

Catch Production Failures Early with Alerts in LangSmith

6 MIN · YOUTUBE · [HTTPS://WWW.YOUTUBE.COM/WATCH?V=RSF3H0BAGJI](https://www.youtube.com/watch?v=RSf3h0bAGjI)
<https://www.youtube.com/watch?v=Rs3h0bAGjI>

SUMMARY

Tinri, a product manager at Langchain, introduces a new alerting feature for Langmith, designed to help users proactively monitor application metrics such as latency and feedback scores. The video demonstrates how to set up alerts, configure thresholds, and integrate notifications with platforms like Slack.

- *New alerting feature in Langmith allows proactive monitoring of application metrics.*
- *Users can set alerts for application latency, error rates, and feedback scores.*
- *Alerts can be configured with customizable thresholds and time ranges.*
- *Integration options for notifications include PagerDuty and webhooks for Slack.*
- *The setup process includes creating a Slack app and configuring permissions.*
- *Users can also set alerts based on online evaluations of LLM response quality.*
- *Documentation is available for setting up tracing and alerts.*
- *Feedback from users is encouraged to improve the feature.*

Hey everyone, my name is Tinri and I'm a product manager at Langchain. Today I'm excited to introduce a new feature that we're launching for Langmith, which is alerting. In this video, we'll walk you through creating your first alert and also show you the different configuration options that we support for configuring alerts in Linksmith. So, we have this application called chat link chain. Chat langchain is an agent that you can use to ask questions about langchain, langraph or lang. Our customers use this pretty heavily, so it's important to make sure it's fast, reliable, and has quality answers. We have a dashboard set up for chat link chain that monitors some important metrics. One of which is application latency. You can actually see in the last week or so, we had a spike in application latency that we've since fixed. But the only way to learn about this would be to go to a dashboard and to view the stats. This is where alerts come in. Alerts allow you to proactively monitor metrics that are important for your application. So, let's set an alert up for application latency so that the next time this happens, we're notified in real time and we're able to come in and fix this immediately.

In order to set up an alert, navigate to your tracing project and then go to the alerts tab. You can see we have a couple of alerts configured already for error rates, but I want to make a new alert for latency. So today we support alerting on three different kinds of metrics. Erid runs, feedback score, as well as application latency. In the future, we do plan on adding more types of alerts here. So, I'm going to select latency for this purpose. And we're going to go with the default filter because we're concerned with end to end application latency. So, I'm actually going to pick a 15minute threshold here instead of 5 minutes because I know that some small fluctuations in app latency are okay.

And I'm going to change this threshold here to the last 14 days so that we're able to have a bird's eye view on what latency has look like when configuring the alert. And so now when you're actually setting the threshold, you can play around with the value that makes the most sense for your application. Let's try starting out at 40 seconds here. I think this is going to be a little bit too noisy and so I might bump this up to 50.

and then the nice thing here is you can actually zoom in on different time frames. So, let's go down to 7 days to get a better indication of when this alert would get triggered. Okay, I think this this I'm happy with. Allow us to minimize the number of false positives while still alerting when we see interesting spikes. And then to set the notification, we support two options today. You can configure a notification with pager duty or send an alert to a web hook. In this case, I'd like to set up alerts for our team Slack channel.

So, I'm going to pick the web hook option. And in our documentation, we have an overview of how you can send alerts to Slack using a web hook. There are a few steps involved before you get to the Langsmith configuration. So, you need to create a Slack app, configure permissions, install the app to your workspace, and then navigate to LinkSmith to actually configure the alert. I've already done all of these steps, so I'm going to go ahead and configure my alert now. And in our documentation, we also provide a template for a an example request body that you might want for your Slack alert. So, I'm going to be working off of that. So you can see I've added the name of my alert here. I've added some text about which alert has been triggered because I have multiple alerts that are set up. I've also included the link to the dashboard that I showed earlier for chat link chain. And and now I'm going to send a test notification. And let's confirm that that was successful. I can also pull up this channel in my Slack and let's confirm that we got the notification. And here we have the test notification for the alert that we just configured. So that's great. and the last step here is I'm actually going to hit create alert. So the one other alert that I want to show off is on feedback scores to measure the quality of an LLM application. It's not just about latency or error rates, but it's actually about the quality of the output of the agent or LLM call.

For these reasons, you might be using evaluations and specifically you can configure online evaluations that allow you to measure various aspects of an application's responses. So for example, here we have this LLM is a judge evaluator for helpfulness. This measures whether the response that the agent provided actually answered the user's question. And so one neat thing with alerting in Linksmith is you can actually set alerts based off of the results of these evaluations. So let's go through setting that up. I'm going to set my alert name to helpfulness. And in this case, we're going to pick this feedback score metric. I'm going to select the same evaluator that I had just shown.

And let's increase the time range a little bit. We can see that there are time periods where our online LLM as a judge evaluator had outputted scores that were much lower than what we would want in terms of quality. So let's set a threshold for this. We're going to have to make this less than or equal to. And in this case, I'd like to set the threshold around 8. And if our quality ever dips below that, I want to get notified so that we can dig into what the issue might be. Similar to the last alert, I'm going to create a web hook notification here.

And I'm going to go ahead and create this alert. That wraps up alerting. Alerting is a really powerful feature that

you should be using if you're already sending your production traces to Langmith. If you don't have tracing set up yet, I'll include some links in the description below on how to get started with that, as well as links to documentation so that you can get started with alerts today. If you have any feedback, please feel free to reach out over Slack. We'd love to hear what you think.