O'REILLY®

AI Engineering

Building Applications with Foundation Models

Chip Huyen

Praise for *AI Engineering*

This book offers a comprehensive, well-structured guide to the essential aspects of building generative AI systems. A must-read for any professional looking to scale AI across the enterprise.

—Vittorio Cretella, former global CIO, P&G and Mars

Chip Huyen gets generative AI. On top of that, she is a remarkable teacher and writer whose work has been instrumental in helping teams bring AI into production. Drawing on her deep expertise, AI Engineering serves as a comprehensive and holistic guide, masterfully detailing everything required to design and deploy generative AI applications in production.

—Luke Metz, cocreator of ChatGPT, former research manager at OpenAI

Every AI engineer building real-world applications should read this book. It's a vital guide to end-to-end AI system design, from model development and evaluation to large-scale deployment and operation.

—Andrei Lopatenko, Director Search and AI, Neuron7

This book serves as an essential guide for building AI products that can scale. Unlike other books that focus on tools or current trends that are constantly changing, Chip delivers timeless foundational knowledge. Whether you're a product manager or an engineer, this book effectively bridges the collaboration gap between cross-functional teams, making it a must-read for anyone involved in AI development.

—Aileen Bui, AI Product Operations Manager, Google

This is the definitive segue into AI engineering from one of the greats of ML engineering! Chip has seen through successful projects and careers at every stage of a company and for the first time ever condensed her expertise for new AI Engineers entering the field.

—swyx, Curator, AI.Engineer

AI Engineering is a practical guide that provides the most up-to-date information on AI development, making it approachable for novice and expert leaders alike. This book is an essential resource for anyone looking to build robust and scalable AI systems.

—Vicki Reyzelman, Chief AI Solutions Architect,
Mave Sparks

AI Engineering is a comprehensive guide that serves as an essential reference for both understanding and implementing AI systems in practice.

—Han Lee, Director—Data Science, Moody's

AI Engineering is an essential guide for anyone building software with Generative AI! It demystifies the technology, highlights the importance of evaluation, and shares what should be done to achieve quality before starting with costly fine-tuning.

—Rafal Kawala, Senior AI Engineering Director, 16 years of experience working in a Fortune 500 company

AI Engineering

Building Applications with Foundation Models

Chip Huyen

AI Engineering

by Chip Huyen

Printed in the United States of America.

Published by O’Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472.

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

Acquisitions Editor: Nicole Butterfield	Indexer: WordCo Indexing Services, Inc.
Development Editor: Melissa Potter	Interior Designer: David Futato
Production Editor: Beth Kelly	Cover Designer: Karen Montgomery

Copyeditor: Liz Wheeler Illustrator: Kate Dullea

Proofreader: Piper Editorial Consulting, LLC

- December 2024: First Edition

Revision History for the First Edition

- 2024-12-04: First Release

See *http://oreilly.com/catalog/errata.csp?isbn=9781098166304* for release details.

978-1-098-16630-4

[LSI]

Preface

When ChatGPT came out, like many of my colleagues, I was disoriented. What surprised me wasn't the model's size or capabilities. For over a decade, the AI community has known that scaling up a model improves it. In 2012, the AlexNet authors noted in their landmark paper that: "All of our experiments suggest that our results can be improved simply by waiting for faster GPUs and bigger datasets to become available."[1, 2]

What surprised me was the sheer number of applications this capability boost unlocked. I thought a small increase in model quality metrics might result in a modest increase in applications. Instead, it resulted in an explosion of new possibilities.

Not only have these new AI capabilities increased the demand for AI applications, but they have also lowered the entry barrier for developers. It's become so easy to get started with building AI applications. It's even possible to build an application without writing a single line of code. This shift has transformed AI from a specialized discipline into a powerful development tool everyone can use.

Even though AI adoption today seems new, it's built upon techniques that have been around for a while. Papers about language modeling came out as early as the 1950s. Retrieval-augmented generation (RAG) applications are built upon retrieval technology that has powered search and recommender systems since long before the term RAG was coined. The best practices for deploying

traditional machine learning applications—systematic experimentation, rigorous evaluation, relentless optimization for faster and cheaper models—are still the best practices for working with foundation model-based applications.

The familiarity and ease of use of many AI engineering techniques can mislead people into thinking there is nothing new to AI engineering. But while many principles for building AI applications remain the same, the scale and improved capabilities of AI models introduce opportunities and challenges that require new solutions.

This book covers the end-to-end process of adapting foundation models to solve real-world problems, encompassing tried-and-true techniques from other engineering fields and techniques emerging with foundation models.

I set out to write the book because I wanted to learn, and I did learn a lot. I learned from the projects I worked on, the papers I read, and the people I interviewed. During the process of writing this book, I used notes from over 100 conversations and interviews, including researchers from major AI labs (OpenAI, Google, Anthropic, ...), framework developers (NVIDIA, Meta, Hugging Face, Anyscale, LangChain, LlamaIndex, ...), executives and heads of AI/data at companies of different sizes, product managers, community researchers, and independent application developers (see “Acknowledgments”).

I especially learned from early readers who tested my assumptions, introduced me to different perspectives, and exposed me to new problems and approaches. Some sections of the book have also received thousands of comments from the

community after being shared on my blog, many giving me new perspectives or confirming a hypothesis.

I hope that this learning process will continue for me now that the book is in your hands, as you have experiences and perspectives that are unique to you. Please feel free to share any feedback you might have for this book with me via X, LinkedIn, or email at hi@huyenchip.com.

What This Book Is About

This book provides a framework for adapting foundation models, which include both large language models (LLMs) and large multimodal models (LMMs), to specific applications.

There are many different ways to build an application. This book outlines various solutions and also raises questions you can ask to evaluate the best solution for your needs. Some of the many questions that this book can help you answer are:

- Should I build this AI application?
- How do I evaluate my application? Can I use AI to evaluate AI outputs?
- What causes hallucinations? How do I detect and mitigate hallucinations?
- What are the best practices for prompt engineering?
- Why does RAG work? What are the strategies for doing RAG?
- What's an agent? How do I build and evaluate an agent?
- When to finetune a model? When not to finetune a model?
- How much data do I need? How do I validate the quality of my data?
- How do I make my model faster, cheaper, and secure?
- How do I create a feedback loop to improve my application continually?

The book will also help you navigate the overwhelming AI landscape: types of models, evaluation benchmarks, and a seemingly infinite number of use cases and application patterns.

The content in this book is illustrated using case studies, many of which I worked on, backed by ample references and extensively reviewed by experts from a wide range of backgrounds. Although the book took two years to write, it draws from my experience working with language models and ML systems from the last decade.

Like my previous O'Reilly book, *Designing Machine Learning Systems* (DMLS), this book focuses on the fundamentals of AI engineering instead of any specific tool or API. Tools become outdated quickly, but fundamentals should last longer.[3]

READING *AI ENGINEERING* (AIE) WITH *DESIGNING MACHINE LEARNING SYSTEMS* (DMLS)

AIE can be a companion to DMLS. DMLS focuses on building applications on top of traditional ML models, which involves more tabular data annotations, feature engineering, and model training. AIE focuses on building applications on top of foundation models, which involves more prompt engineering, context construction, and parameter-efficient finetuning. Both books are self-contained and modular, so you can read either book independently.

Since foundation models are ML models, some concepts are relevant to working with both. If a topic is relevant to AIE but has been discussed extensively in DMLS, it'll still be covered in this book, but to a lesser extent, with pointers to relevant resources.

Note that many topics are covered in DMLS but not in AIE, and vice versa. The first chapter of this book also covers the differences between traditional ML engineering and AI engineering. A real-world system often involves both traditional ML models and foundation models, so knowledge about working with both is often necessary.

Determining whether something will last, however, is often challenging. I relied on three criteria. First, for a problem, I determined whether it results from the fundamental limitations of how AI works or if it'll go away with better models. If a problem is fundamental, I'll analyze its challenges and solutions to address each challenge. I'm a fan of the start-simple approach, so for many problems,

I'll start from the simplest solution and then progress with more complex solutions to address rising challenges.

Second, I consulted an extensive network of researchers and engineers, who are smarter than I am, about what they think are the most important problems and solutions.

Occasionally, I also relied on Lindy's Law, which infers that the future life expectancy of a technology is proportional to its current age. So if something has been around for a while, I assume that it'll continue existing for a while longer.

In this book, however, I occasionally included a concept that I believe to be temporary because it's immediately useful for some application developers or because it illustrates an interesting problem-solving approach.

What This Book Is Not

This book isn't a tutorial. While it mentions specific tools and includes pseudocode snippets to illustrate certain concepts, it doesn't teach you how to use a tool. Instead, it offers a framework for selecting tools. It includes many discussions on the trade-offs between different solutions and the questions you should ask when evaluating a solution. When you want to use a tool, it's usually easy to find tutorials for it online. AI chatbots are also pretty good at helping you get started with popular tools.

This book isn't an ML theory book. It doesn't explain what a neural network is or how to build and train a model from scratch. While it explains many theoretical concepts immediately relevant to the discussion, the book is a practical book that focuses on helping you build successful AI applications to solve real-world problems.

While it's possible to build foundation model-based applications without ML expertise, a basic understanding of ML and statistics can help you build better applications and save you from unnecessary suffering. You can read this book without any prior ML background. However, you will be more effective while building AI applications if you know the following concepts:

- Probabilistic concepts such as sampling, determinism, and distribution.
- ML concepts such as supervision, self-supervision, log-likelihood, gradient descent, backpropagation, loss function, and hyperparameter tuning.

- Various neural network architectures, including feedforward, recurrent, and transformer.
- Metrics such as accuracy, F1, precision, recall, cosine similarity, and cross entropy.

If you don’t know them yet, don’t worry—this book has either brief, high-level explanations or pointers to resources that can get you up to speed.

Who This Book Is For

This book is for anyone who wants to leverage foundation models to solve real-world problems. This is a technical book, so the language of this book is geared toward technical roles, including AI engineers, ML engineers, data scientists, engineering managers, and technical product managers. This book is for you if you can relate to one of the following scenarios:

- You're building or optimizing an AI application, whether you're starting from scratch or looking to move beyond the demo phase into a production-ready stage. You may also be facing issues like hallucinations, security, latency, or costs, and need targeted solutions.
- You want to streamline your team's AI development process, making it more systematic, faster, and reliable.
- You want to understand how your organization can leverage foundation models to improve the business's bottom line and how to build a team to do so.

You can also benefit from the book if you belong to one of the following groups:

- Tool developers who want to identify underserved areas in AI engineering to position your products in the ecosystem.
- Researchers who want to better understand AI use cases.
- Job candidates seeking clarity on the skills needed to pursue a career as an AI engineer.

- Anyone wanting to better understand AI's capabilities and limitations, and how it might affect different roles.

I love getting to the bottom of things, so some sections dive a bit deeper into the technical side. While many early readers like the detail, it might not be for everyone. I'll give you a heads-up before things get too technical. Feel free to skip ahead if it feels a little too in the weeds!