CS388: Natural Language Processing
Lecture 14: Interpretability

Greg Durrett
Announcements

- Project 2, FPs back today
- Project 3 due in a week
Recap: Instruction Tuning

- T0: tries to deliver on the goal of T5 and do many tasks with one model
- **Crowdsourced prompts:** instructions for how to do the tasks

Sanh et al. (2021)
Recap: RLHF

- Apply this approach to optimizing outputs from large language models
- Step 3 (not shown): do RL with this policy

Ouyang et al. (2022)
Today

- We’ve seen a lot of results from black box neural networks. Why can’t we just look at why they make their predictions?
- Interpreting neural networks: what does this mean and why should we care?
- Local explanations: erasure techniques
- Gradient-based methods
- Evaluating explanations
Interpreting Neural Networks
Interpreting Neural Networks

- This is a BERT-based QA model. How do we figure out why it picked Stewart over Devin Funchess?

  **Question:** who caught a 16-yard pass on this drive?
  **Answer:** devin funchess

  **Start Distribution**

  - Green: Heatmap of posterior probabilities over the start of the answer span

there would be no more scoring in the third quarter, but early in the fourth, the broncos drove to the panthers 41-yard line. on the next play, ealy knocked the ball out of manning’s hand as he was winding up for a pass, and then recovered it for carolina on the 50-yard line. a 16-yard reception by devin funchess and a 12-yard run by stewart then set up gano’s 39-yard field goal, cutting the panthers deficit to one score at 16â€“10. the next three drives of the game would end in punts.
Interpreting Neural Networks

the movie was not bad -> **negative** (gold: **positive**)

<table>
<thead>
<tr>
<th>Left side highlights: predictions model makes on individual words</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tells us how these words combine</td>
</tr>
<tr>
<td>What does this experiment tell us?</td>
</tr>
</tbody>
</table>

The table below shows examples of how DAN differs from ground truth:

<table>
<thead>
<tr>
<th>This movie was</th>
<th>DAN</th>
<th>Ground Truth</th>
</tr>
</thead>
<tbody>
<tr>
<td>not good</td>
<td>negative</td>
<td>negative</td>
</tr>
<tr>
<td>good</td>
<td>positive</td>
<td>positive</td>
</tr>
<tr>
<td>bad</td>
<td>negative</td>
<td>negative</td>
</tr>
<tr>
<td>not bad</td>
<td>negative</td>
<td>positive</td>
</tr>
</tbody>
</table>

Iyyer et al. (2015)
Why explanations?

- **Trust:** if we see that models are behaving in human-like ways and making human-like mistakes, we might be more likely to trust them and deploy them

- **Causality:** if our classifier predicts class $y$ because of input feature $x$, does that tell us that $x$ causes $y$? Not necessarily, but it might be helpful to know

- **Informativeness:** more information may be useful (e.g., predicting a disease diagnosis isn’t that useful without knowing more about the patient’s situation)

- **Fairness:** ensure that predictions are non-discriminatory

Lipton (2016)
Why explanations?

- Some models are naturally **transparent**: we can understand why they do what they do (e.g., a decision tree with <10 nodes)

- Explanations of more complex models
  
  - **Local explanations**: highlight what led to this classification decision. (Counterfactual: if these features were different, the model would’ve predicted a different class) — focus of this lecture
  
  - **Text explanations**: describe the model’s behavior in language
  
  - **Model probing**: auxiliary tasks, challenge sets, adversarial examples to understand more about how our model works

Lipton (2016); Belinkov and Glass (2018)
Local Explanations

(which parts of the input were responsible for the model’s prediction on this particular data point?)
Sentiment Analysis with Attention

Similar to a DAN model, but (1) extra BiLSTM layer; (2) attention layer instead of just a sum

Jain and Wallace (2019)
Attention Analysis

BiLSTM encoder

- Attention places most mass on *good* — did the model ignore *not*?
- What if we removed *not* from the input?

Jain and Wallace (2019)
Attention Analysis

- They show it is possible to modify attention while preserving the prediction probabilities
- Does this convince you that explanation is not helpful?

Jain and Wallace (2019)
Local Explanations

› An explanation could help us answer counterfactual questions: if the input were $x'$ instead of $x$, what would the output be?

\[ \text{that movie was not great, in fact it was terrible!} \]

\[ \text{that movie was not } \_\_\_, \text{ in fact it was terrible!} \]

\[ \text{that movie was } \_\_\_\_ \text{ great, in fact it was } \_\_\_\! \]

› Attention can’t necessarily help us answer this!
Erasure Method

- Delete each word one by one and see how prediction prob changes

<table>
<thead>
<tr>
<th>Deletion</th>
<th>Prob</th>
</tr>
</thead>
<tbody>
<tr>
<td>that movie was not great, in fact it was terrible!</td>
<td>0.97</td>
</tr>
<tr>
<td>____ movie was not great, in fact it was terrible!</td>
<td>0.97</td>
</tr>
<tr>
<td>that ____ was not great, in fact it was terrible!</td>
<td>0.98</td>
</tr>
<tr>
<td>that movie ____ not great, in fact it was terrible!</td>
<td>0.97</td>
</tr>
<tr>
<td>that movie was ___ great, in fact it was terrible!</td>
<td>0.8</td>
</tr>
<tr>
<td>that movie was not ____ , in fact it was terrible!</td>
<td>0.99</td>
</tr>
</tbody>
</table>
Erasure Method

- Output: highlights of the input based on how strongly each word affects the output
  
  *that movie was* <span style='color: red'>not</span> <span style='color: green'>great</span>, *in fact it was terrible!*

- *not* contributed to predicting the negative class (removing it made it less negative), *great* contributed to predicting the positive class (removing it made it more negative)

- Will this work well?
  - Inputs are now unnatural, model may behave in “weird” ways
  - Saturation: if there are two features that each contribute to negative predictions, removing each one individually may not do much
LIME

- Locally-interpretable, model-agnostic explanations (LIME)

- Similar to erasure method, but we’re going to delete collections of things at once
  - Can lead to more realistic input (although people often just delete words with it)
  - More scalable to complex settings
LIME

- Break input into components (for text: could use words, phrases, sentences, ...)
- Check predictions on subsets of those
- Now we have model predictions on perturbed examples

This is what the model is doing on perturbed examples of the input.

Now we train a classifier to predict the model’s behavior based on what subset of the input it sees.

The weights of that classifier tell us which parts of the input are important.
This secondary classifier’s **weights** now give us highlights on the input.

The movie is mediocre, maybe even bad.

- **The movie is mediocre, maybe even bad.**  
  - **Negative** 99.8%
- **The movie is mediocre, maybe even bad.**  
  - **Negative** 98.0%
- **The movie is mediocre, maybe even bad.**  
  - **Negative** 98.7%
- **The movie is mediocre, maybe even bad.**  
  - **Positive** 63.4%
- **The movie is mediocre, maybe even bad.**  
  - **Positive** 74.5%
- **The movie is mediocre, maybe even bad.**  
  - **Negative** 97.9%
Problems with LIME

- Lots of moving parts here: what perturbations to use? what model to train? etc.
- Expensive to call the model all these times
- Linear assumption about interactions may not be reliable
Gradient-based Methods
Problems with LIME

- Problem: fully removing pieces of the input may cause it to be very unnatural

LIME/erasure zeroes out certain features

Alternative approach: look at what this perturbation does locally right around the data point using gradients

data manifold (points we observe in practice)
Gradient-based Methods

score = weights * features
(or an NN, or whatever)

Learning a model

Compute derivative of score with respect to weights: how can changing weights improve score of correct class?

Gradient-based Explanations

Compute derivative of score with respect to features: how can changing features improve score of correct class?
Gradient-based Methods

- Originally used for images

\[ S_c = \text{score of class } c \]

\[ I_0 = \text{current image} \]

\[ w = \frac{\partial S_c}{\partial I} \bigg|_{I_0} \]

- Higher gradient magnitude = small change in pixels leads to large change in prediction

Simonyan et al. (2013)
Gradient-based Methods

Simonyan et al. (2013)
Integrated Gradients

- Suppose you have prediction = A OR B for features A and B. Changing either feature doesn’t change the prediction, but changing both would. Gradient-based method says neither is important

- Integrated gradients: compute gradients along a path from the origin to the current data point, aggregate these to learn feature importance

- Intermediate points can reveal new info about features

Sundararajan et al. (2017)
Evaluating Explanations
Faithfulness vs. Plausibility

- Suppose our model is a bag-of-words model with the following:

  the = -1, movie = -1, good = +3, bad = 0

  the movie was good prediction score=+1
  the movie was bad prediction score=-2

- Suppose explanation returned by LIME is:

  the movie was **good**
  the movie was **bad**

- Is this a "correct" explanation?
Faithfulness vs. Plausibility

- **Plausible** explanation: matches what a human would do
  - the movie was **good**  the movie was **bad**
  - Maybe useful to explain a task to a human, but it’s not what the model is really doing!

- **Faithful** explanation: actually reflects the behavior of the model
  - the movie was **good**  the movie was **bad**
  - We usually prefer faithful explanations; non-faithful explanations are actually deceiving us about what our models are doing!

- Rudin: *Stop Explaining Black Box Models for High-Stakes Decisions and Use Interpretable Models Instead*
Evaluating Explanations

- Nguyen (2018): delete words from the input and see how quickly the model flips its prediction?
  - Downside: not a “real” use case

- Hase and Bansal (2020): counterfactual simulatability: user should be able to predict what the model would do in another situation
  - Hard to evaluate
Evaluating Explanations

- Human is trying to label the sentiment. The AI provides its prediction to try to help. Does the human-AI team beat human/AI on their own?
- AI provides both an explanation for its prediction (blue) and also a possible counterargument (red)
- Do these explanations help the human? Slightly, but AI is still better
- Few positive results on “human-AI teaming” with explanations Bansal et al. (2020)
What to Expect from Explanations?

‣ What do we really want from explanations?
  ‣ Explanations should describe model behavior with respect to counterfactuals (Miller, 2019; Jacovi and Goldberg, 2021)

The movie is not that bad.

The movie is not ___ ___.

‣ What about realistic counterfactuals? Since dropping tokens isn’t always meaningful

The movie is not actually bad.

‣ We are going to evaluate explanations based on whether they can tell us useful things about model behavior

Ye et al. (2021)
A Multi-hop QA Example

We formulate a hypothesis about the model’s behavior, and test it using counterfactuals.

Base Example

Are Super High Me and All in This Tea both documentaries?

Super High Me is a 2008 documentary film about smoking.
All in This Tea is a 2007 documentary film.

YES

Token-Level Explanation

<s> Are Super High Me and All in This Tea both documentaries? </s>

<</s> Super High Me is a 2008 documentary film about smoking. All in This Tea is a 2007 documentary film. </</s>

Expected Behavior

The hypothesis is true.

Hypothesis

The QA model is looking at the two documentary tokens.

Realistic Counterfactuals

Super High Me is a 2008 romance film about smoking.
All in This Tea is a 2007 documentary film.

YES

Super High Me is a 2008 documentary film about smoking.
All in This Tea is a 2007 romance film.

YES

Super High Me is a 2008 romance film about smoking.
All in This Tea is a 2007 romance film.

YES

Actual Behavior

The hypothesis is not true. Model always predict YES.

Mismatch
Other work has done similar studies with humans interpreting model explanations to make predictions:

- People misinterpret these maps and conflate them with other factors. We actually need to *modify* what is shown to users to get them to have the right interpretation.

Schuff et al. (2022)

Human Interpretation of Saliency-based Explanation Over Text
Takeaways

- Lots of ongoing research:
  - How do we interpret explanations?
  - How do users interpret our explanations?
  - How should automated systems make use of explanations?

- Emerging consensus: there is no one-size-fits-all solution. There are many formats of explanation that all have their uses — choice may be application specific

- This research has taken a bit of a back seat during the current era of LLMs.
Packages

- AllenNLP Interpret: https://allennlp.org/interpret
- Captum (Facebook): https://captum.ai/
- Various pros and cons to the different frameworks
Takeaways

‣ Many other ways to do explanation:
  ‣ Probing tasks: do vectors capture information about part-of-speech tags?
  ‣ Diagnostic test sets (“unit tests” for models)
  ‣ Building models that are explicitly interpretable (decision trees)

Wallace, Gardner, Singh
Interpretability Tutorial at EMNLP 2020