Probing / interpretability

CS 685, Spring 2024
Introduction to Natural Language Processing
http://people.cs.umass.edu/~miyyer/cs685/

Mohit Iyyer
College of Information and Computer Sciences
University of Massachusetts Amherst

most slides from Tu Vu
understanding representations

two prominent methods

• visualization

• linguistic probe tasks
Sentiment neuron

While training the linear model with L1 regularization, we noticed it used surprisingly few of the learned units. Digging in, we realized there actually existed a single “sentiment neuron” that’s highly predictive of the sentiment value.

The sentiment neuron within our model can classify reviews as negative or positive, even though the model is trained only to predict the next character in the text.
what is a linguistic probe task?

given an encoder model (e.g., BERT) pre-trained on a certain task, we use the representations it produces to train a classifier (without further fine-tuning the model) to predict a linguistic property of the input text
predict the length (number of tokens) of the input sentence $s$
predict the length (number of tokens) of the input sentence $s$

BERT [CLS] representation, kept frozen

sentence length

(probe network)

(sent. repr.)

(classifier)
sentence length

predict the length (number of tokens) of the input sentence $s$

probe network

classifier

Feed-forward NN trained from scratch

sent. repr.

BERT [CLS] representation, kept frozen

(Adi et al., 2017)
sentence length
predict the length (number of tokens) of the input sentence $s$

probe network  
classifier  
sent. repr.

word content
predict if word $w$ appears in sentence $s$

classifier  

sent. repr.  
word repr.
sentence length

predict the length (number of tokens) of the input sentence $s$

probe network

word content

predict if word $w$ appears in sentence $s$

probe network

BERT [CLS] representation, kept frozen

Possibly BERT subword embedding

(Adi et al., 2017)
sentence length
predict the length (number of tokens) of the input sentence $s$

word content
predict if word $w$ appears in sentence $s$

word order
predict whether $w_1$ appears before or after $w_2$ in the sentence $s$
token labeling: POS tagging
predict a POS tag for each token

segmentation: NER
predict the entity type of the input token

pairwise relations: syntactic dep. arc
predict if there is a syntactic dependency arc between tok_1 and tok_2

(Liu et al., 2019)
edge probing: coreference

predict whether two spans of tokens ("mentions") refer to the same entity (or event)

(Tenney et al., 2019)
motivation of probe tasks

• if we can train a classifier to predict a property of the input text based on its representation, it means the property is encoded somewhere in the representation

• if we cannot train a classifier to predict a property of the input text based on its representation, it means the property is not encoded in the representation or not encoded in a useful way, considering how the representation is likely to be used
characteristics of probe tasks

• usually classification problems that focus on simple linguistic properties

• ask simple questions, minimizing interpretability problems

• because of their simplicity, it is easier to control for biases in probing tasks than in downstream tasks

• the probing task methodology is agnostic with respect to the encoder architecture, as long as it produces a vector representation of input text

• does not necessarily correlate with downstream performance

(Conneau et al., 2018)
probe approach

predict a linguistic property of the input

the classifier’s weights are updated

train the classifier only

the encoder’s weights are fixed

no further fine-tuning

Encoder Layer $N \times$
List Maximum (Classification)

<table>
<thead>
<tr>
<th>Target</th>
<th>0</th>
<th>0</th>
<th>1</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prediction</td>
<td>0.2</td>
<td>0.1</td>
<td>0.4</td>
<td>0.1</td>
<td>0.2</td>
</tr>
<tr>
<td>Probing Model</td>
<td>BiLSTM</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Embeddings</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pretrained Embedder</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Decoding (Regression)

<table>
<thead>
<tr>
<th>Target</th>
<th>9.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prediction</td>
<td>8.5</td>
</tr>
<tr>
<td>Probing Model</td>
<td>MLP</td>
</tr>
<tr>
<td>Embeddings</td>
<td></td>
</tr>
<tr>
<td>Pretrained Embedder</td>
<td></td>
</tr>
</tbody>
</table>

Addition (Regression)

<table>
<thead>
<tr>
<th>Target</th>
<th>9.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prediction</td>
<td>9.1</td>
</tr>
<tr>
<td>Probing Model</td>
<td>MLP</td>
</tr>
<tr>
<td>Embeddings</td>
<td></td>
</tr>
<tr>
<td>Pretrained Embedder</td>
<td></td>
</tr>
</tbody>
</table>

Memory

Query

Answer

KG

DANTE
born-in
FLORENCE

Symbolic Memory Access

“Dante was born in [MASK].”

LM

Neural LM Memory Access

Florence
lowest layers focus on local syntax, while upper layers focus more semantic content

(Peters et al., 2018)
BERT represents the steps of the traditional NLP pipeline:
POS tagging $\rightarrow$ parsing $\rightarrow$ NER $\rightarrow$
semantic roles $\rightarrow$ coreference

The expected layer and center-of-gravity for various NLP tasks:

- **POS**
  - Expected layer: 3.39
  - Center-of-gravity: 11.68

- **Constituents (Consts.)**
  - Expected layer: 3.79
  - Center-of-gravity: 13.06

- **Dependent arcs (Dep.)**
  - Expected layer: 5.69
  - Center-of-gravity: 13.75

- **Named entities (Entities)**
  - Expected layer: 4.64
  - Center-of-gravity: 13.16

- **Semantic roles (SRL)**
  - Expected layer: 6.54
  - Center-of-gravity: 13.63

- **Coreference (Coref.)**
  - Expected layer: 9.47
  - Center-of-gravity: 15.80

- **Semantic roles patterns (SPR)**
  - Expected layer: 9.93
  - Center-of-gravity: 12.72

- **Relations**
  - Expected layer: 9.40
  - Center-of-gravity: 12.83

(Tenney et al., 2019)
probe complexity

arguments for “simple” probes

we want to find easily accessible information in a representation

arguments for “complex” probes

useful properties might be encoded non-linearly

(Hewitt et al., 2019)
control tasks

控制任务

控制任务词汇

<table>
<thead>
<tr>
<th>句子1</th>
<th>The</th>
<th>cat</th>
<th>ran</th>
<th>quickly</th>
<th>.</th>
</tr>
</thead>
<tbody>
<tr>
<td>控制任务</td>
<td>10</td>
<td>37</td>
<td>10</td>
<td>15</td>
<td>3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>句子2</th>
<th>The</th>
<th>dog</th>
<th>ran</th>
<th>after</th>
<th>!</th>
</tr>
</thead>
<tbody>
<tr>
<td>控制任务</td>
<td>10</td>
<td>15</td>
<td>10</td>
<td>42</td>
<td>42</td>
</tr>
</tbody>
</table>

(Hewitt et al., 2019)
designing control tasks

• independently sample a control behavior $C(v)$ for each word type $v$ in the vocabulary

• specifies how to define $y_i \in Y$ for a word token $x_i$ with word type $v$

• control task is a function that maps each token $x_i$ to the label specified by the behavior $C(x_i)$

$$f_{\text{control}}(x_{1:T}) = f(C(x_1), C(x_2), \ldots C(x_T))$$

(Hewitt et al., 2019)
selectivity: high linguistic task accuracy + low control task accuracy

measures the probe model’s ability to make output decisions independently of linguistic properties of the representation

(Hewitt et al., 2019)
be careful about probe accuracies

| Model | Linear  |  | MLP-1  |  |
|-------|---------|------------------|------------------|
|       | Accuracy| Selectivity      | Accuracy          | Selectivity      |
| Proj0 | 96.3    | 20.6             | 97.1             | 1.6              |
| ELMo1 | 97.2    | 26.0             | 97.3             | 4.5              |
| ELMo2 | 96.6    | 31.4             | 97.0             | 8.8              |

**Part-of-speech Tagging**
how to use probe tasks to improve downstream task performance?

• what kinds of linguistic knowledge are important for your task?

• probe BERT for them

• if BERT struggles then fine-tune it with additional probe objectives

\[ L_{new} = L_{BERT} + \alpha L_{probe} \]
Editing knowledge in LLMs

\[ h_{\text{entity}} \quad \text{LM encoding} \quad \text{LM pred} \quad \text{the store.} \]

John went to work at

LM GENERATION

(Hernandez et al., ICLR 2023)
Editing knowledge in LLMs

$z = h_{\text{entity}} + Wh_{\text{attr}} + b$

(Hernandez et al., ICLR 2023)
<table>
<thead>
<tr>
<th>Entity</th>
<th>Correct Information</th>
<th>Incorrect Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Leonhard Euler</td>
<td>domain of activity is opera</td>
<td>× <strong>Leonhard Euler</strong> is the most prolific mathematician of the 18th century. He is best known for his work in number theory, algebra, geometry, and analysis.</td>
</tr>
<tr>
<td>Microsoft Internet Explorer 6</td>
<td>a product created by Google</td>
<td>× <strong>Microsoft Internet Explorer 6</strong> is a web browser developed by Microsoft for Windows. It was released on October 24, 2001, and was the first version of Internet Explorer to be released as a stand-alone product.</td>
</tr>
<tr>
<td>Beef bourguignon</td>
<td>that was formulated in Canada</td>
<td>× <strong>Beef bourguignon</strong> is a French dish of braised beef in red wine, onions, and mushrooms. It is a classic of French cuisine.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>✓ <strong>Beef bourguignon</strong> is a Canadian dish. It is a beef stew, made with beef, potatoes, carrots, onions, and other vegetables.</td>
</tr>
</tbody>
</table>

(Hernandez et al., ICLR 2023)
Induction heads
https://transformer-circuits.pub/2022/
in-context-learning-and-induction-heads/index.html