Model inspection

INFO 4940

Interpreting nonlinear models

- Direct examination of feature weights is generally not useful for neural models
- To test input importance: **Permutation, ablation, perturbation**
- To examine the model itself: **linear probes**

Permutation, ablation, perturbation

The intuition is that models should respond to changes to their inputs in ways that make sense to an informed observer

- Remove info about an *important* feature -> model performance should go down meaningfully, consistently
- Remove info about an *unimportant* feature -> model performance is unaffected
- *Small* changes to input -> small, consistent changes in output

Linear probes

Linear probing is a model inspection technique. It has been a key component of BERTology and descendants.

- Extract all or part of a **hidden** layer from a model
- Use the hidden weights as feature representations
- Use representations to learn a supervised task with a simpler model
- If supervised model performs well, then the hidden layer contained a representation of the knowledge required for the task

Linear probes

Caveats

- The supervised model stores info about the task
    - The bigger the probe model, the more it stores, hence the less you can say about the hidden representations of the model
    - Compare probe performance to random baseline
    - Large probes have low selectivity
    - In general, use simple probes (Hewitt and Liang, 2019)
- Just because a model represents info doesn’t mean it *uses* that info to perform a task
    - Probes do not establish causation

LLM attribution

Explaining what drives the outputs of LLMs is hard, but not impossible. All the principles above apply.

Captum is a package that simplifies inspection and feature attribution in neural models and LLMs

See especially their LLM attribution tutorial