neptune.ai

State of Foundation Model Training Report 2025

Table of contents

PUBLICATION DATE: 06/05/2025

Executive summary

- Companies are training foundation models to **solve highly specialized problems, meet compliance requirements, and build competency** in the technology.
- Foundation models can train on large quantities of diverse data, leading to a **shift from diligently handpicked datasets to acquiring and tapping new sources of raw data.** Synthetic data plays an increasingly important role.
- **Models and training datasets continue to grow larger,** generally leading to task performance improvements. However, **scaling up is not a silver bullet,** with small models and alternative architectures showing great potential.
- **Virtually all of today's foundation models are trained on GPUs.** Increasingly, **companies invest in on-premises training infrastructure,** trading flexibility for predictable architecture and availability.
- **Foundation model teams are multi-disciplinary** and have a **strong focus on data and software engineering**. Hiring for the breadth and depth of required skills is a principal concern of companies pursuing foundation model training.
- **Successful foundation model strategies** are characterized by early proof-of-concept projects, building deep expertise across all aspects of training, application-based performance evaluation, and maintaining focus on core objectives.

This report is based on in-depth interviews with senior leaders and technical decision-makers between October 2024 and January 2025. Participants represented organizations operating in financial services, healthcare, manufacturing, and research across North America, Europe, and Asia, ranging from startups to multinational enterprises and government agencies.

Spun out of the deepsense.ai team in 2018, after winning Kaggle's Right Whale Recognition competition, neptune.ai was created to tackle the growing complexity of managing ML experimentation. Today, Neptune's purpose-built experiment tracker for foundation model development is a trusted tool for teams at OpenAI, InstaDeep, Poolside, Bioptimus, Navier AI, Play AI, and more. Backed by $18M in funding and included in CB Insights' "Top 100 AI Startups" list in 2021 and 2022, Neptune helps teams monitor, evaluate, and scale model training with confidence. Learn more at www.neptune.ai.

Introduction

This report gives decision-makers a clear view of foundation model training today—covering the current state, emerging trends, and best practices. It cuts through the hype to show what teams are really doing, planning next, and aiming for long-term.

Companies that are considering a foundation model strategy face several key questions:

- ✓ *Can we use third-party APIs or self-hosted freely available models, or do we need to develop our own foundation models?*
- ✓ *How are other companies approaching foundation model training, and what determines their success?*
- ✓ *What are the resource requirements of foundation model training in terms of compute, data, and expertise?*

However, the publicly available information on these topics is limited. End-user and API products dominate the discourse, both in the media and in the business market, and insights into model training come mainly from marketing materials and academic papers.

Both kinds of publications highlight benchmark scores, the scale of data and models, and technical innovations with the aim of signalling a leading position to their audience. This is of little use to business stakeholders seeking to assess the value of a potential foundation model initiative.

This report aims to close this gap. In three main chapters, it gives decision makers an overview of the current state of foundation model training, provides an assessment of emerging trends, and summarizes best practices.

Throughout, we cut through the hype and look behind the curtain to reveal what companies are actually doing today, their near-term plans, and their long-term strategy.

Why we created this report?

Since 2017—the same year the first transformer model emerged—we've been building tools for training machine learning models. Our product, engineering, and customer-facing teams work closely with organizations around the world that are training foundation models, from academic researchers and startups to enterprise teams and industry research labs.

This close collaboration gives us a front-row seat to the latest developments, challenges, and breakthroughs in the field. We often identify trends and upcoming roadblocks before they are discussed outside of the leading AI labs, and see different strategies play out.

At the same time, we're neutral. We're not developing foundation models ourselves—we're focused entirely on building tools that help those who do. This is the approach we've taken with this report as well: Our goal is to present the current state of foundation model training and the trends we see in a way that allows those who are looking to train foundation models to make informed decisions.

Research methodology

Given the early stage and significant variation in foundation model development practices across organizations, we employed a qualitative research approach.

Quantitative metrics such as headcount and budget allocations would fail to capture the nuanced decision-making processes or strategic considerations characterizing today's foundation model initiatives. Instead, we conducted in-depth interviews that allowed us to explore complex technical and organizational questions in detail.

Between October 2024 and January 2025, we conducted 13 semi-structured interviews with senior leaders and technical decision-makers. Our participants represented organizations headquartered across North America, Europe, and Asia, providing a diverse international perspective. The organizational profile ranged from early-stage startups with minimal number of employees to established multinational enterprises. Additionally, we included a governmental agency in our sample. The participating companies operate across various domains including financial services, healthcare, manufacturing, and research.

13 semi-structured interviews

3 continents

Senior leaders and technical decision-makers

6 industry sectors

We analyzed the interview transcripts to identify recurring themes, shared challenges, and divergent approaches. We then combined these findings with our own first-hand observations and secondary research. Our interview protocol and anonymized participant information are documented in the appendix.

Some of the companies featured in the report

Genentech
DeepL
Inria
Blue Raven AI
Silurian AI
Synerise
Kaiko

How to navigate this report?

This report is organized into three main chapters, each addressing a key question.

1. **Current State of Foundation Model Training:** What approaches are companies currently implementing, and what strategic objectives drive these initiatives?

2. **Trends in Foundation Model Training:** Which patterns and developments are emerging consistently across teams, companies, and industries?

3. **Best Practices for Foundation Model Training:** What practical recommendations do successful teams offer based on their hard-won experience and lessons learned?

Recurring elements

Throughout the report, you will encounter these recurring elements:

Spotlight: Direct insights and quotes from industry experts we interviewed. These represent unfiltered, first-hand perspectives from trailblazers in foundation model

Neptune insight: Practical lessons and observations from our work with teams training foundation models.

Key term: Brief definitions and background information on important concepts mentioned in the main text. Each definition includes a link to an in-depth knowledge article.

A note on terminology

Terms like **“LLM”** (large language model), **“foundation model”**, **“transformer model”**, or simply **“AI”** are often used interchangeably.

In this report, we follow the definition given in the [2021 Stanford Report]() that popularized the term “foundation model”:

✓ **Foundation model:** A model “trained on broad data (generally using self-supervision at scale) that can be adapted to a wide range of downstream tasks.”

This adaptation occurs through:

✓ **Prompting (in-context learning):** The model’s behavior and capabilities are influenced through specific inputs, while its weights (parameters) remain unchanged.

✓ **Fine-tuning:** The model’s weights are modified through additional training on task-specific data, resulting in a new model artifact specialized for particular applications.

Today, transformer-based generative models dominate this category and are often treated as synonymous with “foundation model.” However, what makes these models “foundational” is not their architecture, but their applicability to a broad range of downstream tasks.

For the purpose of this report, we will further distinguish between:

✓ **Open models (open-weight models):** Models with downloadable weights, even if subject to restrictive licenses. The architecture and implementation are typically public, allowing others to evaluate, use, and adapt these models.

✓ **Closed models:** Models that are either unreleased or accessible only through APIs or hosted services. While the architecture may be described in publications, the implementation details, training process, and weights remain proprietary. Benchmarks and other evaluations cannot be independently verified.

For both open and closed models, training data is rarely fully disclosed beyond general descriptions (e.g., "billions of tokens of web-scraped data"), and complete training code or configurations are seldom made publicly available.

Current state of foundation model training

Chapter summary

- Companies are training foundation models to **solve highly-specialized problems, meet compliance requirements, and to build competency** in the technology.
- Common foundation model strategies are training **models for direct application**, creating **a main model to derive downstream models through fine-tuning, training task-specific small models, or fine-tuning open models**.
- Today, virtually all foundation models are **trained on GPUs**. Maintaining and efficiently utilizing this hardware platform is a major challenge for foundation model teams.
- Foundation models can train on large quantities of diverse data, leading to a **shift from diligently hand-picked datasets to acquiring and tapping new sources of raw data**. Synthetic data plays an increasingly important role.
- **Teams training foundation models are multi-disciplinary**. Hiring for the breadth and depth of required skills is a principal concern of companies pursuing foundation model training.

There is a significant number of companies currently training foundation models, with many more exploring this path (see Foundation Model Training Landscape: a list of more than 100 companies training from scratch). In this chapter, we'll answer key questions executives and stakeholders ask us:

Why are companies training foundation models?

What kinds of foundation models are companies training?

What hardware and data infrastructure is required?

Where does training data come from and what role does it play?

What does a successful foundation model team look like?

Spotlight

> *In 2022, we made the decision to invest in LLMs. This was pretty speculative at the time. We didn't necessarily expect products to come out of this, but we wanted to have the capability.*
>
> *Later that year, with the release of ChatGPT, the question was no longer whether it was going to work but how quickly we could get there. So we doubled down on our LLM efforts, and since late 2022, most of our products have run on top of LLMs rather than the much more complicated architectures we used prior.*
>
> **Stefan Mesken, Chief Scientist**
> **DeepL**

Why are companies training foundation models?

The companies we interviewed for this report arrived at the decision to invest in foundation models through a combination of technical, product, and strategic considerations. Apart from the companies that were founded with the explicit aim of developing foundation models, all companies were already actively working with machine learning prior.

In the following, we will discuss the driving factors in their decision-making process.

Foundation model training landscape

A list of more than 100 companies training from scratch.

See the landscape

Solving highly-specialized problems

Regardless of their size, industry, or location, the companies we interviewed for this report have one thing in common. They work in a specific domain, cater to a niche in their market, or provide solutions for a particular task. While this is hardly unusual for companies, it sets them apart from the companies at the center of the current public discourse around foundation model training. Industry high-flyers like Anthropic, Mistral, OpenAI, and established tech giants like Google and Meta, develop foundation models for extremely generic tasks.

It is likely that the mass market addressed by these models will remain reserved for a few key players—just like the consumer and business computer chip market, which is dominated by a handful of global champions.

However, these generic foundation models are (and will remain) unsuitable for specialist applications. For one, they might outright lack the necessary capabilities, such as handling particular data modalities. Further, generic models often fail to reach the desired performance level in contexts that require domain-specific knowledge, such as analyzing and drafting legal documents. More generally, the models might not meet business, security, or regulatory constraints.

Against this backdrop, a company's decision to train its own foundation models is logical and almost inevitable. If a generic foundation model could solve the problem at the core of their business, they would not (or no longer) have a business case in the first place. However, if foundation models promise a superior solution or allow tackling new problems, training them becomes an integral part of maintaining an edge over the competition.

Data privacy, regulatory requirements and license restrictions

Using open foundation models seems attractive because it offloads the personnel-intensive and resource-hungry pre-training to a third party. Further, it allows companies to try a range of fully-trained models with different architectures and sizes at little cost.

However, many companies process data that is subject to strict privacy and governance requirements—whether this is due to regulations, customer demands, or because the data constitutes a trade secret.

For them, not knowing in any appreciable detail what data and processes were used to train a model quickly becomes an untenable liability.

Only when a company controls everything from the raw data to the downstream application, can it provide provenance information and accurately explain the data processing steps to customers, end users, and regulatory authorities. This becomes even more important if faced with different demands in different jurisdictions across the world or from different customers.

Furthermore, despite being labeled “open”, FM’ licenses might prohibit commercial use at scale or for specific purposes, as is the case for Meta’s Llama model family. This is similar to the well-known problems with freely available software that is subject to export restrictions or distributed under licenses containing non-compete clauses.

Building and maintaining competency

For companies that find themselves with foundation models at the core of their business, building and maintaining competency becomes a key strategic concern.

Even if relying on third-party API products or fine-tuning open foundation models could solve their needs in the short term, companies decide to invest in FM development to gain expertise in this key technology, believing that it will prove impossible to catch up in the future.

Only limited information about FM development is shared publicly. While the basic architectural principles and training techniques are widely known, little detail about specific setups and processes leaves a company’s boundaries. If it does, this information is often outdated already, with teams only sharing what they no longer consider a trade secret, promising a competitive advantage. Further, academic publications and company blogs accompanying model releases are not looking to provide guidance to developers but are predominantly created to signal innovation and novelty.

Often, their focus is on highlighting newly introduced features and comparing benchmark performances rather than describing the essential groundwork in a way that would allow others to replicate it.

The companies interviewed for this report share the belief that competency in FMs can only be acquired through first-hand experience and experiments. They consider training foundation models a rare capability today and believe this will likely remain the case going forward. Already, there are big skill and knowledge gaps between teams, which many of the interviewed companies expect only to grow as foundation model technology evolves.

Still, the argument for investing in foundation model capabilities can be difficult to make in light of widely available access to state-of-the-art foundation models through APIs. On the one hand, integrating with APIs offers clear technical parameters and predictable cost structures. On the other hand, the performance of flagship models like GPT, Gemini, and Claude sets user expectations that can be challenging to match for internal teams out of the gate.

However, the pricing of foundation model applications often widely undercuts the actual costs of providing the service. Stalling progress in performance through scaling model size (see “Trends—Scaling”), concerns about exhausting the readily available training data, and a lack of a clear path to profitable products make it unlikely that foundation model providers will be able to continue selling access below costs.

Reflecting on the decision to train foundation models

As discussed throughout this section, the decision to train foundation models hinges on a strong business case. If it is expected to bring clear benefits, alternatives have proven less favorable, and if a company is able to come up with the necessary budget, an investment in foundation model capabilities is a sound business decision.

Notably, whether this is the case appears to be independent of the teams' or companies' size. Several of the companies featured in this report are fairly small but are nevertheless fully committed to a foundation model strategy. Then again, we know of international enterprises that have decided against it.

Perhaps surprisingly, even among companies that are currently investing in FM capabilities, there can be a sentiment that training their own foundation models is usually not necessary. When asked to make recommendations to others considering going down this path, they instead suggest adapting open models.

Even among companies that are ***currently investing in foundation model capabilities****, there can be a sentiment that training their own foundation models is* ***usually not necessary****.*

However, the companies interviewed for this report do train their own foundation models, and—as we have already seen in this section—for good reason. What might explain this apparent contradiction is that the companies have, through their effort, closed the specific gap in the foundation model landscape they initially targeted.

Spotlight

In the therapeutic domain, the current so-called "frontier LLMs" are still pretty far below the bar, and our highly specialized tasks are really not their top priorities. There are a lot more low-hanging (yet pretty challenging) fruits for leading players like OpenAI and Anthropic to pick than fully understanding biology, physiology, and health.

So if your company has been collecting a huge amount of private data for a challenging task, you have a good reason to train your own models and to learn a lot by doing so. The expertise you'll have acquired will be totally worth it. First-hand experience is the best teacher in this area, and the better you understand LLMs, the better you set your strategy and build state-of-the-art tools.

There's always skepticism about training our own model, arguing we should outsource it. If biomedical science is going to be solved by the next GPT, then yes, perhaps they're right. But no, it's going to take a very long time, and we definitely need to grow our expertise unless we're going to be happy with relying on commoditized knowledge and tools – in which case, I have no idea how we can differentiate ourselves from others and win the competition.

Keunwoo Choi, Senior Principal ML Scientist
Genentech

What kinds of foundation models are companies training?

Foundation models play very different roles in companies' products and services. This, in turn, informs what kinds of models they train—the business objectives directly drive modeling decisions.

Throughout this report, we use the term "foundation model" in its original definition to describe large, pre-trained, generic models that can be adapted (see "Introduction—A note on terminology").

In this section, we'll deviate from this and use the term "foundation model" in its broadest possible meaning, encompassing any LLM, transformer-based model, or adaptable large-scale model.

As for the foundation model approach, the companies interviewed for this report fall into one of four broad categories:

1. Training FMs for direct application
2. Training a FM to derive models for downstream applications
3. Training smaller, task-specific models
4. Fine-tuning open models

Training foundation models for direct application

Companies in this category solve broad and complex tasks like language translation or climate modeling. The models they train must be applicable across many circumstances and/or process vast amounts of diverse data.

Companies following this strategy typically train a single flagship foundation model (line). They either use this model in their products or on behalf of customers, but do not sell the model itself.

Foundation models in this category are not used to derive models through fine-tuning but are adapted through in-context learning. They are also "foundational" to the company's business (their core product) and are typically among the largest models.

Training a foundation model to derive models for downstream applications

A typical company in this category uses AI technology to solve specific tasks in a domain, either for internal applications or for their customers. A dedicated team trains a foundation model that is domain-specific in multiple dimensions: the information it contains, the data modalities and structures it can process, and its operational requirements.

Based on this FM, the team derives task-specific models through fine-tuning to be used by internal or external customers. The foundation model is not shared, nor is access to it sold. These models are characterized by being "foundational" for downstream applications, not so much by their size or broad applicability.

They are typically built for a relatively narrow domain or set of tasks, allowing them to be significantly smaller than the generic models described by the term's original definition.

Training smaller, task-specific models

Organizations in this category are structurally similar to those in the previous category: They work in a specific domain and have to solve particular tasks. Often, a dedicated foundation model team caters to internal customers.

Instead of creating one generic FM from which task-specific models are derived, this team creates a dedicated model for each problem to be solved.

Although these models are not intended to be adapted to different downstream tasks, they are nevertheless referred to as foundation models because they share architecture and training techniques.

Training small task-specific models limits the scope and scale of the development effort in multiple ways. First, the models can be trained rather quickly on modest infrastructure. Second, they require only data for the particular task to be solved.

This allows training to commence immediately once this data becomes available, minimizing the time between data collection and model deployment. Third, model evaluation can focus on a single well-defined application scenario. There is no need to balance performance across multiple tasks, and over-optimizing on one task at the expense of others is no concern.

Spotlight

> *If I have to solve a particular problem, I don't need an 80-billion-parameter model. I need a small model that can understand the specific input related to that task and solve it. We are not building an end-to-end solution for everything—we are creating a suite of small-scale models that help us achieve our goals.*
>
> **Leader in AI Research**
> **Automation Platform for Financial Institutions**

Overall, this strategy is a path to quickly reaching good performance on a particular task. However, this fast return of business value comes at the expense of giving up the benefits of transfer learning (see "Current State of Foundation Model Training—Where does the training data come from, and what role does it play?—The shifting role of data in machine-learning").

Therefore, once teams have mastered the development of task-specific models, they often start to combine multiple tasks into one model. This reduces the overall training costs, unlocks performance improvements through transfer learning, and avoids the need to maintain an increasingly large number of models in parallel. Thus, while not foundation models according to the strict definition, small task-specific models can be "foundational" to a company's AI strategy.

Fine-tune open models

In contrast to training foundation models from scratch, fine-tuning them requires far fewer resources. Additionally, the process is more standardized, and comparatively mature tooling is available.

Thus, teams fine-tuning third-party foundation models can focus on curating high-quality datasets and optimizing task performance.

This motivates even companies with the budget and skills for training foundation models to restrict their efforts to adapting third-party models. A dedicated team either derives task-specific models similar to the teams in the second category or creates company-internal versions of standard FM applications through in-context learning.

While fine-tuning techniques like distillation and low-rank adaptation (LoRA) are essential for foundation model teams to master, fine-tuning is a very different process from training. Thus, a strategy exclusively focused on fine-tuning third-party models does not build FM capabilities to the extent that a strategy focused on small, task-specific models does. This is even more pronounced when adaptation is exclusively achieved through in-context learning or by utilizing a third-party platform.

Key term

Optimizing with quantization & distillation

Foundation model exhibit strong task performance but their resource requirements can make their application unviable or outright prevent their deployment.

Widely applied optimization techniques like quantization (decreasing memory usage and computation time by using lower numeric precision) and knowledge distillation (deriving a smaller "student" model from a complex "teacher" model) can produce less demanding model variants that still yield competitive task performance.

Learn more

The more a team depends on high-level abstractions and tools, the more it risks struggling with adapting to new developments in upstream models. Furthermore, this strategy's success and long-term viability hinge on the availability of suitable upstream models.

In any case, the fine-tuned models derive from a foundation model whose training data is unknown, which can lead to surprising behavior and generally constitutes a liability.

Addendum: hybrid approach

Not all companies' strategies can be sorted neatly into just one of the four categories. A particularly common hybrid approach is using open models where they are available and combining them with custom foundation models to create multi-modal models. This approach strikes a balance between resource demands, skill-building, and delivery timelines in service of the overall business objectives.

What does the hardware and data infrastructure look like?

The scale of the data centers and the vast amounts of energy that frontier labs like OpenAI and Anthropic spend on training their flagship foundation models are widely discussed. Thus, training foundation models might seem out of reach for companies not operating on billion-dollar budgets.

However, it can be accomplished on a much smaller scale. At Neptune, we've talked to teams that trained foundation models on as few as two GPUs. In our experience, the median currently seems to hover between 24 and 32 GPUs, with an average well above 128.

> *In our experience, the median currently seems to hover* ***between 24 and 32 GPUs****, with an* ***average well above 128****.*

Compute platform and budget

Graphics processing units (GPUs) are the default choice for FM training. They are the core building blocks of today's high-performance computing (HPC) clusters, as they provide unmatched performance on parallelizable computations. Specialized chips for deep learning workloads are available, but there is no widespread market adoption for foundation model training.

The most prominent example is Google's Tensor Processing Units (TPUs), which are mainly used by the company itself and its subsidiaries like DeepMind.

Nvidia continues to dominate the GPU market and has established a de facto industry standard with its CUDA framework. However, AMD has tripled its revenue from equipping data centers between Q2/2023 and Q4/2024, with half of the world's top 10 HPC clusters as of November 2024 running on their Instinct GPUs. Intel plays a significant role in the data center GPU market as well.

The scale of infrastructure and amount of energy required to train a foundation model depend on its size and architecture. In turn, the specific hardware constrains size and architecture, with the GPU memory as a key restriction. Further, larger models generally need more training data, leading to longer training times.

FM teams typically solve this chicken-and-egg problem by defining a compute budget beforehand. As a general rule of thumb, about a fifth of this budget can be spent on the main training run, with the remainder needed for experimentation and test runs.

Key term

Hyperparameter optimization

Hyperparameters influence model's makeup, learning behavior, and output generation. In contrast to model parameters (also called model weights), they are not learned from the training data but are set by the model's developers.

Finding an optimal set of hyperparameters is essential for efficient and effective training of FMs. Due to their computational demands, traditional methods such as grid search, tend to be impractical. Advanced strategies, like population-based training, Bayesian optimization, and adaptive LoRA, promise to balance computational effort and outcome.

Learn more

Neptune insight

Main and experimental training runs

FM teams often run main and experimental training runs in parallel. Their main run, which is training their model at full scale, often spans several weeks. Simultaneously, they launch experimental runs on the side that are short and use a smaller model variant. The teams use these experimental runs to explore new architectures, hyperparameters, or training schedules. They closely monitor for promising early signals and once they identify beneficial shifts in metrics, they incorporate these findings into the main training run.

This iterative approach helps teams rapidly converge on optimal configurations without risking disruption to their core training pipeline. However, implementing it effectively is a real challenge. Teams need robust monitoring systems to ingest and analyze high-volume training metrics in close-to-real-time, and the expertise to differentiate actual performance signals from statistical noise. Finally, the training infrastructure has to allow for seamless adjustments to the main training run without introducing instability.

Maintaining and operating foundation model training infrastructure

Except for the smallest ones, training foundation models requires multiple GPUs that are usually distributed across a cluster. While gaining access to GPUs was difficult and costly over the last couple of years, the companies interviewed for his report found that the availability of GPUs in the cloud and for purchase has significantly improved. Whether the GPU cluster is set up at a cloud provider or on-premises, teams find that a lot of engineering is required before foundation model training can commence. Compared to CPUs, which have been the backbone of compute platforms for decades, GPUs are a less mature platform.

Foundation model teams often report issues with drivers and encounter opaque hardware failures. The prerequisite expertise in hardware-level debugging, networking, and distributed systems is traditionally not found in data science and machine learning teams.

Likewise, this knowledge is not common among existing IT and infrastructure teams. Many ML teams that previously relied on their services find that when it comes to foundation model training, they need to take on more of the infrastructure work themselves. Some companies even go as far as assembling dedicated teams to handle hardware and infrastructure for FM training.

Optimizing hardware utilization

A perpetual challenge in FM training is using the available resources efficiently. Keeping the expensive GPUs under constant load often requires engineering a specialized training loop in accordance with the model architecture and the specific cluster setup.

The main bottlenecks are typically the limited size of the GPUs' memory and the transfer speed between cluster nodes. Teams can achieve a reduction in required memory and data transfer by re-computing intermediate results locally.

Another challenge is loading the training data. A single high-resolution pathology scan can be several GB in size, even when compressed. Handling data at this scale and providing it to the GPUs during training is a serious data engineering effort.

Spotlight

> *We're building foundation models for large-scale Earth simulations. For us, one training data point is several gigabytes. At this scale, it's very difficult to load the data fast enough to keep the GPUs utilized. A lot of our engineering efforts go into just that.*
>
> **Christian Bodnar, Co-Founder**
> **Silurian AI**

Still, cost and utilization optimization is not the be-all end-all. Experienced teams interviewed for this report tell us that once they had gotten accustomed to working with their training infrastructure and had resource utilization and cost management in place, they quickly found that it was prudent not to focus too much on improving easily quantifiable infrastructure costs and utilization.

Spotlight

> *It's tempting to focus on infrastructure optimization because it is easy to put numbers on it. However, there are often more important goals the team should focus on, even if they are harder to measure and quantify. We'll sometimes have to leave some GPU utilization on the table to make progress.*
>
> **Keunwoo Choi, Senior Principal ML Scientist**
> **Genentech**

Where does the training data come from, and what role does it play?

From a 10,000-foot view, companies and the public are increasingly becoming concerned about data sovereignty. In addition to owning models and the training infrastructure, as well as building expertise, controlling where data comes from and how it is used is a key component of many foundation model strategies.

The shifting role of data in ML

Prior to the advent of deep learning, machine-learning models were typically trained on meticulously curated and labeled datasets. A few hundred to a few thousand highly expressive samples were often sufficient.

Feature engineering, the practice of identifying high-signal features and compressing the data into lower-dimensional representations, played a major role.

With the advent of deep learning and increased compute and memory capacity, the datasets became significantly larger. ImageNet, a widely used, internet-curated dataset, consists of about 14.2 million images with labels and annotations. It's most popular subset, ImageNet-1K, contains 1.2 million images totalling 170 GB (about 140 KB per image).

FMs have brought yet another shift. The datasets and samples are orders of magnitude bigger, the individual samples are larger, and the data is less clean. The effort that was previously spent on selecting and compressing samples is now devoted to accumulating vast datasets.

The exemplary dataset is FineWeb, an English-language text dataset. It was derived from CommonCrawl corpus, consisting of websites scraped from the open internet. FineWeb's curators removed duplicate data, text in languages other than English, and applied a series of additional filters. The resulting dataset consists of over 15 trillion tokens with a total size on the order of 50 TB.

Beyond being able to process larger and more diverse data sets, foundation models exhibit strong transfer learning abilities, i.e., learning to solve a task by first training on data that does not contain examples of the task and later only on a few high-quality task-specific samples. In-context learning, where task-specific examples or instructions are only provided at inference time, is the pinnacle of this capability.

Transfer learning capabilities influence data acquisition for foundation model training. For example, training on monolingual data can improve the ability of foundation models in language translation. An LLM can learn to process a new language through fine-tuning on a monolingual dataset, retaining its ability to solve tasks like summarization or question answering originally acquired in a different language.

Key term

Zero-shot and few-shot learning

Chat and reasoning LLMs can solve tasks they were not explicitly trained to solve either out-of-the-box (zero-shot prompting) or when a couple of examples of how to solve the task are included in the prompt (few-shot prompting).

Zero-shot prompting is well-suited for simple tasks, exploratory queries, or tasks that only require general knowledge. It doesn't work well for complex tasks that require context or when a very specific output form is needed. Few-shot prompting is useful when a model has to learn a new concept or when a precise output format is required.

If complex multi-step reasoning is needed, neither zero-shot nor few-shot prompting can be expected to yield good performance. In these cases, fine-tuning of the LLM will likely be necessary.

Learn more

Making new data sources accessible to machine learning

The inherent ability of FMs to process large amounts of data without preprocessing, missing or contradictory information, and different modalities opens up ways to utilize previously untapped data sources.

Many organizations are in possession of large data repositories that until now could not be processed. Others rely on publicly available sources. Examples include large web crawl datasets like Common Crawl and FineWeb, open source code repositories on GitHub, or digitized libraries.

However, utilizing such data is not without issues. Scraping data from the internet, as practiced by the leading foundation model providers, raises concerns about low-quality and harmful information being ingrained into FMs, as well as about data privacy and copyright.

Spotlight

Most of today's climate and weather models rely on nicely structured input data coming from weather agencies. They put a three-dimensional grid over the planet, and for each cell, you get a data point comprising well-defined quantities.

But there's a ton of data out there that's a lot uglier and far less processed. There's data from weather stations randomly spread across the globe, from radar, from satellites. It's all very heterogeneous—some data is dense, some data is sparse, and the shapes and formats are different. It's challenging to work with, but it's literally lying around for people to use.

We're now at a point where we can build models that are able to absorb this kind of information. I believe that's the next frontier: utilizing any information you have, no matter if it's a dataset meticulously prepared by a government agency or signals you scrape directly off a sensor.

Christian Bodnar, Co-Founder
Silurian AI

Data qualities in FM training

Curating high-signal data remains a top priority for foundation model teams. Training on low-signal data, at best, makes training slow (and, in turn, costly). Commonly, it is detrimental to downstream performance. Data quality is particularly important toward the end of a training run. An established practice is to train with carefully vetted high-quality data in the final iterations.

In light of the vast amounts of data required for foundation model training, it is unrealistic to inspect every data sample or even a significant fraction of it. Thus, foundation model teams rely on heuristics to filter out undesirable samples at scale.

The role of human annotators in foundation model training

With the change in data sources used, the role of domain experts in the model training process evolved as well. Traditionally, they were involved in curating and annotating data before training. In foundation model training, their primary role is to evaluate models' performance on downstream tasks.

Still, many foundation model efforts rely on human labelers to create and prepare training datasets. While in language or climate modeling, the data itself contains the desired output (e.g., the next word), this is not always the case. Then, domain expertise is required to turn the data into information a model can learn from.

Reinforcement Learning from Human Feedback (RLHF)—credited for the breakthrough success of OpenAI's ChatGPT—relies on data created by labelers explicitly hired to do so, but also on feedback collected from users of foundation model applications.

Key term

Reinforcement Learning from Human Feedback

Reinforcement Learning from Human Feedback (RLHF) is credited with the breakthrough success of LLMs. The process consists of three steps: collecting human feedback in the form of a preference dataset, training a reward model to mimic human preferences, and fine-tuning the LLM using the reward model.

RLHF is an efficient way of integrating human feedback into the training process to ensure that models not only produce coherent and useful outputs but also align more closely with human values, preferences, and expectations.

Learn more

Synthetic data

An ML model requires a certain number of data samples to learn a concept or relationship. Thus, as discussed above, the relevant quantity is not the number or size of the data samples but the amount of pertinent data samples contained in a dataset.

This becomes a problem for signals that rarely occur and thus are rare in collected data. In order to include a sufficient number of data samples that contain the signal, the dataset has to become very large. Even though the majority of the additionally collected data samples are redundant. Oversampling rare signals risks overfitting on the samples rather than learning robust representations of the signal. A more useful approach is to create data samples that contain the rare signal artificially.

Synthetic data has a long history in machine learning. It is used to augment or replace collected data samples, often with the goal to reduce costs, maintain privacy, and balance datasets. The companies interviewed for this report that utilize synthetic data treat the generation as an inherent part of their foundation model efforts. They develop their own approaches, building on established methods and recent progress in the field.

In some scenarios, the validity and integrity of synthetic samples can be tested automatically. For example, generated source code can be compiled and executed to ensure it produces the desired outcome.

In many other cases, having a human in the loop is crucial to ensuring that synthetic samples resemble the desired real-world signal. Otherwise, models optimize for irrelevant artificial information, which might even negatively influence their performance on the downstream task.

Spotlight

In our medical image data, a lot of crucial features are extremely rare. Even if we train a model on a very large dataset, the number of samples a feature appears in is too small for the model to learn it sufficiently. Thus, we synthesize images that contain such crucial features– but in a way that we, as developers, are not the ones prescribing what exactly those features are. Then, we validate these images with experts.

Robert Berke, CTO and Co-Founder
Kaiko.ai

How are foundation model teams organized?

Traditionally, machine-learning teams consisted of data scientists and ML engineers who handled deployment and operations. This is insufficient for successful FM projects. Implementing the model architecture, preprocessing a dataset, and maintaining training pipelines are not enough.

Therefore, while their setup varies, what all foundation model teams surveyed for this report have in common is that they are multi-disciplinary.

As one team lead put it, "You need someone to do the maths," but at the same time, people who can handle distributed data processing and training on large-scale infrastructure.

Software engineering is crucially important and can no longer be neglected at the scale of foundation model training. Some companies establish dedicated infrastructure teams, but many embed infrastructure and SRE experts within their teams as boundaries between traditional roles blur.

Hiring for foundation model teams

Because of the depth and range of skills that have to come together to train a foundation model, assembling and paying for a team can be a bigger hurdle than the infrastructure costs or data availability. The smallest team among those we interviewed have just a handful of members, with each member bringing deep expertise in several domains. The largest foundation model initiatives carried out by companies we interviewed for this report involve up to 500 people.

Since the skills required are broad and specific, hiring for FM training teams is a challenge. While there are many people who have used and fine-tuned foundation models, but very few people have experience pretraining foundation models. Thus, companies typically have to rely on identifying top performers who can get up to speed quickly. For smaller, less-resourced companies, the high salaries afforded by frontier labs are another barrier to attracting talent.

*There are many people who have experience **using and fine-tuning** foundation models, but **very few people have pretraining experience**.*

Spotlight

You need to make sure to take hiring very seriously. There's a seemingly endless influx of applications for research positions at DeepL. So it's a lot more about finding the right people out of all the ones that apply and then making sure that you help them grow and develop. It's a hard-fought battle, but we've done really, really well.

Stefan Mesken, Chief Scientist
DeepL

Trends in foundation model training

Chapter summary

- **Models and training datasets are getting larger,** generally leading to task performance improvements. However, **simply scaling up is not a silver bullet,** with small models and alternative architectures showing great potential toward efficient high-performance models.

- While **foundation models that are currently in the spotlight predominantly process text and images**, great progress is underway **developing models for other data modalities like tabular and event data**.

- Increasingly, **companies are investing in on-premises training infrastructure**, opting to build highly customized compute clusters rather than relying on cloud services, trading flexibility for predictable architecture and availability.

- Faced with high resource requirements, teams increasingly **focus on data and software engineering** to make foundation model training more efficient. This requires skills and capabilities beyond those of a traditional machine-learning engineer.

Since the advent of large language models eight years ago, the landscape has evolved significantly and there is no end to this development in sight. Across the interviews conducted for this report, we identified four broad categories of trends:

1. Scaling foundation models for better task performance
2. Training foundation models for data modalities beyond text and images
3. Establishing and expanding on-premises training infra
4. Improving training efficiency through data and software engineering

In the following sections, we will take a closer look at each of these trends, discussing the underlying drivers and highlighting how leading foundation model teams view the prospects and challenges.

Scaling

Ever since the first transformer model was released in 2017, model sizes have increased dramatically. While OpenAI's first GPT model had about 120 million parameters, the largest models in Alibaba's Qwen family and Meta's Llama family contain around 70 billion. While this increase in parameters correlates with major task performance improvements, there is no simple causal link between the two.

Intuitively, a bigger model has more capacity to store information and represent complex relationships. Further, the larger the training dataset, the more variations it potentially contains, and the smaller the impact of outliers or erroneous samples. Empirically, foundation model teams find that scale helps with robustness and reduces hallucinations.

So-called scaling laws underpin this intuition. They predict the model's performance on the training objective scales with model size, dataset size, and compute budget according to a power law. This insight led to a concept of compute-optimal training, an empirically validated hypothesis stating that model size and the number of training tokens should be scaled equally.

Overall, we see a trend towards larger models and longer training runs. While the current scale of foundation model training already stretches what is possible with today's hardware and data center infrastructure, analysts predict that we are still far from the limit. Nevertheless, simply scaling model size, acquiring more training data, and increasing compute budgets does not necessarily translate to improved performance on a particular downstream task.

Neptune insight

Monitoring long training runs

Long-running FM training jobs frequently branch into multiple experimental paths or variations. This is where forking a training run (conceptually similar to forking code under version control) proves invaluable.

By creating a new branch of the training from a specific checkpoint, data slice, or hyperparameter setup, FM teams maintain a clear lineage of each experiment. This approach allows researchers to backtrack when needed and trace exactly which adjustments were made at each stage of training, ultimately improving reproducibility and clarity.

Explore forking in Neptune

A closer look at the impact of scale

Increasing the number of model parameters and training samples can lead to better performance on tasks, but it's not a guaranteed solution. If the architecture or training process is not exactly right, training a foundation model at larger scales amounts to little more than a higher electricity or cloud bill at the end of the month.

Some of the FM teams we interviewed for this report have access to more data than their current models can take in. They find that performance levels off well before training data is exhausted. Thus, they are focused on increasing their models' capacity.

Other teams we interviewed do have plenty of data at their disposal, but find that it lacks variety. In other words, their models have more capacity than the information contained in the training data. Thus, there is potential for achieving the same performance with smaller, less resource-demanding models and reducing training duration, allowing for more frequent model updates. Indeed, the performance of small FMs on standard benchmarks has significantly improved in recent years.

Since training smaller models and using less data generally leads to lower costs, there is a strong tendency to aim for reducing model size. However, as experienced FM teams interviewed for this report point out, certain capabilities and effects only emerge at scale. (To give an intuitive example, consider training a randomly initialized multi-million parameter model with state-of-the-art architecture on a handful of data samples.) As a consequence, they emphasize the importance of conducting experiments at the scale of the target model. In fact, running fewer and shorter large-scale experiments can provide more valuable insight than long-running small-scale experiments.

Scaling beyond model and data size

A particular way in which foundation model architectures have been scaled in recent years is their context size, i.e., the number of input tokens they can process at once. While earlier LLM generations were restricted to a few thousand tokens, in 2025, leading models like Meta's Llama 3.3 and OpenAI's GPT 4.5 have context sizes of 128,000 tokens, enabling them to process short books like Tolkien's "The Hobbit" in their entirety. Meta's Llama 4 "Scout", released in April 2025, has a context length of 10 million tokens, comfortably fitting the entire "Lord of the Rings." Further increasing context sizes is particularly important for models processing images, as it allows for training them on high-resolution samples rather than having to down-scale images or split them up into smaller patches.

A second prominent trend is scaling model capacity and improving task performance without simply increasing the number of parameters. Mixture-of-Experts (MoE) models like Google's Gemini, DeepSeek's R1, and Meta's Llama 4 only use part of their parameters for any given input, allowing them to scale the total parameter size while keeping the computational demands and latency in check.

In contrast, State Space Models like Mamba have an entirely different architecture that efficiently handles long-range dependencies in sequences, thereby overcoming an inherent limitation of transformer models.

Key term

Mixture of Experts

Mixture of Experts (MoE) is a type of neural network architecture that employs sub-networks, so-called experts, to process specific input parts. Only a subset of experts is activated per input, enabling models to scale efficiently.

Experts are typically distributed across multiple devices. Gating and load balancing mechanisms to dynamically route inputs to the most relevant experts, ensuring evenly distributed computation. MoE Foundation Models exhibit faster training and inference compared to equally-sized dense Foundation Models and yield better performance, especially on multi-domain tasks.

Learn more

Key term

State Space Models

State Space Models (SSMs) use first-order differential equations to represent dynamic systems. Through maintaining continuous representations of time-dependent data, they are able to efficiently approximate long-range dependencies in sequence modeling.

Discretization of continuous-time SSMs lays the groundwork for computationally efficient natural language processing. Increasingly sophisticated sequence-to-sequence state-space models that emerged since the early 2020s pave the way for viable SSM-based alternatives to transformer models.

Learn more

Modalities beyond text and images

Traditionally, the transformer architecture and foundation models were seen as synonymous with large language models. Later, text-to-image (e.g., OpenAI's Dall-E and Stability AI's Stable Diffusion), as well as text-to-video models (e.g., OpenAI's Sora) broadened the general public's perception to include these modalities.

Key term

Multimodal Large Language Models

Multimodal Large Language Models (MLLMs) process data from different modalities like text, audio, image, and video. Compared to text-only models, MLLMs achieve richer contextual understanding and can integrate information across modalities, unlocking new areas of application.

Prime use cases of MLLMs include content creation, personalized recommendations, and human-machine interaction.

Combining different modalities and dealing with different types of data come with new challenges, such as the need to align heterogeneous data.

Learn more

The most common combination—and the companies interviewed for this report are no exception here—is the combination of textual and visual information. A typical strategy for creating multi-modal FMs is to integrate pre-trained vision and language models, conjoining the models' internal representations.

Frequently, when foundation models are first applied to new data modalities, this is achieved by converting the data into a modality that existing models can process. For example, a website is either ingested as raw source code by an LLM or its rendered form is processed as a static image by a vision model.

Yet, the inherent loss of information incurred with this transformation is a major limitation, encouraging teams to explore FMs that can natively handle modalities beyond text and images.

Tabular data

Numerical and categorical data organized in tables is the core domain of many traditional machine-learning approaches. To this day, regression models and boosted trees remain the work horses of most data science teams working with tabular datasets.

Deep learning and, in extension, foundation models, excelled at non-structured and non-numerical data like text and images. While they outperformed prior methods in these modalities, FMs struggled with tabular data.

Approaches that attempted to convert tables into text and feeding it to an LLM came up against the information loss inherent to this transformation and the limited context length of early models. The TabTransformer published in 2020 uses a transformer to compute contextual embeddings for categorical features, which are then further processed by a standard deep learning architecture.

More fundamentally, LLMs cannot perform numerical operations in the same way other ML algorithms inherently can. While LLMs can answer simple questions like unit conversions or assist with algebraic transformations following standard patterns, they cannot genuinely perform calculations or solve equations.

Despite all these limitations, LLMs offer a distinct advantage. They can understand text, whether it's column names, legends, or categorical labels. It's this information that allows data scientists to bring in domain expertise and interpret the data, informing every processing and modeling step.

Traditional machine-learning approaches are restricted to working with the numerical and encoded categorical data in the table, which in itself is meaningless.

Spotlight

CARTE, our tabular foundation model, takes contextual information into account. The benefit this brings over traditional approaches is bigger on some downstream tasks than on others. That's probably related to how much background information is useful for the particular tasks.

There might also be downstream tasks that are very specific to a niche application in a niche domain, for which it's very hard to include prior knowledge in the foundation model. It's an open question if we're going to be able to make the foundation models big enough to cover these niche tasks. If it turns out that it's not possible, we'll need ways to detect situations like this to avoid eliciting hallucinations.

Gael Varoquaux, Research Director
Inria

Going beyond language, foundation models bring transfer learning to tabular data. However, transfer learning on tabular data is challenging. On the level of individual table entries, transfer learning requires entity matching (i.e., identifying if two—similar or different—words refer to the same entity). On the level of entire tables, transfer learning requires the ability to match the structure of different tables (schema alignment). TabPFN is a foundation model that frames predicting labels for tabular data as a Bayesian problem. It takes a training dataset and unseen samples as input and generates predictions in a single forward-pass, making it fast and efficient. Internally, TabPFN consists of layers that combine row-wise and column-wise attention with a multi-layer perceptron. The model is exclusively trained on synthetic data, eliminating any risk of data leakage.

Multi-dimensional tokens

Before an LLM can process a text, it has to be converted into one-dimensional tokens: Each token is represented by a single number, corresponding to its index in the vocabulary. But not all data is like that.

Just like text, events also occur in an ordered sequence. But each event is a multi-dimensional entity with multiple, potentially high-cardinality fields. To effectively perform sequence-to-sequence modeling on such data, a foundation model should natively be able to handle multi-dimensional tokens.

A typical application of foundation models pursued by companies interviewed for this report is processing financial transactions. Given a sequence of transactions, the model is asked to produce likely subsequent transactions. Based on this output, analysts can derive market developments or automatically derive insights about a customer's future spending. Accurately predicting transaction patterns—and detecting deviations from expectations—is a key capability to prevent fraud, a major concern for financial institutions all over the world.

A key challenge in working with multi-dimensional tokens is the required vocabulary size. Events like financial transactions can comprise multiple high-cardinality fields (e.g., merchants or banks), and new values might appear over time. Ongoing research addresses this problem in two directions. On the one hand, by enabling models to handle larger vocabularies without diluting the information contained in the input. On the other hand, by applying embedding techniques to compress multi-dimensional data so that it can be accurately represented with a small vocabulary.

On-prem training infrastructure

Media discourse about training infrastructure for foundation models centers on investments in data centers by cloud providers.

Industry leaders and public stakeholders alike seemingly outbid each other in adding compute capacity, announcing initiatives like the $500 billion "Stargate Project" (OpenAI, Oracle, and SoftBank) or the 200 billion € "InvestAI" (European Union). However, many companies are already training FMs in-house or are looking to move from cloud-hosted infrastructure to on-premises setups.

For some organizations, fully owning and controlling their hardware has been a necessity from day one due to regulatory and data privacy requirements (see "Why are companies training foundation models?—Data privacy, regulatory requirements, and license restrictions"). In the extreme case, they operate their foundation model training and inference infrastructure in air-gapped environments.

Others have started out in the cloud and decided to move to their own hardware as they establish their foundation model initiatives as a core part of their business. The scale of this differs wildly. While some teams run on workstations, several of the companies interviewed for this report have recently purchased Nvidia DGX SuperPOD clusters.

Neptune insight

Tooling for on-premise FM training

Companies that self-host GPU clusters for foundation model training must ensure their supporting tools, such as experiment tracking, model-serving frameworks, and data versioning solutions, can also run reliably on-premises.

The scale of foundation model training can overwhelm systems that weren't designed for massive throughput, so it's important to choose or build tooling that can handle high-volume logging and maintain robust performance within your own infrastructure.

Check self-hosted Neptune

Decision criteria and strategies

Motivations for investing in on-premises FM training infrastructure so can be subsumed under one of three categories.

First, it guarantees available capacity. While many cloud providers allow customers to reserve capacity, the availability of these contracts is limited, and requirements (e.g., min volume or lengths) can make them financially unfavorable.

Second, for companies with a predictable, steady capacity demand, buying and operating their own hardware can undercut cloud costs significantly.

Third, building out on-premise training infra allows organizations to tailor the setup to their needs. Conversely, it makes the hardware platform (e.g., GPU models and network layout) highly predictable. This allows teams to heavily optimize their training code, without running the risk that the investment is rendered wasted once a migration to a new cloud platform becomes necessary because the current contract cannot be renewed due to changing offers or it becomes economically unviable.

GPU availability in the cloud

In the years following the advent of the transformer model, chip manufacturers and cloud providers struggled to keep pace with the rapid rise in demand for GPU capacity. This "GPU scarcity" was a particular problem for teams entering the field without a big budget.

For the companies we interviewed for this report, the problem has eased considerably in the recent past. None describe GPU access as a limitating factor. Some even find that they can purchase on-demand capacity when needed, rendering long-term contracts unnecessary.

Still, it remains challenging and costly for many companies to ensure permanent access to GPU capacity for foundation model training.

While some analysts see developments like Microsoft's abandoning of data center projects and withdrawing from commitments with AI-focused data center provider CoreWeave as heralds of falling prices, others believe that even if data center capacity is scaled at the maximum speed possible, it will be the major inhibitor of industry growth and progress in foundation models.

Improving training efficiency through data and software engineering

Although today's foundation models exhibit satisfactory performance on many tasks, the computational resources required to train them remain a barrier to widespread application. Even where models on the order of ten billion parameters, which are considered small by 2025 standards, prove sufficient (see "What kinds of foundation models are companies training?—Training smaller, task-specific models"), their training costs still make many applications economically unviable. Thus, improving resource efficiency is a major objective.

In pursuit of this goal, teams reach out to software and data engineering to improve training efficiency and hardware utilization, in contrast to further downsizing models and exploring architecture like Mixture-of-Experts (see "Scaling—Scaling beyond model and data size"), which potentially negatively impacts task performance.

Shifting focus from task performance to efficiency

Most companies we interviewed for this report highlighted the importance of software and data engineering to the success of their foundation model initiatives. Once they solved their modeling and training fundamentals, they found that focusing on increasing system uptime and efficiency offered the greatest return on investment.

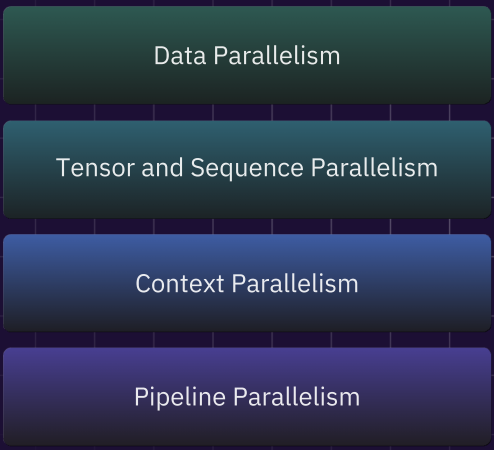
Common focus areas comprise data loading and processing, including re-implementation of performance-critical code paths in lower-level languages like Rust. Many teams coming from a research or data science background experience this shift to an engineering focus as a major challenge (see "Current State of Foundation Model Training—How are foundation model teams organized?").

As maximizing infrastructure utilization becomes a key objective, monitoring and debugging take center stage. Successful foundation model teams establish observability practices and tooling to quickly identify bottlenecks and the root cause of issues, minimizing downtimes and the impact of inevitable errors on training progress (see "Best Practices for Foundation Model Training—"Track all your experiments – and keep backups").

Going deep on GPUs and networking

Aside from optimizing data pipelines, improving training efficiency mainly involves building up the training loop according to available infrastructure and model architecture, balancing memory and compute requirements as well as minimizing network communication overhead. While the available memory puts a hard upper bound on the maximum size of a training step, teams can decide to trade an increase in computations and network communication for a reduction in memory consumption.

At a high level, the main strategies are:



Data Parallelism

Tensor and Sequence Parallelism

Context Parallelism

Pipeline Parallelism

- ✓ *Data Parallelism: Replicating the model on several GPUs, processing a different micro batch of data on each, and combining the results.*
- ✓ *Tensor and Sequence Parallelism: Sharding the model weights, gradients, optimizer states, and activations across GPUs.*
- ✓ *Context Parallelism: Splitting the input and model along the sequence dimension, with each GPU holding and computing only a fraction.*
- ✓ *Pipeline Parallelism: Splitting the model's layers across GPUs.*

Efficiently engineering foundation model training processes requires an intimate understanding of low-level details. For example, Meta forked Nvidia's NCCL library (NCCLX) to improve network latency when training their Llama 3 models.

Another prominent example is DeepSeek, a Chinese company that, due to trade restrictions, only had access to a limited number of Nvidia's H800 GPUs, a restricted variant of its former flagship H100 model.

Through an improved training paradigm and engineering, the DeepSeek team was still able to train their R1 MoE foundation model to state-of-the-art performance, requiring but a fraction of the compute resources that their competitor in North America and Europe required to achieve similar results.

Best practices for foundation model training

Chapter summary

- **Begin with proof-of-concept projects** to assess if foundation models can solve a problem and to identify current gaps in skills and infrastructure.
- **Build deep expertise across all components of the foundation model training process,** as small details can make or break success. Debugging requires intimate hardware knowledge.
- Thoroughly **document and analyze all experiments, configurations, and model checkpoints** to understand which changes truly improve performance.
- Don't lose sight of business objectives by **developing evaluation suites that test downstream performance in realistic scenarios** rather than relying solely on training metrics and benchmarks.
- **Resist chasing every AI trend and alleged breakthrough** by keeping research or business objectives as your central focus.

We concluded all interviews conducted for this report by asking our interviewees for recommendations, lessons learned, and advice they would give to teams starting out with training foundation models.

"Nothing beats getting started and trying"

"Know what you're doing"

"Track all your experiments—and keep backups"

"Keep an eye on downstream performance"

"Stay focused on what you're doing, and don't give in to every trend"

“Nothing beats getting started and trying”

Once you have concluded that training foundation models is a promising path toward reaching your business goals, your next step should be to get started with a proof of concept.

Given the rapidly evolving foundation model landscape and lack of long-term experience with the technology, it is impossible to determine if and how foundation models can solve a particular task without actually running experiments. Similarly, assessing the readiness of your team and infrastructure and identifying the gaps you’ll have to close is best done through working on a prototype.

Spotlight

> *The number one thing is to start with something as quickly as possible. In the beginning, we picked an architecture that seemed promising and made it work. But then, after some iterations trying different hyperparameters, we started to analyze what’s working, what’s not working, and why.*
>
> *Initially, this meant spending a lot of time looking at metrics and adjusting the model architecture. But over time, as a team, we learned to look for subtle signals that tell us how our model is learning and performing—and why it’s failing in certain scenarios.*
>
> **Leader in AI Research**
> **International Financial Services Company**

“Know what you’re doing”

Training foundation models requires an in-depth understanding of model architectures, training data, hardware infrastructure, and training algorithms. The breadth of knowledge and skill required to train FMs often appears daunting to newcomers. This is exacerbated by a lack of standard approaches or end-to-end third-party solutions to fall back on.

Teams that are training foundation models unanimously stress that it’s paramount to truly understand all the components and own the entire training process. Often, it’s small details that make or break a training run, and debugging and optimally utilizing compute resources requires deep familiarity with the hardware.

Not every team member has to become an expert on everything. Successful teams divide responsibilities and foster a culture of shared learning to reach a holistic understanding. They tap different knowledge sources and record their failures and progress (see “Current state of foundation model training—Why are companies training foundation models?—Building and maintaining competency”).

Neptune insight

Monitoring model internals

Owning the training process also means capturing detailed metrics about your model’s internals. Logging layer-level parameters, gradients, and activations can be incredibly revealing when investigating subtle issues or unexpected learning behaviors.

For instance, if a particular layer’s gradient consistently saturates or vanishes, you can spot the warning signs early, before it derails a costly multi-week training run. A robust experiment tracker that supports fine-grained logging helps you piece together these clues swiftly and prevents small divergences from turning into large-scale failures.

See an example in Neptune

“Track all your experiments—and keep backups”

A large number of moving parts have to work together to produce a foundation model, and minor configuration adjustments can have a large influence on training performance. Thus, logging training, hardware, and evaluation metrics is an essential part of the process.

In the early days, teams often find themselves making ad-hoc adjustments to hyperparameters based on spurious insights. Later, they discover they are not only unable to tie performance improvements to a particular change but also struggle to return to a previous model, data, and training configuration that yielded better outcomes. During our interviews for this report, one team even described losing several months of work due to a lack of backups.

At a basic level, FM efforts are no different from data science or software engineering projects. You should use a version control system like Git for your source code and notebooks, and save artifacts like model checkpoints in a consistent way.

When it comes to logging, foundation model teams interviewed for this report recommend logging as much as possible. To avoid drowning in noise, teams have to make a conscious effort to become familiar with the data they collect and identify the signals to look out for. Initially, directly plotting logged metrics and creating simple model scorecards can go a long way.

Over time, teams tend to build out sophisticated monitoring and evaluation platforms that can handle the vast amounts of signals collected during foundation model training runs and evaluations.

Neptune insight

Experiment tracking for large-scale training

Some teams worry that "logging everything" during foundation model training might swamp their tracking system or cause performance bottlenecks. However, tools purpose-built for FM training, like Neptune, have architectures that scale horizontally to handle any data load. This means you can safely log the full breadth of metrics (every loss curve, layer-level activation, or gradient norm). You gain the freedom to capture all the details you need for post-hoc analysis without sacrificing training speed or system stability.

Neptune insight

Reporting training progress and results

FM teams evaluate models across multiple dimensions, ranging from technical metrics to domain-specific benchmarks. Standardized reports with effective visualizations enable engineers and researchers to determine which experiments to stop or continue. These reports also facilitate communication with stakeholders and other teams, making them essential for steering model and product development.

Explore reports in Neptune

"Keep an eye on downstream performance"

Foundation model teams risk losing sight of business objectives amidst the countless potential improvements waiting to be explored and problems to solve. It requires an ongoing, conscious effort to look beyond training metrics and performance on standard benchmarks.

While important indicators, training performance does not strongly correlate with a model's performance on downstream tasks or its suitability for adaptation. Metrics and benchmark scores are typically averages across many samples, hiding systematic failures and providing little insight into the quality and consistency of a model's outputs.

The teams interviewed for this report agreed that organizations should invest in an evaluation suite. This can comprise curating collections of input/output pairs that a prior model version has struggled with or performed well on. Further, it can include metrics like response latency and resource utilization. The closer the evaluation tasks mimic a model's production scenario, the more expressive the scores and qualitative results.

Neptune insight

Training stability

While it’s crucial to extend your evaluation beyond raw training metrics or benchmark averages, don’t lose sight of training stability. Even seemingly minor instabilities can undermine downstream performance, despite solid metrics on paper. Staying vigilant about stability from the start helps ensure that any improvements reflected in your evaluation suite truly carry over to downstream scenarios.

“Stay focused on what you’re doing, and don’t give in to every trend”

With all the media attention and widespread interest in generative AI, it is difficult for FM teams to avoid the perception that one is falling behind and delivering sub-par work.

In a self-reinforcing cycle, it is precisely the great public attention paid to foundation models that tempts stakeholders to advertise every new development as an outstanding breakthrough. On any given day, new research papers are shared on social media, tools and frameworks are released, and companies are announcing new AI-powered products. It’s all too easy for teams to build up an ever-growing list of new ideas to try or improvements to make.

To stay on course, teams have to remind themselves that most alleged breakthroughs will come and go without leaving a lasting mark. If a particular new development truly passes the test of time, it is unlikely to go unnoticed even by those who are not tracking each and every announcement. While being an early adopter can constitute a competitive advantage, it comes with an increased risk of wasting time and resources.

Successful foundation model teams avoid this trap by putting their research or business objectives front and center. As with every other decision and activity, they attach importance to knowing exactly why they are doing something and how it will bring them closer to their goal.

Summary and outlook

In this report, we examined the current state of foundation model training across organizations. Companies are training foundation models to address domain-specific problems, meet compliance requirements, and build internal competency. This goes along with a major shift in data strategy, with organizations moving from diligently handpicked datasets toward acquiring diverse raw data sources, complemented by synthetic data generation.

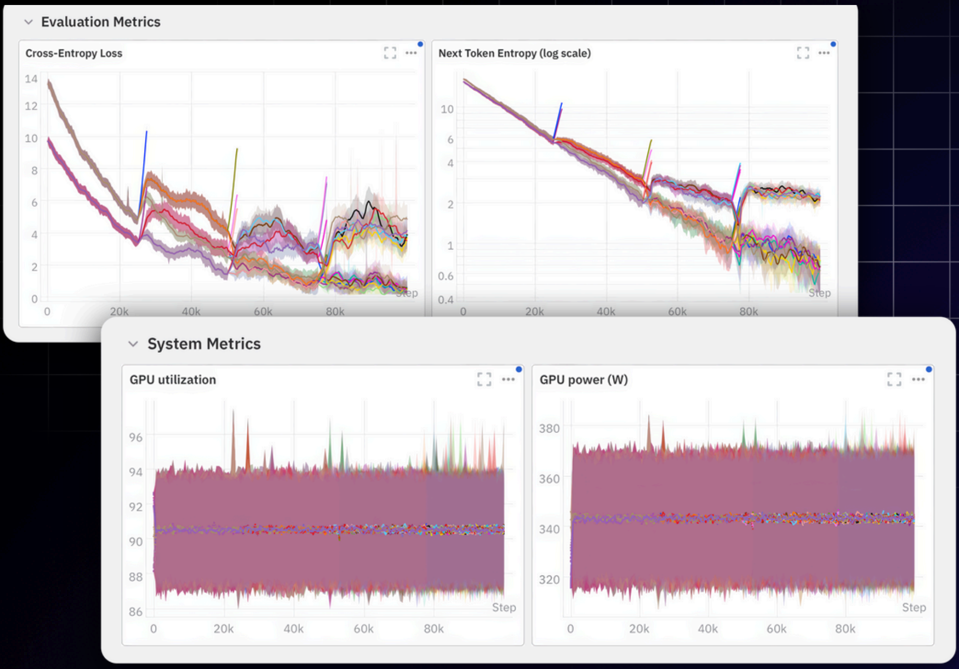
FM development requires multi-disciplinary teams with strong data and software engineering capabilities, making recruitment of appropriate talent a primary challenge. While model and dataset size continue to grow in pursuit of corresponding performance improvements, our analysis highlighted that scaling the number of model parameters and data samples alone is insufficient. Smaller models and alternative architectures like Mixture-of-Experts models show considerable promise.

In light of the substantial costs of training large-scale models, teams increasingly focus on data and software engineering to make training more efficient.

GPU infrastructure remains the predominant FM training platform, with a growing trend toward on-premises infra investment that prioritizes predictable architecture and availability over flexibility.

Finally, the report identified key characteristics of successful foundation model strategies, such as early proof-of-concept projects, a holistic approach to building expertise, application-based performance evaluation, and maintaining a strong focus on core business objectives.

Thank you!



neptune.ai is a purpose-built experiment tracker for teams training foundation models. It provides real-time monitoring of thousands of metrics, including losses, evals, and model internals like layer-level gradients or activations. This, combined with lightning-fast navigation through logs and support for experiment forking allows researchers to quickly evaluate model performance, debug issues, and keep the training stable while reducing wasted GPU cycles.

To meet the infrastructure and compliance requirements of large-scale training, apart from the cloud version, Neptune offers scalable, reliable, and secure self-hosted deployment.

Neptune is **trusted by leading AI labs**—such as OpenAI, Poolside, Bioptimus and InstaDeep—training foundation models of all sizes and across various domains.

OpenAI poolside Navier AI InstaDeep™ Achira_ BIOPTIMUS

Appendices

Participants

Company	Size	Type	Industry	Location	Role
DeepL	Mid-sized enterprise	Privately held	Software	Germany	Chief Scientist
Kaiko.ai	Scale startup	Privately held	Healthcare	Netherlands	CTO & Co-Founder
Genentech	Enterprise	Privately held	Biotech & pharma	USA	Senior Principal ML Scientist
Synerise	Scale startup	Privately held	Software	USA	Chief AI Officer
Blue Raven AI	Startup	Privately held	Research	Singapore	Head of Applied AI
Inria	—	Government agency	Applied research	France	Research Director
Silurian AI	Startup	Privately held	Environmental sciences	USA	TO & Co-Founder
Anonymized	Scale startup	Privately held	Software	USA	Leader in AI Research
Anonymized	Scale startup	Privately held	Cybersecurity	USA	Leader in AI Research
Anonymized	Enterprise	Privately held	Multi-industry	Germany	Leader in AI Research
Anonymized	Enterprise	Public company	Financial services	USA	Leader in AI Research
Anonymized	Enterprise	Public company	Software	USA	Leader in AI Research

References

- Alsop, Thomas. “Data Center Segment Revenue of Nvidia, AMD, and Intel from 2021 to 2024, by quarter.” Statista, 27 Feb. 2025, source.
- Ben Allal, Loubna, et al. “SmolLM – Blazingly Fast and Remarkably Powerful.” Hugging Face, 16 July 2024, source.
- Bommasani, Rishi, et al. On the Opportunities and Risks of Foundation Models. Center for Research on Foundation Models, 2021, source.
- Dzieza, Josh. “AI is a Lot of Work.” Intelligencer, 20 June 2023, source.
- Epoch AI. “Notable AI Models.” Epoch AI, source. Updated 16 May 2025.
- “EU Launches InvestAI Initiative to Mobilise €200 Billion of Investment in Artificial Intelligence.” European Commission, 10 Feb. 2025, source.
- Ford, Brody, et al. “Microsoft Pulls Back on Data Centers from Chicago to Jakarta.” Bloomberg, 3 Apr. 2025, source.
- Frymire, Luke. “The Length of Time Spent Training Notable Models is Growing.” Epoch AI, 16 Aug. 2024, source.
- Gottlieb, Isabel, and Cassandre Coyer. “AI’s Data Appetite is Huge. That’s a Problem for Privacy Laws.” Bloomberg Law, 24 July 2024, source.
- Grattafori, Aaron, et al. “The Llama 3 Herd of Models.” Meta, 23 July 2024, source.
- Gu, Albert, and Dao, Tri. “Mamba: Linear-Time Sequence Modeling with Selective State Spaces.” Carnegie Mellon University and Princeton University, 31 May 2024, source.
- Hawkins, Mackenzie, et al. “DeepSeek’s AI Model Tests Limits of US Restrictions on Nvidia Chips.” Bloomberg, 27 Jan. 2025, source.
- Hoffmann, Jordan, et al. “Training Compute-Optimal Large Language Models.” DeepMind, 2022, source.
- Hollmann, Noah, et al. “TabPFN: A Transformer that Solves Small Tabular Classification Problems in a Second.” Eleventh International Conference on Learning Representations, 1-5 May 2023, Kiwali, source.
- Honderich, Holly. “Major Canadian News Outlets Sue OpenAI.” BBC News, 29 Nov. 2024, source.
- Huang, Xin, et al. “TabTransformer: Tabular Data Modeling Using Contextual Embeddings.” Amazon AWS and PostEra, 2020, source.
- Hu, Edward J., et al. LoRA: Low-Rank Adaptation of Large Language Models. Microsoft Corporation, 2021, source.
- Kaplan, Jared, et al. “Scaling Laws for Neural Language Models.” OpenAI, 2020, source.
- Kim, Myung Jun, et al. “CARTE: Pretraining and Transfer for Tabular Learning.” International Conference on Machine Learning, 21-27 July 2024, Vienna, source.
- Korinek, Anton, and Jai Vipra. “Concentrating Intelligence: Scaling and Market Structure in Artificial Intelligence.” Institute for New Economic Thinking Working Papers, 2024, source.
- Liu, Ruibo, et al. “Best Practices and Lessons Learned on Synthetic Data.” Conference on Language Modeling, 7-9 Oct. 2024, Philadelphia, source.
- Maffulli, Stefano. “Meta’s LLaMa License is Not Open Source.” Open Source Initiative, 20 July 2023, source.
- Marshall, Matt. “DeepSeek-R1’s Bold Bet on Reinforcement Learning: How it Outpaced OpenAI at 3% of the Cost.” Venture Beat, 25 Jan. 2025, source.

References

- Martens, Bertin. "Why Artificial Intelligence is Creating Fundamental Challenges for Competition Policy." Bruegel, 18 July 2024, source.
- Miller, Katharine. "Covert Racism in AI: How Language Models are Reinforcing Outdated Stereotypes." Human-Centered Artificial Intelligence – Stanford University, 3 Sept. 2024, source.
- Newton, Casey. "AI Companies Hit a Scaling Wall." Platformer, 14 Nov. 2024, source.
- "November 2024." Top 500, source. Accessed 16 May 2025.
- "NVIDIA DGX SuperPod." Nvidia, source. Accessed 16 May 2025.
- Penedo, Guilherme, et al. "FineWeb: Decanting the Web for the Finest Text Data at Scale." Hugging Face, 31 May 2024, source.
- Perrigo, Billy. "Exclusive: OpenAI Used Kenyan Workers on Less Than $2 Per Hour to Make ChatGPT Less Toxic." Time, 18 Jan 2023, source.
- Radford, Alec, et al. "Improving Language Understanding by Generative Pre-Training." OpenAI, 11 June 2018, source.
- Savov, Vlad. "Microsoft Reduces Commitments to CoreWeave Ahead of IPO, FT Reports." Bloomberg, 6 Mar. 2025, source.
- Sevilla, Jaime, et al. "Can AI Scaling Continue Through 2030?" Epoch AI, 20 Aug. 2024, source.
- Tazi, Nouamane, et al. "The Ultra-Scale Playbook: Training LLMs on GPU Clusters." Hugging Face, 19 Feb. 2025, source.
- Werner, John. "Running Out of Data: It Could Be a Concern." Forbes, 4 Nov. 2024, source.
- Wiggers, Kyle. "'Open' AI Model Licenses Often Carry Concerning Restrictions." TechCrunch, 14 Mar. 2025, source.
- "OpenAI Teams Up with SoftBank and Oracle on $500B Data Center Project." TechCrunch, 21 Jan. 2025, source.
- Witzenberger, Kevin, and Michael Richardson. "Microsoft Cuts Data Centre Plans and Hikes Prices in Push to Make Users Carry AI Costs." The Conversation, 2 Mar. 2025, source.
- Zitron, Edward. "There is No AI Revolution." Where's Your Ed At, 24 Feb. 2025, source.