#LLM

7

Large Language Model Parameters

datasciencedojo
data science for everyone

#LLM

Getting the best output from *Language AI*

LLMs are characterized by several key parameters that influence their functionality and output. The Large Language Model predicts the output (words) on the basis of an input (the prompt). Out of all the possible words, the model only shows the final result based on the set parameters.

datasciencedojo
data science for everyone

1 Model *Size*

The model size refers to the number of parameters in the LLM. A parameter is a variable that is learned by the LLM during training. The model size is typically measured in billions or trillions of parameters. A larger model size will typically result in better performance, but it will also require more computing resources to train and run. Different models have varying sizes and are suitable for different tasks.

For example, GPT-3 is a large model with 175 billion parameters, making it highly capable in various natural language understanding and generation tasks

2 Context *Window*

The context window determines how far back in the text the model looks when generating responses. A longer context window enhances coherence in conversation, crucial for chatbots.

For example, when generating a story, a context window of 1024 tokens can ensure consistency and context preservation

3 Number of *Tokens*

The number of tokens refers to the size of the vocabulary that the LLM is trained on. A token is a unit of text, such as a word, a punctuation mark, or a number. The number of tokens in a vocabulary can vary greatly, from a few thousand to several million. A larger vocabulary allows the LLM to generate more creative and accurate text, but it also requires more computing resources to train and run.

For instance, GPT-2 has a vocabulary size of 1.5 billion tokens. Larger vocabularies allow the model to comprehend a wider range of words and phrases

4 Temperature

Temperature controls text randomness. Higher values, like 1.0, result in diverse outputs, while lower values, like 0.2, produce more focused and deterministic responses. This parameter is useful when generating creative or precise content.

For example, if you set the temperature to 1.0, the LLM will always generate the most likely next word. However, if you set the temperature to 2.0, the LLM will be more likely to generate less likely next words, which could result in more creative text.

#LLM

5 Top-k and *Top-p*

These techniques filter token selection. Top-k selects the top-k most likely tokens, ensuring high-quality output. Top-p, on the other hand, sets a cumulative probability threshold, retaining tokens with a total probability above it.

For example, if you set Top-k to 10, the LLM will only consider the 10 most probable next words. This will result in more fluent text, but it will also reduce the diversity of the text. If you set Top-p to 0.9, the LLM will only generate words that have a probability of at least 0.9. This will result in more diverse text, but it could also result in less fluent text.

#LLM

6 Stop *Sequences*

LLMs can be programmed to avoid generating specific sequences, such as profanity or sensitive information. This can be useful for preventing the LLM from generating spam, offensive, or irrelevant text.

For example, you could add the stop sequence "spam" to the LLM, so that it would never generate the word "spam".

7 Frequency and *Presence Penalties*

Frequency Penalty penalizes the LLM for generating words that are frequently used. A presence penalty discourages the use of specific tokens, while a frequency penalty encourages token use. For instance, in language translation, a frequency penalty can be applied to ensure that rare words are used more often.

This can be useful for preventing the LLM from generating irrelevant text.

#LargeLanguageModels

Like this Post? 👍

Check out our blogs for more interesting information

datasciencedojo

data science for everyone