Large Language Models

Announcements

Large Language Models

Language Model

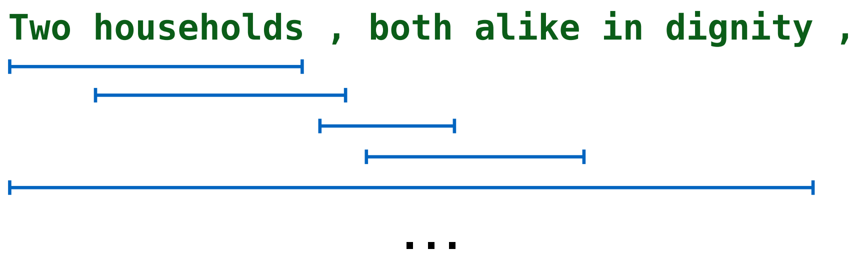
Language models (including large language models like ChatGPT, Gemini, Claude, etc.) are real-valued functions f(**context**, **word**) that return a real number (often from 0.0 to 1.0).

- Historically, the **context** was some incomplete text, and the **word** was a possible way to continue that text.
    - "Oski the Bear (Oski) is the official..."
    - "Oski was suspended for two weeks in January 1990, for throwing a cake towards..."
- Recent language models have expanded the notion of context to include other data (e.g., images) as well as multiple text sources (e.g., system prompts).

(Demo)

Language Models Before Neural Networks

The most effective language models prior to the rise of neural networks were based on counts of sequences of words in text, called n-grams.



Two households , both alike in dignity ,

...

The simplest score for a **context** and a **word** is just the count of **(context[-1], word).**

(Demo)

Generating Text Using a Language Model

One simple way to generate text from a language model give a context is to:

- Pick the word with the highest score for that context
- Add that word to the context

(Demo)

Sampling

Sampling from a probability distribution over words means selecting each word in proportion to its probability.

(Demo)

Neural Language Models (No Math Version)

A neural network defines a function from some large structured input (such as a context) to a fixed-size collection of numbers (such as a score for each word) based on parameters.

The parameters can be trained so that the function it defines is close to any relationship we want between the input context and the output scores.

Given a neural network, the first step in training is to make it able to recognize the text in a large collection (which always tends to include wikipedia):

- For the context "Oski the Bear (Oski) is the official",
  the model should score "mascot" near 1.
- For the context "Oski was suspended for two weeks in January 1990, for throwing a",
  the model should score "cake" near 1.

The amazing part: the people who build these systems don't choose how the text context is represented. It's up to the training process to invent a representation of the context that scores words correctly.

An important property of neural networks is that they capture similarity: two contexts are similar if they are followed by similar words.

Instruction Fine-Tuning & Human Feedback

The language model that results from matching a large text corpus is not a good chat bot.

Training it on a collection of good prompt−response pairs is helpful.

A critical later step in training a neural language model is to incorporate human ratings of its responses (alignment; reinforcement learning from human feedback).

Sketch of how this can be done:

- Generate two responses for the same context using sampling.
- Have a human rate which one they prefer.
- Train the model parameters so that, given a context, the words scored highly tend to be the ones that humans prefer.

Large Language Models and Code

(Demo)