LLM SURVEY REPORT

MLOPS Community

INDEX

MLOps Community | Am Sonnenberg 49, Heidenrod 65321, Germany

THANK YOU TO OUR SUPPORTERS

This report was written by Demetrios Brinkmann for the **MLOps Community**

The report would not have been possible without all the help from various community members including **Katie O'Leary, Shreya Shankar Médéric Hurier, Rajko Radovanovic, Rohit Agarwal, Diego Oppenheimer,Berndt Lindner, Jeremy McMinis** and **Lawrence Hechtofficil**

LLM SURVEY RESULTS

INTRODUCTION

Since the release of ChatGPT six months ago, we've seen a Cambrian explosion of AI use cases once thought to be out of reach. Research and development is moving at a lightning pace now, with new iterations of ideas, applications and models almost daily. Developers have raced to embrace the shift, building new kinds of apps on top of Large Language Models (LLMs) such as GPT-4, Bloom, LLaMA, Falcon, Starcoder and Gorilla, along with other state-of-the-art foundation models like SAM (Segment Anything Model), Stable Diffusion, Gen1 and Gen 2.

Foundation models are no small feat to create, often costing millions, to tens of millions of dollars to train. LLMs in particular also present major challenges when it comes to serving inference with them, often taking multiple datacenter level GPUs and backend tricks like weight streaming to make them work. It can take hundreds or thousands of the most advanced GPUs running for months to train these models and as many to serve them too. Some analysts have noted that many of the top models lose money on inference as the world waits for economies of scale to kick in and drive down prices.

Companies behind the AI models have seen a race to "outdo" the latest model in parameter size, with the most well known and powerful model (GPT-4) rumored at over a trillion parameters.

The MLOps Community has always worked hard to stay at the forefront of data science, providing a place to share and learn together. But in the last few months we've seen an explosion of interest beyond the data science and data engineering community. Everyone from traditional programmers, to business people and pundits have taken a major interest in AI and for good reason. We decided it was high time to dive in and see what we could learn about the current state of the art and how organizations are now using LLMs.

We polled the MLOps Community to get a deeper understanding of who, how and where LLMs are being used in applications. The community responded, showing us a wide range of use cases and many challenges with running these systems in production. We asked many of the questions in an open ended way, and because of that much of the data is unstructured and thus the insights are multifaceted but also a bit subjective. We've broken it down in great detail in the pages that follow but for the raw data please go here

Lets dive in and have a look at the rapidly changing world of LLMs and foundation models now.

WHO PARTICIPATED?

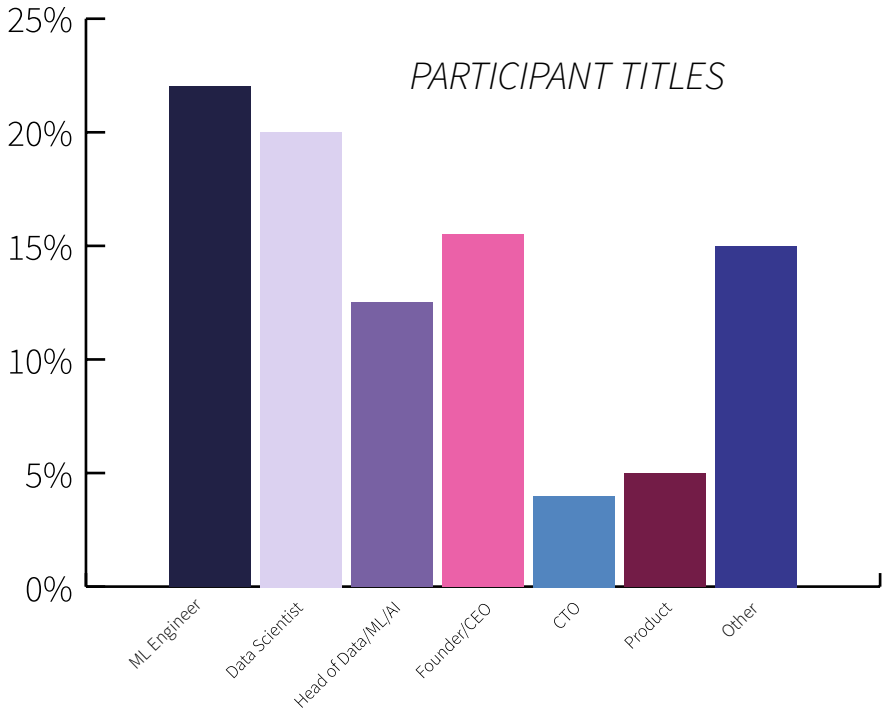
The survey saw a large number of the usual suspects. There was a varying degree of seniority, with the data skewing towards more senior-level participants.

We created this survey in combination with our roundtable session on using LLMs in production. At the event, we asked attendees to fill it out. We also had a few of our influencer friends tweet out a link to the survey to garner more traction. Overall, 123 participants gave candid feedback about using LLMs, what their use case is, what the main challenges are, and what questions they have. Responses were collected from March 11th until April 20th.

After bucketing the different responses the main titles were:

- ML engineer/MLOps Engineer
- Data scientist
- Head of data/ML/MLOps
- Founder/CEO
- Product
- Other - cloud engineer, marketing manager, analytics engineer, consultant.

Of these titles, over half are senior or hold executive positions.



SURVEY AT A GLANCE

The main takeaway from the 2023 LLMs in Production survey is that while companies are eager to use LLMs in production, challenges in cost, latency, output variability, and other infrastructure sophistication efforts still hinder teams from extracting the most value from their AI investments. As a result, we will likely see a boom in the number of LLM infrastructure companies providing services to overcome these obstacles in the near term.

In this report, we focus on seven key survey findings and what they say about the LLM in Production landscape. Those key findings are as follows:

01 Common use cases like **text generation and summarization** are useful, but **participants are going deeper** and exploring a long tail of ways to use LLM for tasks such as **data enrichment, data labeling augmentation, synthetic data creation, and question generation for subject matter experts.**

02 The use of Large Language Models within an organization is **still unclear due to the high costs and unknown ROI.** (Cost and ROI were cited 38 times in the survey.)

03 **Hallucinations** present real **ethical** and **business concerns**, so much so that for some it is a **show stopper**. 24.6%, or 28, respondents raised the issue of hallucinations and output reliability at some point in the survey, making it the **second most referenced concern** when dealing with LLMs.

04 Challenges around **LLMs speed of inference** make certain use cases such as fraud detection or recommender systems impossible. 10.5% of respondents highlighted the latency of models being an issue.

05 Current **infrastructure** is not designed for such massive models. 53% of respondents reported using Open AI's API, while 29.6% cite the use of an open-source model, 9.3% use in-house models, and 7.4% use some other model provider's API. Six respondents raised the question of whether it was better to be using a smaller, more task-specific model vs. calling an API.

06 Larger questions, such as **which part of the current workflows will be replaced vs. augmented**, are still unclear. 14% mentioned a lack of clarity around how to best leverage LLMs.

07 Prompting is a valuable skill. Even the **best prompts cannot force the model to give consistently correct output.** 6% of respondents cited the obscurity around the probabilistic nature of LLMs.

LIMITATIONS OF THE ANALYSIS

Before we begin, we must state this disclaimer. As was said by George Box, “All models are wrong, but some are useful.”

This sample may not be representative of the entire market. Future studies may wish to ensure to recruit participants that do not speak English, more non-engineers, and more females. This approach could broaden the geographic footprint of the study, which was mostly in the USA, Europe, and to a lesser extent India.

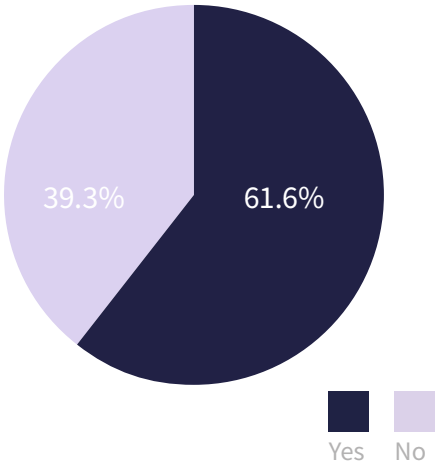
The responses were collected in March 2023. The AI landscape experiences the passage of time faster than your family‘s Golden Retriever. That means this data (although it is only a few months old) may already be stale or have significantly changed.

Being that the responses were primarily received as unstructured text data, the key findings are interpreted through the lens of the writers and editors of this report. All takeaways from the data have been checked and cross-checked by various peer reviewers with a variety of backgrounds. This does not mean that our own subjective interpretations have not been added to the findings. To come to your own conclusions, please review the raw data.

Let us remember that this is but a small sample size of all parties currently using LLMs in production. We warn against reading too much into the smaller trends. We have laid them out for you in order to start a conversation. They should not be thought of as an accurate representation of everyone using LLMs.

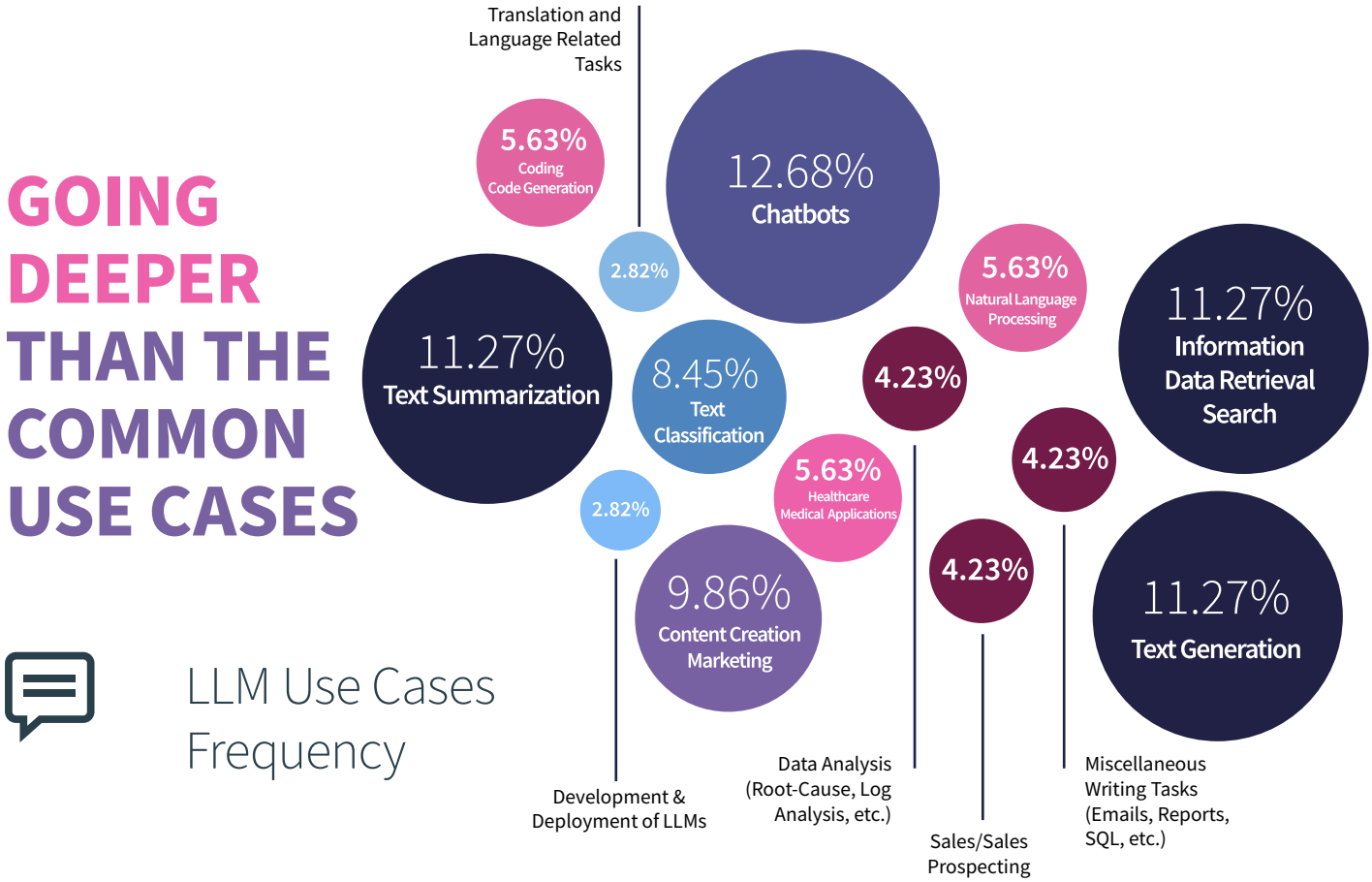
ARE YOU USING LLMS AT YOUR ORGANIZATION?

Using LLMs in your company?



Out of those surveyed, 61.6% were currently using LLMs for at least one use case. As we will see below, the most high-value and highest-leveraged use cases aren‘t super clear, nor is the ROI. That does not stop respondents from experimenting to find product market fit within their organization or as a customer-facing application.

GOING DEEPER THAN THE COMMON USE CASES



There are quickly emerging sets of use cases that lend themselves well to this type of technology (in its current state). Participants are also testing out new ideas to see how else this technology can be leveraged. The following list enumerates some of the main use cases we have identified from the survey:

- Niche Chatbots
- Customer Support
- Coding
- Data Enrichment
- Product Recommendations

Niche chatbots

Chatbots have been the bane of the internet since being introduced. They held so much promise and gave us a glimpse into what was possible. Yet just as with customer service robot recordings, chatbots have not lived up to expectations. That is, until now.

By leveraging the generality that LLMs bring, many new applications are able to actually get valuable information and understandings from off-the-shelf models incorporated into chatbot interfaces. There are some obvious caveats to this; otherwise, you are just adding a ChatGPT extension to your offering.

The most powerful use cases for chatbots are what we call “niche chatbots” – or, rather, specialized LLM chatbots on narrow domain topics. These chatbots are being used for both external- and internal-facing scenarios. The former is one that most people are typically accustomed to interacting with on a customer service page. The latter, we believe, is much more powerful.

Internal niche chatbots are showing promise to help workers quickly query diverse knowledge bases across the company. A predominant sign of this is that three respondents answered with the phrase „information retrieval“ while explaining their use case. While only mentioned twice, an outlier case full of potential is the use of internal-facing chatbots to help “power up” managers to quickly understand the context around projects, get the right data that they need, or answer relevant questions about the organization and project details.

Customer support

While this use case is not novel, it is quickly being proven as a strong one. Aside from the chatbots mentioned above, five participants cited using LLMs in this domain for use cases such as entity matching (determining if multiple records match the same real-world entity), analyzing customer service responses, and analyzing customer feedback forms.

Digging deeper into the analyzing customer feedback forms use case, LLMs are being used to categorize text sentiment and automate the writing of reports based on the feedback data.

Coding

Github Copilot X has been blazing the way for this type of use. However, respondents are doing more than just asking to create a function. LLMs are helping with analyzing logs, finding security vulnerabilities, and root cause analysis.

A larger theme that three respondents cited was using LLMs as a type of translation service. Some use cases for this are translating Excel reports to Python code, or natural language to SQL. Due to the effectiveness of LLMs in this area, it is freeing up developers' time so they can spend it on harder problems. It is also allowing devs to familiarize themselves with new programming languages more quickly. A whole survey and report can be done solely on the effects of LLMs on developers.

Data enrichment

As the famous pop culture personality Xzibit said, "Yo, dawg, I heard you like cars so we put a car in your car so you can drive while you drive." We are seeing this meta-use case for leveraging LLMs.

Four respondents cited using LLMs for synthetic data and image generation, data annotation, and zero-shot/few-shot classification to help make other ML models in a company's arsenal more robust.

Let's zoom in on synthetic data generation for a moment. This has huge implications for the ML community. It implies that the cost of data labeling has the potential to come down sharply. However, we have seen recent papers come out that highlight the limitations of training smaller LLMs on the output of larger LLMs

A unique use case that two respondents pointed out was around question generation. In these scenarios, LLMs were being called at various stages of a project's lifecycle to ask humans a question about the input data. These questions go beyond missing fields in a form. The models are analyzing all the data that had been collected and from this information ask if other possibilities had been thought about by the human before they proceed to the next task. This could be used by a lawyer who gives a defense argument or an accountant deciding what tax loopholes to take advantage of.

Product Recommendations

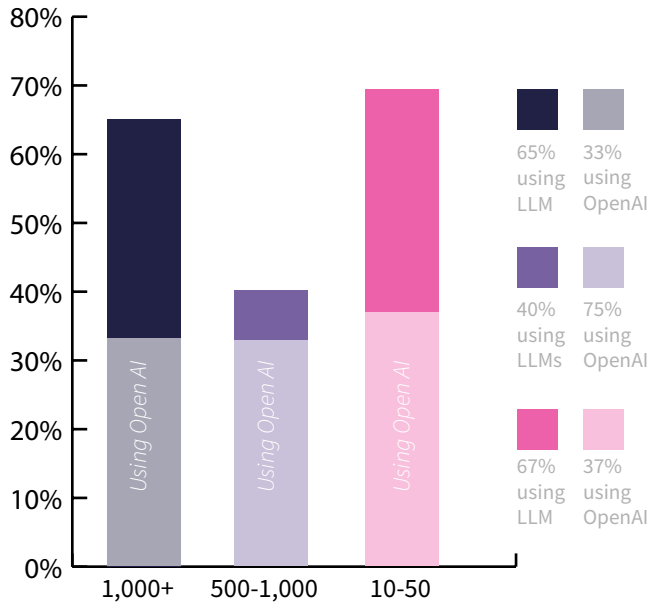
Does this change recommender systems as we know them? At a high level, neither the technological capability nor the latency efficiency is present for using LLMs in real-time recommender system use cases. However, respondents did highlight leveraging LLMs for their recommender system workflow in the following ways:

- To help with feature extraction (this being the most popular answer)
- To help create a semantic vector embedding features
- To help with feature recommendations

USING OpenAI

Making sense of the OpenAI usage data, one theory arises. If you're company is very small, you may feel AI is an existential threat. Therefor, you feel LLMs need to be a part of your IP, hence no OpenAI.

Open source vs OpenAI by Company Size



If you're a large organization, it's more likely someone somewhere is going to be using LLMs. Plus you may not trust OpenAI with your data and you have the resources to do your own thing. It is also more likely at this stage you hold very sensitive customer data.

However, if you're in the middle, you are trying to focus on your core product until you really get product market fit. This means you're really just dabbling in LLMs. Thus OpenAI is good enough and quite useful for fast iterations.

Yes we are building special tooling

Of the 68 LLM users that answered our question about tooling, 23 (36.5%) indicated their organization has built or integrated internal tooling to support LLMs. Specific tooling ranged from modular Python libraries to interact with various APIs to LangChain and vector database integrations.

Of the eleven respondents who were using LangChain 100% of them were using the OpenAI API. No one using LangChain was using open source models or any other model provider than OpenAI.

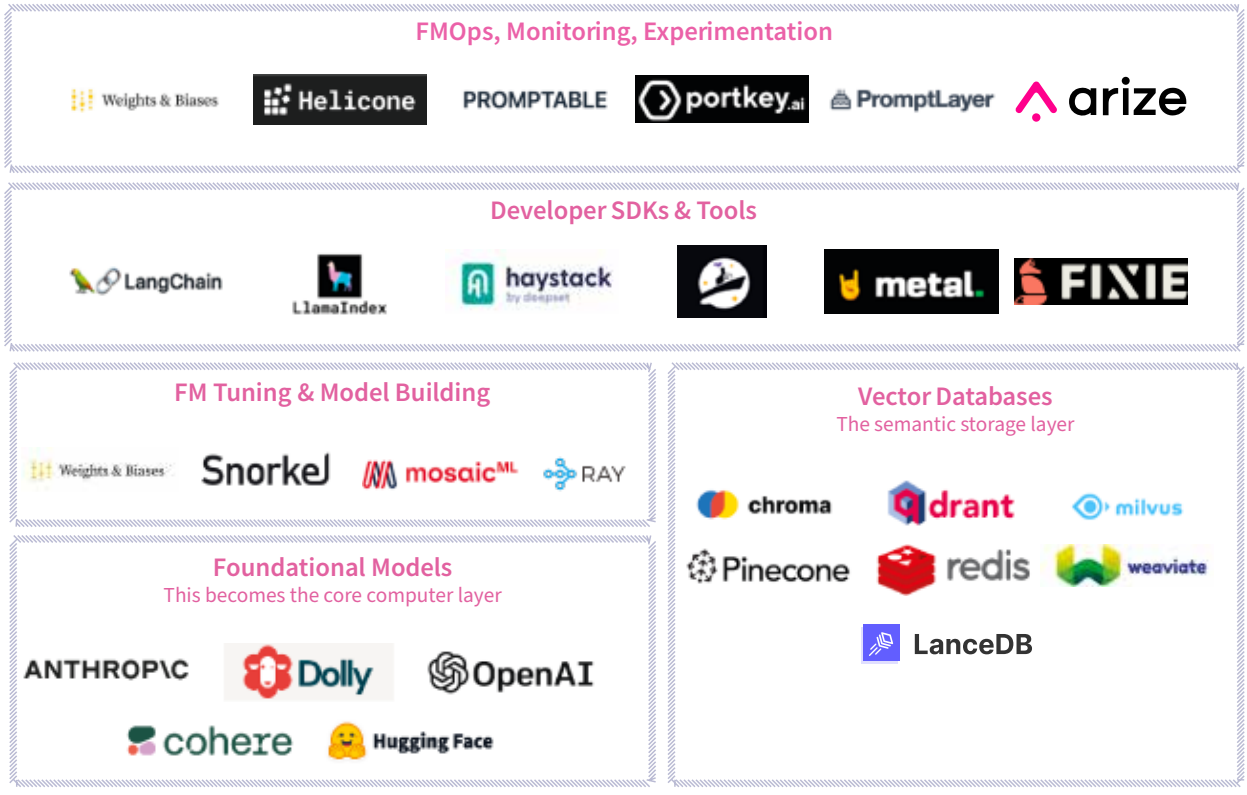
Have you integrated or built any internal tools to support LLMs in your org? If so what?

Just a bunch of shitty scripts I need to refactor in the future.

A few other examples of specialty tooling being created by respondents are around preprocessing of prompts, custom Python libraries, output validation, and high-scale serving on top of Kubernetes.

Below you can see an example of a Generative AI Stack

GENERATIVE AI APP STACK



MAIN CHALLENGES AND CONCERNS

> Size in all regards. Also training time, response time and multi GPU debugging is weird

Respondents were quite vocal about the challenges they encountered productionizing LLMs in their organizations. We were able to bucket the data into five categories.

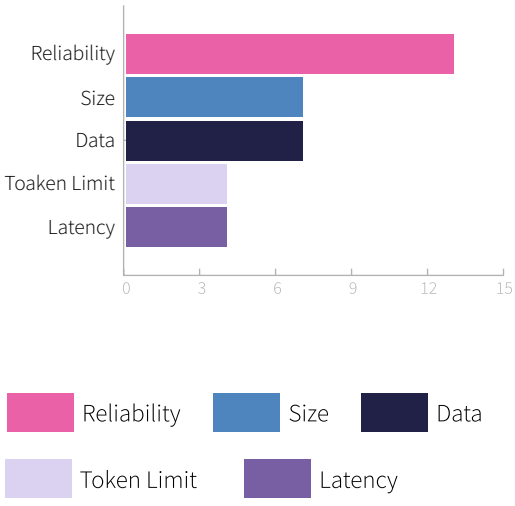
- **Output** - Hallucinations, explainability, consistency
- **Infrastructure** - size, reliability, latency, token limit, data
- **Prompting** - Prompt debugging, prompt engineering
- **Cost** - infrastructure costs, engineering costs
- **Uncertainty** - data privacy, unclear standards, compliance

Of the 58 responses, 40 people identified some type of infrastructure problem as being a main challenge. Within the infrastructure bucket, five main themes emerged.

- **Size** - Big models, compute power, training times, deployment, memory requirements
- **Reliability** - Tests, reproducibility, changing APIs, scalability
- **Latency** - Slow, reducing latency
- **Token Limit** - Handling large documents, input length limitation
- **Data** - Embeddings, fine-tuning, obtaining data, data labeling

Below you can see the distribution of these themes

Infrastructure Challenges



Does the high cost actually have an ROI?

The cost came up 41 times in the responses. It was undoubtedly something that both respondents who are and aren't using LLMs in production are thinking about. Costs can range from infrastructure, inference, and engineering all the way to organizational.

Production inference costs can be astronomical whether teams are using in-house models or making API calls. In-house models need expensive infrastructure requirements and babysitting while API pricing quickly gets out of hand at scale.

There is no clear consensus yet if the time spent implementing LLM-based features warrants the return on investment. SaaS companies building on top of LLMs may see margins shrink as they must now factor in these costs to their COGS. And does this new AI feature justify raising prices for the end user? If not, it is the company that must eat the expenses.

The meta question many were debating was, "Is it worth it?" We have all had the experience of spending time prompt-tuning something only to realize it would have been easier to complete the task without the LLM altogether. In the last six months, many product roadmaps have gone out the window as teams embark on a wild goose chase to incorporate LLMs into their offering. These side quests take their toll on a team's output and productivity, burning through organizational resources. Other questions that came up around ROI were how to gauge which parts of a process can be augmented by the new capabilities of LLMs and which parts can be completely replaced.

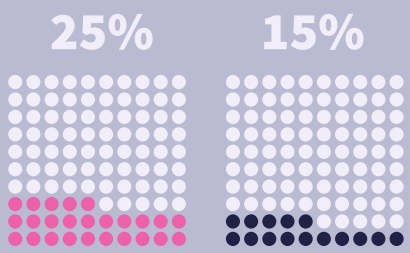
Lastly, as trends have suggested, some respondents questioned the use of smaller, more efficient models trained on relevant data as opposed to a larger generalized model.

Trust

Considering the college antics of some of the writers of this report, we must address the elephant in the room: hallucinations.

Trusting the LLM output was mentioned in some form by respondents 17 times throughout the survey. Hallucinations raise ethical concerns and make high-risk use cases a non-starter.

Getting the ground truth of the output can be tricky, and allowing the models to give any answer without proper validation can have detrimental effects even on harmless use cases (see Google stock price after the initial release of Bard). Not only did respondents cite hallucinations as putting the reputation of the company at risk, but they also mention how it creates a poor user experience.



25% of other model providers mention cost vs 15% of OpenAI users.

The data suggests that other model providers, be it open source models or 3rd party API's such as Anthropic or Cohere, are a cheaper option. The *cost conscious consumer* looks for other options to avoid paying the OpenAI premium.

It took us less than 3 minutes to get BioGPT to tell us vaccines cause autism

Not Using

Out of the 47 respondents who said they were not using LLMs in production, the most common response (34%) said there was no real use case for the business.

The long tail of responses that came up is worth noting. Respondents spoke of data privacy concerns (see Samsung employees using OpenAI). Within the data privacy theme, both data governance and data access were mentioned. Lack of provenance was also cited by respondents who spoke about not knowing what data the model was trained on as being detrimental.

Other responses included participants highlighting that the current tooling is not mature enough to make them feel confident, or that they had tried to use LLMs but ultimately gave up as it did not work for their use case.

Just as in music, it is not always about the notes you play but rather the notes you don't play. It is worth pointing out what respondents did not say. Not one response highlighted the lack of budget as a reason for not using LLMs. Respondents know that costs are high; from the survey data this is clear. However, being that projects are so new, budgets may not even be considered yet.

OPEN QUESTIONS AND OPPORTUNITIES

Is what I'm building going to be obsolete in the next three months? With the rapid pace of change on a technological level, respondents highlighted this as a fear they have in the back of their minds.

Over the past three months, this idea has become more pronounced. We can see the rise of vector databases vs. increased context length, GPU shortages, the distillation of models to run on CPUs, and a whole host of research papers that make it difficult for engineers architecting systems to have confidence in design decisions.

Below is the raw data of some of the most interesting questions and comments that respondents submitted.

What are some of the best practices for dealing with LLMs or foundation models?

I mainly wonder how everyone is monitoring these. ALSO. For my company, GDPR is a big deal. My DPO shoots down our ideas more often than not or limits them to using the last three months of data. How do other people deal with this?

Growing LLM features iteratively & coherently vs Alphabet's "put LLM in all the things!"

When does the ROI and cost trade-off happen for open ai API vs open source? if serving 5k users what does that look like vs serving 500k users?

Where are others drawing the line between training models locally and using LLMs? Have others had success with using LLMs to bootstrap training data for smaller local models?

Best practices to prevent leakage for client-facing models that employ online learning?

How are you thinking about tracking user feedback in production?

How are you managing the training/deployment of these?

How do you decide where to use LLM?

How do you decide which model to use and what is good enough?

How do you get traditional engineers to understand that they shouldn't underestimate the complexity of getting ANYTHING ML-related into prod?

Is anyone measuring LLM inference power consumption (and CO2 emissions)? What are the observed ranges of watts/inference of the various model sizes?

LOOKING FORWARD

Respondents highlighted four key areas they would like to learn more about in this space. One key to note here is that the majority would like to learn more about how to fine-tune LLMs. This suggests that OpenAI/LLM APIs for inference are not enough for many use cases.

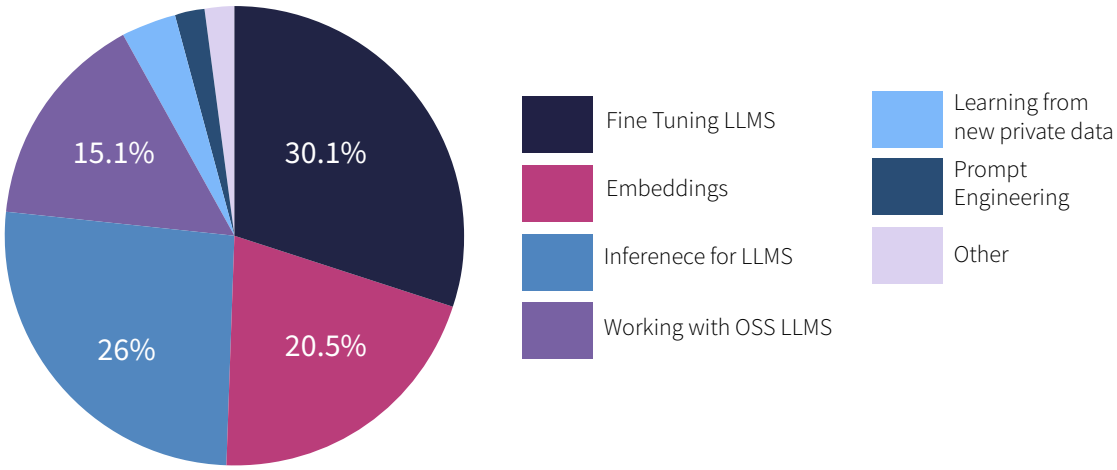
What areas are you most interested in learning about?

30.1%
Fine-tuning

20.5%
Embeddings.

26%
Inference

15.1%
Working with OSS models



In the MLOps Community, we have already started creating content around these topics. On June 15 and 16, we will be holding our next LLM in-production virtual event. Join over 70 speakers and have a whole lot of fun! Let's shape the future of this space together.

ABOUT OUR SPONSORS

Anyscale is the AI infrastructure company built by the creators of Ray, the world's fastest growing unified open-source framework for scalable computing. Organizations like OpenAI, Uber, and Spotify rely on technology from Anyscale to efficiently scale their most demanding ML workloads.

Databricks is the lakehouse company. More than 7,000 organizations worldwide — including Comcast, Condé Nast, H&M and over 50% of the Fortune 500 — rely on the Databricks Lakehouse Platform to unify their data, analytics and AI. Databricks is headquartered in San Francisco, with offices around the globe. Founded by the original creators of Apache Spark™, Delta Lake and MLflow, Databricks is on a mission to help data teams solve the world's toughest problems.

LlamaIndex is a data framework for building your LLM applications. It contains a comprehensive toolset allowing you to unlock the full capabilities of LLMs on top of your private data, including files, workplace apps, databases, and more. It offers an extensive array of integrations with other storage providers and downstream applications. The end result is that you can build a variety of amazing knowledge-intensive LLM applications.

Arize AI is a machine learning observability platform that helps ML teams deliver and maintain more successful AI in production. Arize's automated model monitoring and observability platform allows ML teams to quickly detect issues when they emerge, troubleshoot why they happened, and improve overall model performance across both structured

ABOUT OUR SPONSORS

Portkey.ai enables companies to launch generative AI apps with confidence by being the observability, model management, compliance and fine-tuning layer.

Deepgram helps developers and enterprises power their apps with world-class language AI models. We offer the world's most powerful speech-to-text API: 22% more accurate, 23x faster, 3-7x cheaper than nearest competitor.

Prem AI is a powerful AI platform designed to deploy self-hosted, Open source AI models, without exposing sensitive data to third-party. Prem provides a secure and flexible environment for self-hosting AI models quickly and efficiently.

ABOUT THE MLOPS COMMUNITY

There are many ways to interact with the community. Below are a few cool things we are doing.

- **Podcast**
- **Newsletter**
- **In-person Meetups**
- **Slack Workspace**
- **Virtual Events**
- **Blog**
- **Virtual Reading Group**
- **Founders Corner**

LLM SURVEY REPORT

MLOps Community

MLOps Community

Am Sonnenberg 49
Heidenrod 65321
GERMANY