Non-deterministic nature of llms

Even if you set Temperature as Zero, it is still possible to get different answers because you are using GPUs.
In this case, the model will choose the token with the highest probability. However, when there are multiple words competing probabilities associated super close to each other, due to how GPUs calculate via floating points, it can be a coin toss.

Why LLM's responses vary from time to time?

(1) In addition to the inherent non-deterministic underlying architecture built in the core ethos of modern AI work.

(2) At the end, models generate next tokens. We see the output as text but inside it is generating a string of numbers and probabilities from the tokens distribution it can pick. For example, GPT-4 has 100,000 tokens it can choose from, so each result is the 100,000 list of model's probabilities to each possible token, selecting the top-k inherent in the configuration metric used.

Then, these probabilities are passed to a SoftMax (https://www.baeldung.com/wp content/uploads/sites/4/2023/05/softmax_animation.gif) where Temperature is applied, which smooths out the probability distribution, decreasing the likelihood of the most probable token choices and increasing the likelihood of the less likely tokens. After SoftMax and Temperature is applied, one token is picked in accordance with the metrics and that is how you get the token the user sees. (Note: So, with too low temperature the results are more predictable in responses every time but setting it too high you increase the possibilities of receiving nonsense sentences or causing coherent but incorrect responses.)

Quotes

"LLM non-deterministic nature is more of a feature than a bug"

Best Quote

"If you are asking for deterministic results on an LLM you basically do not understand the basics of an LLM. But the good news is you can lower the chances of different answers by setting some parameters, however it still does not be deterministic"