O'REILLY®

Large Language Models

The Hard Parts

Open Source AI Solutions for Common Pitfalls

Thársis T. P. Souza &
Jonathan K. Regenstein, Jr.
Foreword by Simon Guest

"Written by two leading practitioners, this indispensable guide equips readers at every skill level with the tools to navigate LLMs successfully."

Sudheer Chava, Alton M. Costley Chair and director, Financial Services Innovation Lab, Georgia Tech

"This book delivers a thoughtful and practically grounded treatment of the challenges that truly matter in real-world LLM systems."

Tomaso Aste, professor of complexity science, University College London

Large Language Models: The Hard Parts

Large language models (LLMs) have transformed natural language processing, but deploying them in applications introduces numerous technical challenges. *Large Language Models: The Hard Parts* offers a clear, practical examination of the limitations developers and AI engineers face when building LLM-based applications. With a focus on implementation pitfalls (not just capabilities), this book provides actionable strategies supported by reproducible Python code and open source tools.

Readers will learn how to navigate key obstacles in application evaluation, input management, testing, and safety. Designed for builders and technical product leads, this guide emphasizes practical solutions to real-world problems and promotes a grounded understanding of LLM constraints and trade-offs.

- **Design testing and evaluation strategies for nondeterministic systems**
- **Manage context, RAG, and long-context retrieval**
- **Address output inconsistency and structural unreliability**
- **Implement safety and content moderation frameworks**
- **Explore alignment challenges and mitigation techniques**
- **Leverage open source models locally**

Thársis T. P. Souza is a computer scientist, author, and product leader focused on AI-driven products. He has held product leadership roles at some of Wall Street's largest hedge funds and at early-stage Silicon Valley startups. He holds a PhD in computer science from UCL, University of London.

Jonathan K. Regenstein, Jr., has spent his career working at the intersection of data, machine learning, technology, and asset management. He is a research affiliate at Georgia Tech's Financial Services Innovation Lab and an advisor to early-stage AI companies. He holds a BA from Harvard University and a JD from NYU School of Law.

DATA

US $79.99 CAN $99.99
ISBN: 979-8-341-62252-4

O'REILLY®

Praise for *Large Language Models: The Hard Parts*

LLM-based applications are easy to demo but hard to take to production. This book focuses on that gap. It explains where these systems fail (hallucinations, evaluations, alignment) and how to think about building reliable (enterprise-ready) applications. It helped me in how we approach building an agentic financial workspace for institutions.

—Didier Rodrigues Lopes, founder and CEO, OpenBB

This book focuses on the challenges that truly matter in real-world LLM systems, providing insights that are valuable in practice as well as from a broader academic perspective. It delivers a thoughtful and practically grounded treatment of the "messy layer" that most books avoid, with clear insight into alignment, evaluation, and deployment.

—Tomaso Aste, professor of complexity science, University College London

Despite remarkable progress in recent years, deploying LLMs in real-world business applications remains fraught with challenges and pitfalls. Written by two leading practitioners, this indispensable guide equips readers at every skill level with the tools to navigate them successfully.

—Sudheer Chava, Alton M. Costley Chair and director, Financial Services Innovation Lab, Georgia Tech

Large Language Models: The Hard Parts

Open Source AI Solutions for Common Pitfalls

Thársis T. P. Souza and Jonathan K. Regenstein, Jr.

Foreword by Simon Guest

O'REILLY®

Large Language Models: The Hard Parts
by Thársis T. P. Souza and Jonathan K. Regenstein, Jr.

Published by O'Reilly Media, Inc., 141 Stony Circle, Suite 195, Santa Rosa, CA 95401.

O'Reilly books may be purchased for educational, business, or sales promotional use. Online editions are also available for most titles (*https://oreilly.com*). For more information, contact our corporate/institutional sales department: 800-998-9938 or *corporate@oreilly.com*.

Acquisitions Editor: Nicole Butterfield
Development Editor: Jeff Bleiel
Production Editor: Clare Laylock
Copyeditor: Piper Content Partners
Proofreader: Audrey Doyle
Indexer: Ellen Troutman-Zaig
Cover Designer: Susan Brown
Cover Illustrator: José Marzan Jr.
Interior Designer: David Futato
Interior Illustrator: Kate Dullea

May 2026: First Edition

Revision History for the First Edition
2026-05-06: First Release

See *https://oreilly.com/catalog/errata.csp?isbn=9798341622524* for release details.

979-8-341-62252-4

[LSI]

To Yoko and Lioto—my knowledge-seeking inspiration
—Thársis

To Bea, and to Olivia, Roxanne, and Eloisa, and to my parents
—Jonathan

Table of Contents

Foreword

In 2024, we started working on our first GenAI curriculum at Code.org. We had an ambitious goal: instead of simply introducing K-12 students to ChatGPT (or an interface that wrapped ChatGPT), we wanted to develop a small language model that they could experiment with more directly.

We believed this approach would have several benefits. First, we felt that a smaller, open source model would give us more control over hosting, pricing, and model versions. We also believed that a smaller model would hallucinate more frequently. While this might sound like an odd requirement, frequent hallucinations offer unique teaching opportunities in the classroom. Finally, we imagined a world where students could learn how to fine-tune these models using their own training data.

This approach of using small language models, however, wasn't without its challenges. With frequent hallucinations, we had to set a high safety bar for the students, creating similar safeguards to the ones available with frontier models. We also wanted to accurately evaluate our small models to prevent regressions on the learning objectives we had set out in the curriculum. Finally, we wanted to create a framework where we could implement many of the capabilities found in more sophisticated systems, such as RAG (retrieval augmented generation) and multimodal input, but make it accessible for middle school and high school students.

It was in the middle of this work that I was introduced to Thársis Souza. Over the course of the following year, and with Thársis's advice and mentoring, the Code.org team successfully implemented and shipped a set of small, student-focused models based on Mistral 7B. Since their release, these models have been used by tens of thousands of students who have created hundreds of thousands of in-classroom conversations.

Fast-forwarding to today, I'm delighted to see that Thársis has continued to build on our learnings and that he and coauthor Jonathan Regenstein have written this valuable book.

As the title suggests, I recommend thinking about this book as a guide to the complex areas beyond just getting an LLM up and running. In the first few chapters, you'll learn the first principles of building with LLMs, methodologies for evaluating these nondeterministic models, real-world strategies for using RAG, and how to generate predictable, structured output.

All of these were critical areas for Code.org as we started to scale our models for our students. Having an automated, well-defined evaluation strategy was particularly crucial. Not only did it improve our confidence when we upgraded or tested different models, but it also enabled our product management team to replace much of the manual testing we had been doing up to that point.

The second half of the book pivots naturally into the critical area of safety, covering both inadvertent and adversarial risks and tackling the challenging topic of testing models for compliance with a safety policy. When we were developing our safety strategy, we developed safety datasets, tested many different filters, and implemented a custom judge validator, each of which you'll find described in detail within these chapters.

Rounding out the book, Thársis and Jonathan explore the importance of alignment for maintaining a consistent model voice. They bring clarity to many acronyms you've likely heard of but may have been unsure of, and they cover alignment techniques and approaches that you can implement for your own fine-tuned models.

While many of these topics have been covered at a high level in other works, Thársis and Jonathan deliver a hands-on perspective with the right balance of code samples and case studies that you can apply to your own projects.

In closing, although the book you have in your hands is titled *The Hard Parts*, in my experience it's really about getting LLM deployment right. In an educational context, the stakes are critically high—especially when introducing language models to students in what may be an unsupervised or semisupervised setting. The stakes are likely just as high for your own projects, and I believe you'll enjoy this book as much as I enjoyed collaborating with Thársis.

— Simon Guest,
Computer science and AI educator,
Adjunct professor at DigiPen Institute of Technology,
Former CTO at Code.org,
Seattle, April 2026

Preface

An alternative title for this book could have been *LLMs Behaving Badly*. If you come from a background in financial modeling, you may have noticed the parallel with Emanuel Derman's seminal work *Models. Behaving. Badly*.[1] Just as Derman cautioned against treating financial models as perfect representations of reality, this book aims to highlight the limitations and pitfalls of large language models (LLMs) in practical applications.

Like financial models that failed to capture the complexity of human behavior and market dynamics, LLMs have inherent constraints. They can hallucinate facts, struggle with logical reasoning, and fail to maintain consistency in long outputs. Their responses, while often convincing, are probabilistic approximations based on training data rather than true understanding, even though humans insist on treating them as "machines that can reason."

In recent years, LLMs have emerged as a transformative force in technology. From ChatGPT and Gemini to Claude and Mistral, these systems have captured the public's imagination and sparked a gold rush of AI-powered applications. However, beneath this technological revolution lies a complex landscape of challenges that developers, data scientists, and technical leaders must navigate.

We wrote this book because we're optimistic about the power and possibilities of LLMs but realistic about how hard it is to deploy them successfully, widely, and reliably. This book focuses on bringing awareness to key LLM challenges and harnessing open source solutions to overcome them. It offers a critical perspective on implementation, backed by practical and reproducible Python examples. By the book's end, readers will be armed with the open source tools they need to become more than users of LLMs, and those tools will put us in a position to thrive in a world that is changing rapidly and offering the chance to move and build very quickly.

1 Emanuel Derman, *Models. Behaving. Badly: Why Confusing Illusion with Reality Can Lead to Disaster, on Wall Street and in Life* (Free Press, 2011).

The skills in this book—evaluation, safety, context, alignment, structured outputs, retrieval augmented generation (RAG), fine-tuning—are what differentiate "I played with LLMs" from "I built a reliable LLM-based application that my organization depends on."

Who Should Read This Book

We wrote this book for data scientists, software engineers, and professionals across many fields. We list here some of those who we think will benefit most, but in reality we think anyone who wants to work with LLMs should read this book. More specifically:

- *Data scientists and machine learning (ML) engineers* who've worked with traditional ML but are now tasked with integrating LLMs into production systems.
- *Product managers and technical leaders* who need to make informed decisions about whether, when, and how to use LLMs. You're responsible for outcomes.
- *Researchers and domain experts* in fields like finance, health care, law, or education—those who see potential for LLMs in their domain but need to understand how to validate their outputs. You know your domain inside out but may be skeptical of the AI hype. This book shows you how to rigorously evaluate whether an LLM aligns with your expertise.
- *Analytics and business intelligence professionals* exploring how LLMs can augment data analysis, report generation, or insight discovery. You're comfortable with data, but LLMs are a new paradigm.
- *Anyone tasked with "figuring out how we use AI"*—an ever-growing group. Whether you're in operations, HR, marketing, legal, or elsewhere, organizations are asking people without ML backgrounds to explore LLM applications. This book won't make you an expert at building LLMs, but it will make you literate enough around LLM challenges to evaluate vendors, assess risks, understand what's possible versus what's hype, and have informed conversations with technical teams about implementation.

Navigating This Book by Chapter

The most difficult part of this book was deciding what not to cover in an ever-expanding world. Here's what we settled on:

Chapter 1, "First Principles: What to Consider Before We Start Building with LLMs"
: We begin by addressing questions around value, data, stakeholders, and strategy that determine whether an LLM-based application succeeds or fails. This chapter establishes that technical excellence isn't enough without strategic clarity about what we're building and why.

Chapter 2, "The Evals Gap"

We tackle the challenge of measuring quality in nondeterministic systems. We start with traditional metrics like Bilingual Evaluation Understudy (BLEU) and Recall-Oriented Understudy for Gisting Evaluation (ROUGE) that measure textual similarity, understanding both their utility and their limitations. We then explore LLM-as-a-judge, learning how to use one LLM to assess another's outputs.

Chapter 3, "Evaluation Tools for LLM-Based Applications"

We move from evaluation concepts to practical tools, exploring three tools that help us systematically test and improve our LLM-based applications (LLMBAs): LangSmith for end-to-end evaluation and experiment tracking, Promptfoo for rapid prompt comparison and adversarial testing, and LightEval for lightweight econometric evaluation.

Chapter 4, "From Data to Context"

We next delve into what happens when unstructured data becomes context for LLMs—or, in other words, how to give LLMs access to information beyond their training data. We learn the mechanics of chunking strategies, vector embeddings, RAG, and long-context window models. Will RAG survive, or will everything shift to long-context models in the future?

Chapter 5, "Structured Data Output"

We explore the challenge of getting reliable, structured outputs from LLMs, using JSON mode constraint, Pydantic schemas to enforce specific structures with type validation, and logit postprocessing to prove mathematical guarantees by manipulating token probabilities.

Chapter 6, "LLM Safety Considerations"

In this chapter with no code, we establish a conceptual foundation for LLM safety before implementing any technical solutions. We explore general AI safety principles, examine LLM-specific vulnerabilities like jailbreaking, and study how Anthropic, OpenAI, and Google approach safety through different frameworks. We learn what red teaming means in practice and how Constitutional AI systematically embeds safety principles.

Chapter 7, "Evaluating LLMs for Safety"

We translate safety concepts into code, starting with evaluation benchmarks like SALAD-Bench, TruthfulQA, and HarmBench that measure how safe and truthful our models actually are. We then implement runtime protection through guardrails like Llama Guard and NeMo Guardrails that filter harmful content in real time, and moderation application programming interfaces (APIs) from OpenAI and Mistral that provide hosted safety classification.

Chapter 8, "LLM Alignment: A Case Study"
: We focus on a case study to learn how to reshape model behavior through preference-based alignment. We master the mechanics of Direct Preference Optimization (DPO) and use practical tools like Transformer Reinforcement Learning (TRL) to fine-tune a model in alignment with a policy. Then we evaluate our own work using an LLM-as-a-judge.

Chapter 9, "Epilogue: LLMBAs in the Era of Falling Costs"
: We conclude by examining the dramatic collapse in inference costs and its implications for everything we've learned. We explore what this means for data professionals whose work shifts from optimizing for cost to optimizing for quality and for organizations where competitive advantage comes from rigorous evaluation and safety rather than just access to models.

Conventions Used in This Book

The following typographical conventions are used in this book:

Italic
: Indicates new terms, URLs, email addresses, filenames, and file extensions.

`Constant width`
: Used for program listings, as well as within paragraphs to refer to program elements such as variable or function names, databases, data types, environment variables, statements, and keywords.

`Constant width bold`
: Shows commands or other text that should be typed literally by the user.

`Constant width italic`
: Shows text that should be replaced with user-supplied values or by values determined by context.

This element signifies a general note.

This element indicates a warning or caution.