Repetitions | LangSmith Evaluation - Part 23

5 MIN · YOUTUBE · HTTPS://WWW.YOUTUBE.COM/WATCH?V=PVZ24JDZZF8
https://www.youtube.com/watch?v=Pvz24JdzzF8

SUMMARY

The discussion focuses on the importance of using repetitions in evaluations to enhance the reliability of results when working with language models (LMs). By running multiple iterations of evaluations, users can account for variability in outputs and grading, leading to more consistent and trustworthy results.

- *Repetitions help address the non-deterministic nature of LMs, which can produce variable outputs.*
- *The new feature in lsmith allows users to specify the number of repetitions easily through the SDK and UI.*
- *Evaluations can be run multiple times on the same dataset, providing a mean score that smooths out variability.*
- *Users can compare results across different configurations and see how grading may differ between repetitions.*
- *The feature is particularly beneficial for complex evaluation sets, where variability in grading is more pronounced.*
- *Running multiple repetitions increases confidence in the evaluation results and facilitates better comparisons across experiments.*
- *The ability to visualize each repetition's results aids in investigating output variability and grading differences.*

hey this is L from L chain we're Contin our lsmith evaluation series talking about repetitions so the intuition here is actually pretty straightforward we've talked a lot about different types of evaluations for example that run on like larger eval sets that have different and maybe complex LM as judge evaluators and in a lot of these cases we run an evaluation we get some statistics or some metrics on our performance across the data set and you might as the question how reliable is

this you know can I trust this result if I run it again can I reproduce it and you can have noise introduced from different things your chain May produce kind of variable outputs depending on how you run it again LMS are largely for the most part non- deterministic you know your LM is judged evaluator again it's using an LM so there's some variability that can be introduced from the grading itself so in any case the idea of repetitions is a

way to address this automatically run your evaluation end times to see whether or not it's consistent it's very intuitive it's very useful and I've done this manually many times but lsmith is actually introducing a nice kind of new flag that's run simply with the SDK where you can specify some number of iterations to run and it's all nicely supported in the UI so let's take an example case this is an eval set I've already worked with related to Lang

expression language this is an evaluator that actually used previously with a rag chain that operates on Lang chain expression language documentation and so this evaluator is basically going to grade an answer from a rag chain relative to the ground truth answer between 1 and 10 okay so that's all it's happening here and this is my

rag bot which I'm just initializing with a few different parameters I'm going to run it with GPD 40 with Vector store and I'm going to

run with gbd4 turbo without Vector store so these are just two example experiments I might run and here's where it gets interesting when I set up my evaluate function just as we've done previously I can just specify number repetitions and specify how many times I want to run this experiment so in this particular case my evil set has 20 questions it's run three times and so I run this it actually runs 60 different evaluations that's it and again I can run it on

on different configurations just like I've done previously so if I go over to the UI here's my here's my data set just like we've seen before here's my experiments and you're going to see something new and kind of interesting here you're going to see this repetitions flag noted here so what's cool about this this allows you then if you open up any of your experiments right so let's for example look at this experiment GPD for Turbo you can see if you open

up for any example this is your input right here's your rag chain you actually can see each each repetition run and so what's nice about this is that that you can compare the the answer for each your repetitions so that's kind of what you see here and you can look at the grading so you can see there's there's interesting differences depending on kind of the repetition in the answer itself which can happen because certain llm chains do have some

variability right so the answers can differ by the chain and also the grer given the same output can sometimes change right so what's nice about this is I can kind of go through my data set and I can actually investigate cases of variability in the output you know in this particular case you know one is great one in one case repetition three is greater as one8 7 right so what's nice about this these scores reported here are the mean

of those three repetitions so what's nice is these perform some smoothing across V variability that's inherent potentially in your chain itself or in your LM as judge evaluator it's a nice way to build bit confidence in your results and in this particular case this is working on with a larger more complex eval set so in this case it was a 20 question eal set you can look at the examples here these are kind of harder questions so I do expect that the

various experiments are going to have more trouble with them and I'm using an llm as judge evaluator in this case with custom criteria we're I'm grating from 0 to 10 so again that's like a more tricky grading scheme there's more opportunity for variability there and I can see in my experiments that indeed if you kind of dig in we showed some examples previously but there is some variability across my grading you know grade of one here versus 0.

5.5 and you know the ability to run repetitions gives me a little bit more confidence in the result so it also makes it easier to compare with some confidence across different experiments when you've used repetitions to smooth out noise in your greater or in your chain itself and really that's kind of the intuition behind using repetitions it's very intuitive you can see I've run a number of these different experiments with three repetitions each and this

is

kind of the aggregate of those means for each example being reported so I have a little bit more confidence in the difference in the difference between my various chains looking across these experiments relative to looking at a single trial or single experiment that only ran out a single repetition so really that's just that's all there is to it's really simple and it's a very nice feature I've used it extensively just kind of manually but actually having as now as a feature in lsmith

makes it much easier to run experiments with repetitions to build more confidence in your results thanks