E.g. Subject-verb agreement.

• Achieve the diagnostic goal by training another model to recognize the information relevant to the phenomenon in the internal activation of the LSTM.

• A trained DC model that succeeds in predicting the correct label from the given intermediate representation constitute evidence that the LSTM is in fact computing or keeping track of the hypothesized information.
First Experiment

• Use 1000 sentences with the following attributes (K1 – L5 – M1 – A*).

• Extract activation data for both hidden and gate activation (the hidden activation $h_t$, memory cell $c_t$, forget gate $f_t$, input gate $i_t$, and output gate $o_t$) of the two layered model trained by Gulordava et al. (2018).
  • For example, for a single sentence of length $n$ they obtain $5 \times 2 \times n$ activation vectors, because they have 2 layers, $n$ timesteps, and 5 types of activations at each time step.

• Label all activations with the number of the main verb of the sentence (either singular or plural).

• Train separate DCs for each of the 10 components of the LSTM.
Result of First Experiment

Accuracies over time (on WD-K1-L5-M1-A3) of 10 diagnostic classifiers trained and tested on data from different components of the LSTM. As in this test set one word occurs before the subject, the subject is at timestep 1. Green lines represent sentences for which the LSTM predicts the correct verb, blue lines sentences for which the LSTM assigns a higher probability to the incongruent counterpart.
Second Experiment

• Created a corpus of sentence with the following attributes (K* - L5 - M* - A3).

• Train separate DCs for each timestep.
  • Each $DC_t$ is trained with activation data at timestep $t$ only.

• Test each $DC_t$ on data from all other timesteps.
  • This would result on of $T \times T$ DC-accuracies where $T$ is the total of timesteps.
Results of Second Experiment

Correctly classified sentences

Incorrectly classified sentences

The temporal generalization matrices for DCs trained on memory cell activation at different timesteps. Timestep 0 corresponds to the subject of the sentence, the attractor and main verb of the sentence occur at timesteps 3 and 6, respectively. The corpus used for testing here is \textit{WD-K*-L5-M*-A3}.
Third Experiment

• Use the same training set as from experiment-2 (K* - L5 - M* - A3).

• Instead of focusing on timesteps, the DCs are trained on different components at a fixed timestep (timestep-4).
Result of Third Experiment

Correctly classified sentences

Incorrectly classified sentences

The spatial generalization matrices at timestep 4. Shown are accuracies of DCs trained on activation data of each component separately (horizontal), and tested on each component separately (vertical).
Improving LSTM Models Using DCs

• Use the same dataset as from the last two experiments (K0 – L5 – M0 – A3).

• Train 4 DCs to predict the number of the sentence from the hidden layer activations and memory cell activations for both layers.

• Use trained DCs to actively influence the course of processing by the LSTM.
  • After processing the subject of the sentence, they halt the LSTM processing, extract the hidden activation and the activation of memory cell, and apply the trained DC to predict whether the main verb in the sentence is singular or plural.
  • They then slightly adapt the activations based on the error that is defined by the difference between the predicted label and the correct label of the particular sentence.
  • After adapting the activation they continue processing the rest of the sentence.
Effect of Intervention

• The intervention shows a significant increase in the performance of the LSTM model.

• From the results, it is evident that DCs are able to pick up features that are actually used by the LSTM.

• The result shows that DCs can be used not only for analyzing language model, but actively influencing black box neural models.

<table>
<thead>
<tr>
<th></th>
<th>without intervention</th>
<th>with intervention</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original</td>
<td>78.1</td>
<td>85.4</td>
</tr>
<tr>
<td>Nonce</td>
<td>70.7</td>
<td>75.6</td>
</tr>
</tbody>
</table>
Comments