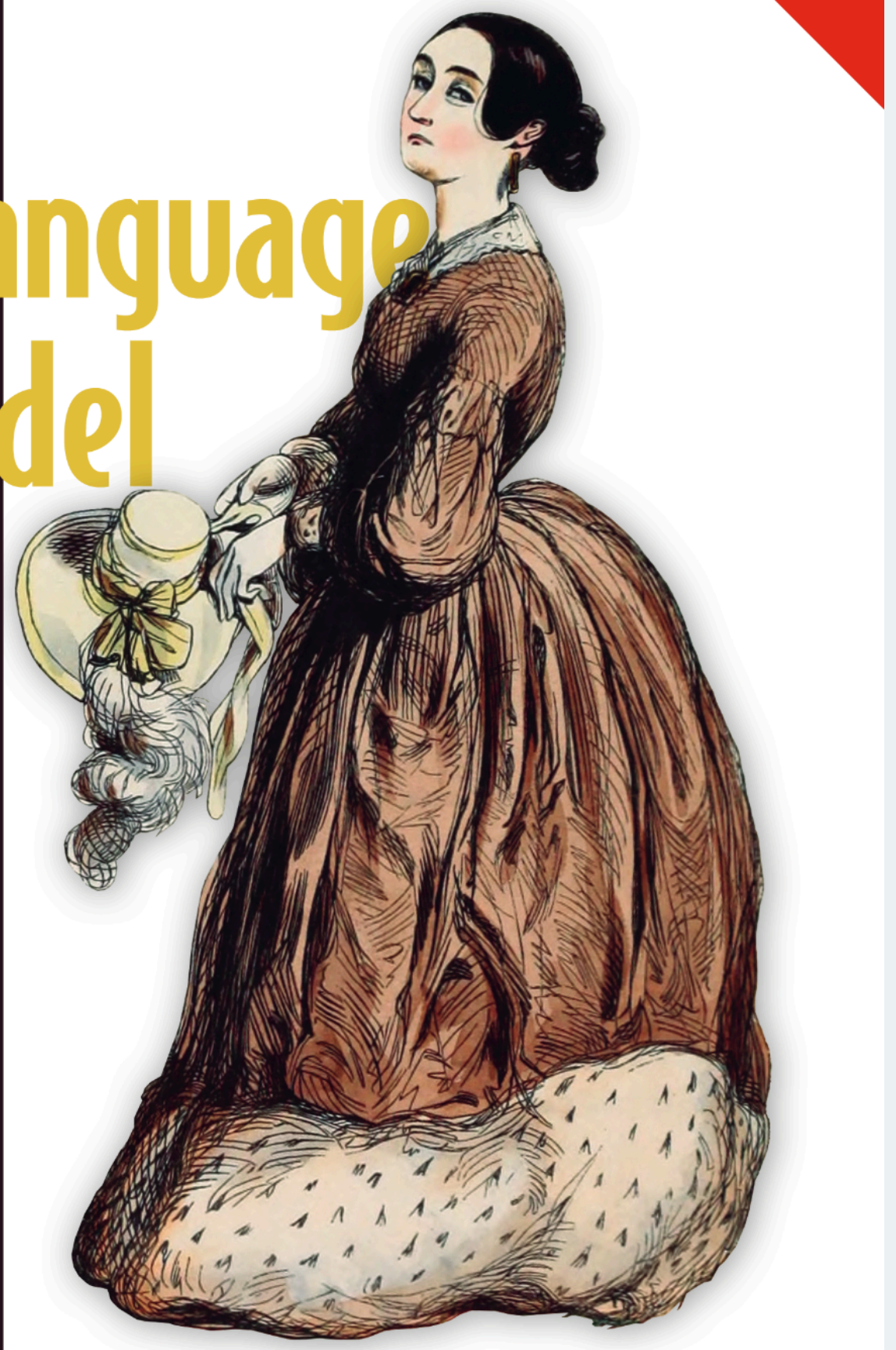
FROM SCRATCH

BUILD A

Large Language Model



Sebastian Raschka

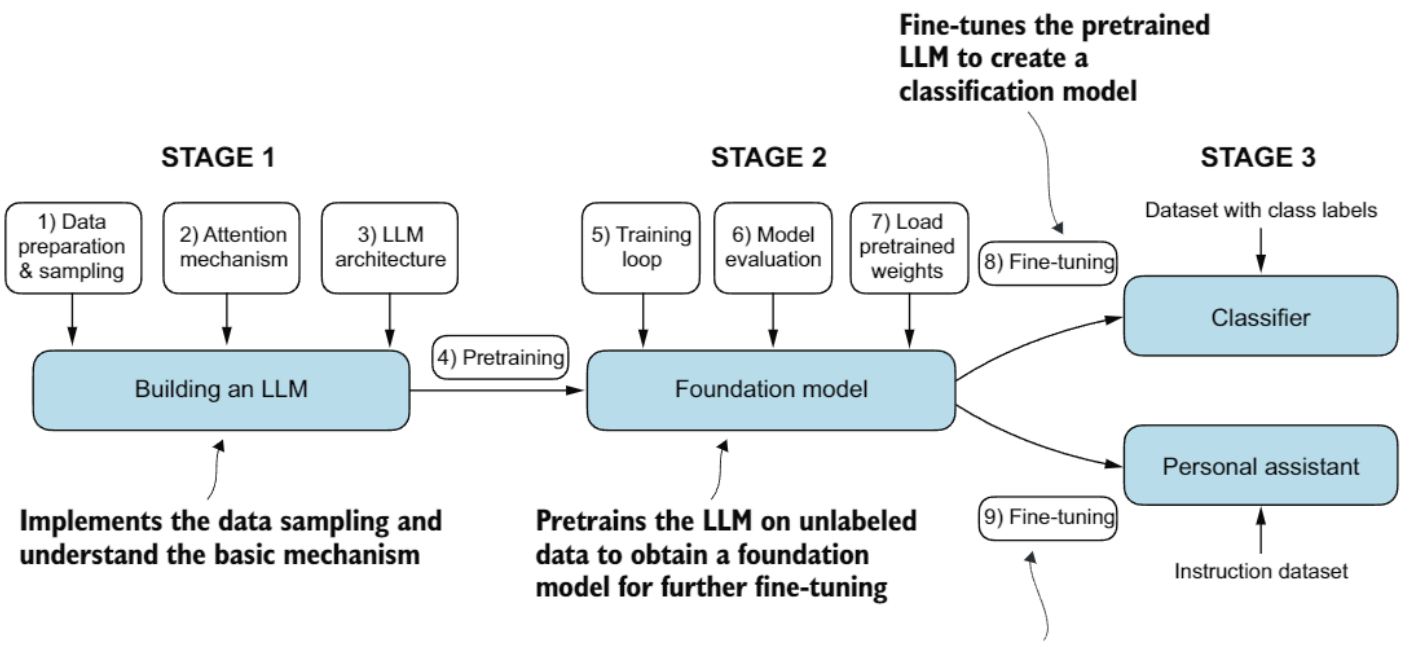
MANNING

FY
涂
鸦

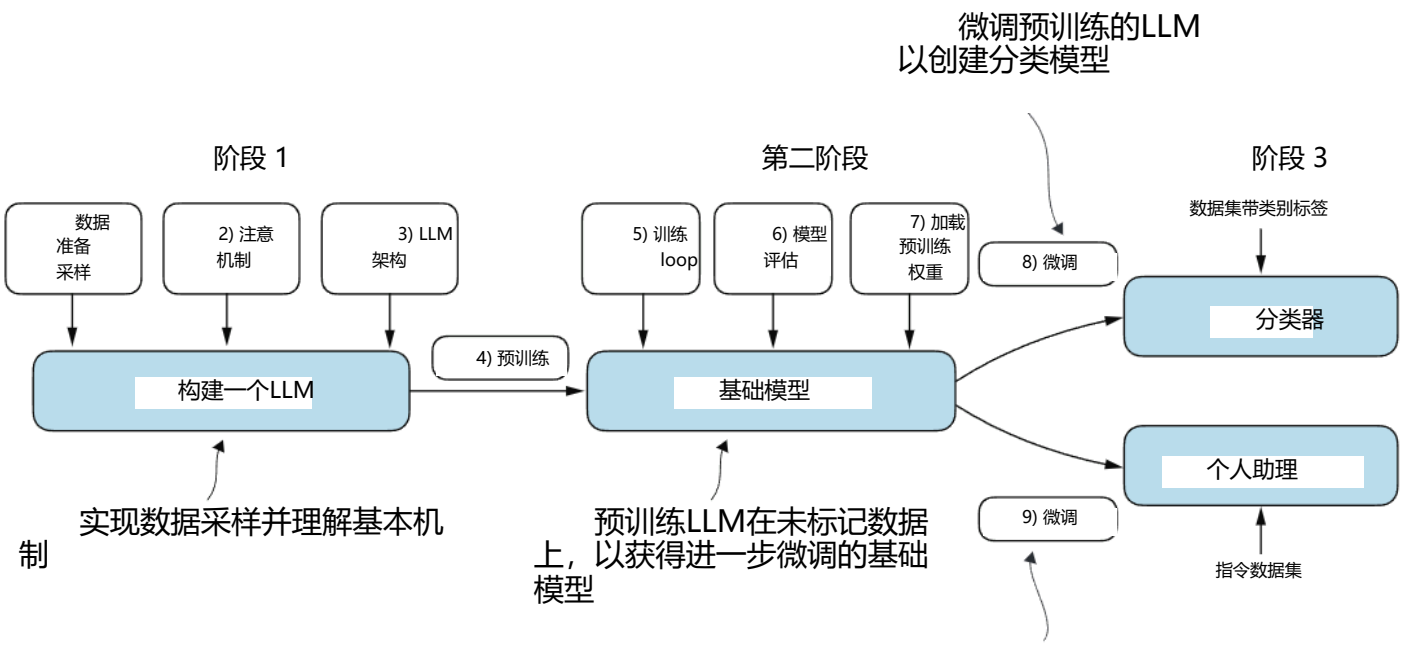
构建一个 Large Language Model

塞巴斯蒂安·拉斯卡

曼宁



The three main stages of coding a large language model (LLM) are implementing the LLM architecture and data preparation process (stage 1), pretraining an LLM to create a foundation model (stage 2), and fine-tuning the foundation model to become a personal assistant or text classifier (stage 3). Each of these stages is explored and implemented in this book.



大型语言模型（LLM）编码的三个主要阶段是实施LLM架构和数据准备过程（阶段 1）、预训练LLM以创建基础模型（阶段 2）以及微调基础模型以成为个人助理或文本分类器（阶段 3）。本书探讨了并实现了这些阶段。

Build a Large Language Model (From Scratch)

构建大型语言模型（从零开始）

Build a Large Language Model (From Scratch)

SEBASTIAN RASCHKA

MANNING
SHELTER ISLAND

构建大型语言模型（从零开始）

塞巴斯蒂安·拉斯卡

曼宁
避难岛

For online information and ordering of this and other Manning books, please visit www.manning.com. The publisher offers discounts on this book when ordered in quantity. For more information, please contact

Special Sales Department
Manning Publications Co.
20 Baldwin Road
PO Box 761
Shelter Island, NY 11964
Email: orders@manning.com

Manning Publications Co.
20 Baldwin Road
PO Box 761
Shelter Island, NY 11964

Development editor:	Dustin Archibald
Technical editor:	David Caswell
Review editor:	Kishor Rit
Production editor:	Aleksandar Dragosavljević
Copy editors:	Kari Lucke and Alisa Larson
Proofreader:	Mike Beady
Technical proofreader:	Jerry Kuch
Typesetter:	Dennis Dalinnik
Cover designer:	Marija Tudor

ISBN: 9781633437166
Printed in the United States of America

请访问 www.manning.com 获取在线信息和订购此书及其他 Manning 书籍。当批量订购时，出版社对此书提供折扣。如需更多信息，请联系

特别销售部 曼宁出版社公司 20 巴德温路 邮政信箱 761 舍利岛，纽约 11964 邮箱：orders@manning.com

认识到保护已撰写内容的重要性，Manning 的政策是将我们出版的书籍印刷在无酸纸上，并为此付出最大努力。同时，也认识到我们有责任保护我们星球上的资源，Manning 的书籍至少有 15%的纸张是回收的，并且没有使用元素氯进行加工。

作者和出版社已尽最大努力确保本书在印刷时信息准确。作者和出版社不承担任何因错误或遗漏造成的损失、损害或中断的责任，无论这些错误或遗漏是由于疏忽、事故或其他原因，或是对本书中信息的任何使用造成的。

曼宁出版社公司
20 巴勒斯坦路
邮政信箱 761
避风岛，纽约 11964

开发编辑器：达斯汀·阿奇博尔德
技术编辑：大卫·卡斯韦尔
审稿编辑：Kishor Rit
生产编辑：Aleksandar Dragosavljevic´
校对编辑：Kari Lucke 和 Alisa Larson 校对：Mike Beady 技术校对：Jerry Kuch

排版师：Dennis Dalinnik 封面设计师：Marija Tudor

ISBN: 9781633437166 美国印刷

brief contents

简要内容

contents

内容

preface

I've always been fascinated with language models. More than a decade ago, my journey into AI began with a statistical pattern classification class, which led to my first independent project: developing a model and web application to detect the mood of a song based on its lyrics.

Fast forward to 2022, with the release of ChatGPT, large language models (LLMs) have taken the world by storm and have revolutionized how many of us work. These models are incredibly versatile, aiding in tasks such as checking grammar, composing emails, summarizing lengthy documents, and much more. This is owed to their ability to parse and generate human-like text, which is important in various fields, from customer service to content creation, and even in more technical domains like coding and data analysis.

As their name implies, a hallmark of LLMs is that they are "large"—very large—encompassing millions to billions of parameters. (For comparison, using more traditional machine learning or statistical methods, the Iris flower dataset can be classified with more than 90% accuracy using a small model with only two parameters.) However, despite the large size of LLMs compared to more traditional methods, LLMs don't have to be a black box.

In this book, you will learn how to build an LLM one step at a time. By the end, you will have a solid understanding of how an LLM, like the ones used in ChatGPT, works on a fundamental level. I believe that developing confidence with each part of the fundamental concepts and underlying code is crucial for success. This not only

前言

我一直对语言模型很着迷。十多年前，我的 AI 之旅始于一门统计模式分类课程，这让我开始了我的第一个独立项目：开发一个模型和 Web 应用程序，根据歌词检测歌曲的情绪。

2022 年，随着 ChatGPT 的发布，大型语言模型（LLMs）席卷全球，并彻底改变了我们许多人工作的方式。这些模型非常灵活，可以帮助检查语法、撰写电子邮件、总结长篇文档等等。这得益于它们解析和生成类似人类文本的能力，这在各个领域都至关重要，从客户服务到内容创作，甚至在更技术性的领域如编码和数据分析也是如此。

正如其名所示，LLMs的一个显著特点是它们“很大”——非常大——包含数百万到数十亿个参数。（相比之下，使用更传统的机器学习或统计方法，使用只有两个参数的小型模型，Iris 花数据集可以以超过 90%的准确率进行分类。）然而，尽管LLMs与更传统的方法相比规模很大，LLMs不必是一个黑盒。

在这本书中，您将学习如何一步一步地构建一个LLM。到结束时，您将深入理解像 ChatGPT 中使用的LLM在基本层面的工作原理。我相信，对基本概念和底层代码的每个部分建立信心对于成功至关重要。这不仅

helps in fixing bugs and improving performance but also enables experimentation with new ideas.

Several years ago, when I started working with LLMs, I had to learn how to implement them the hard way, sifting through many research papers and incomplete code repositories to develop a general understanding. With this book, I hope to make LLMs more accessible by developing and sharing a step-by-step implementation tutorial detailing all the major components and development phases of an LLM.

I strongly believe that the best way to understand LLMs is to code one from scratch—and you'll see that this can be fun too!

Happy reading and coding!

这不仅有助于修复错误和提升性能，还使人们能够尝试新想法。

几年前，当我开始与LLMs合作时，我不得不通过艰难的方式学习如何实现它们，通过筛选许多研究论文和不完整的代码库来发展一个一般性的理解。通过这本书，我希望通过开发和分享一个逐步实现教程，详细说明LLM的所有主要组件和开发阶段，使LLMs更加易于理解。

我坚信，理解LLMs的最好方式是从零开始编写代码——你会发现这也很有趣！祝阅读愉快，编码愉快！

acknowledgments

Writing a book is a significant undertaking, and I would like to express my sincere gratitude to my wife, Liza, for her patience and support throughout this process. Her unconditional love and constant encouragement have been absolutely essential.

I am incredibly grateful to Daniel Kleine, whose invaluable feedback on the in-progress chapters and code went above and beyond. With his keen eye for detail and insightful suggestions, Daniel's contributions have undoubtedly made this book a smoother and more enjoyable reading experience.

I would also like to thank the wonderful staff at Manning Publications, including Michael Stephens, for the many productive discussions that helped shape the direction of this book, and Dustin Archibald, whose constructive feedback and guidance in adhering to the Manning guidelines have been crucial. I also appreciate your flexibility in accommodating the unique requirements of this unconventional from-scratch approach. A special thanks to Aleksandar Dragosavljević, Kari Lucke, and Mike Beady for their work on the professional layouts and to Susan Honeywell and her team for refining and polishing the graphics.

I want to express my heartfelt gratitude to Robin Campbell and her outstanding marketing team for their invaluable support throughout the writing process.

Finally, I extend my thanks to the reviewers: Anandaganesh Balakrishnan, Anto Aravinth, Ayush Bihani, Bassam Ismail, Benjamin Muskalla, Bruno Sonnino, Christian Prokopp, Daniel Kleine, David Curran, Dibyendu Roy Chowdhury, Gary Pass, Georg Sommer, Giovanni Alzetta, Guillermo Alcántara, Jonathan Reeves, Kunal Ghosh, Nicolas Modrzyk, Paul Silisteanu, Raul Ciotescu, Scott Ling, Sriram Macharla, Sumit

致谢

撰写一本书是一项重大任务，我要衷心感谢我的妻子丽莎在整个过程中的耐心和支持。她无条件的爱和持续的鼓励是至关重要的。

我非常感谢 Daniel Kleine，他对正在进行的章节和代码的宝贵反馈超出了预期。凭借他对细节的敏锐眼光和有见地的建议，Daniel 的贡献无疑使这本书的阅读体验更加流畅和愉快。

我也想感谢 Manning Publications 的出色团队，包括 Michael Stephens，感谢他们许多富有成效的讨论，这些讨论有助于塑造本书的方向，以及 Dustin Archibald，他的建设性反馈和遵循 Manning 指南的指导至关重要。我还赞赏您在适应这种非常规的从头开始方法时的灵活性。特别感谢 Aleksandar Dragosavljevic´、Kari Lucke 和 Mike Beady 在专业布局方面的工作，以及 Susan Honeywell 及其团队在精炼和润色图形方面的工作。

我想对 Robin Campbell 及其卓越的市场营销团队表示衷心的感谢，感谢他们在整个写作过程中给予的无价支持。

最后，我要感谢以下审稿人：Anandaganesh Balakrishnan、Anto Aravinth、Ayush Bihani、Bassam Ismail、Benjamin Muskalla、Bruno Sonnino、Christian Prokopp、Daniel Kleine、David Curran、Dibyendu Roy Chowdhury、Gary Pass、Georg Sommer、Giovanni Alzetta、Guillermo Alcántara、Jonathan Reeves、Kunal Ghosh、Nicolas Modrzyk、Paul Silisteanu、Raul Ciotescu、Scott Ling、Sriram Macharla、Sumit

Pal, Vahid Mirjalili, Vaijanath Rao, and Walter Reade for their thorough feedback on the drafts. Your keen eyes and insightful comments have been essential in improving the quality of this book.

To everyone who has contributed to this journey, I am sincerely grateful. Your support, expertise, and dedication have been instrumental in bringing this book to fruition. Thank you!

Sumit Pal、Vahid Mirjalili、Vaijanath Rao 和 Walter Reade 对草稿的详细反馈。您敏锐的目光和深刻的评论对于提高本书质量至关重要。

对每一位为这次旅程做出贡献的人，我衷心感谢。您的支持、专业知识和奉献精神对于这本书的完成至关重要。谢谢！

about this book

Build a Large Language Model (From Scratch) was written to help you understand and create your own GPT-like large language models (LLMs) from the ground up. It begins by focusing on the fundamentals of working with text data and coding attention mechanisms and then guides you through implementing a complete GPT model from scratch. The book then covers the pretraining mechanism as well as fine-tuning for specific tasks such as text classification and following instructions. By the end of this book, you'll have a deep understanding of how LLMs work and the skills to build your own models. While the models you'll create are smaller in scale compared to the large foundational models, they use the same concepts and serve as powerful educational tools to grasp the core mechanisms and techniques used in building state-of-the-art LLMs.

Who should read this book

Build a Large Language Model (From Scratch) is for machine learning enthusiasts, engineers, researchers, students, and practitioners who want to gain a deep understanding of how LLMs work and learn to build their own models from scratch. Both beginners and experienced developers will be able to use their existing skills and knowledge to grasp the concepts and techniques used in creating LLMs.

What sets this book apart is its comprehensive coverage of the entire process of building LLMs, from working with datasets to implementing the model architecture, pretraining on unlabeled data, and fine-tuning for specific tasks. As of this writing, no

关于这本书

构建大型语言模型（从零开始）旨在帮助您理解和

创建您自己的从零开始的大语言模型（类似于 GPT）（LLMs）。它首先关注处理文本数据和编码注意力机制的基础知识，然后引导您从头开始实现一个完整的 GPT 模型。接着，本书涵盖了预训练机制以及针对特定任务（如文本分类和遵循指令）的微调。到本书结束时，您将深入理解LLMs的工作原理，并具备构建自己模型的能力。虽然您将创建的模型规模较小，但它们使用相同的概念，并作为强大的教育工具，帮助您掌握构建最先进LLMs所使用的核心机制和技术。

谁应该阅读这本书

构建大型语言模型（从零开始）面向机器学习爱好者、工程师、研究人员、学生和实践者，他们希望深入了解LLMs的工作原理，并学习从头开始构建自己的模型。无论是初学者还是有经验的开发者，都能利用他们现有的技能和知识来掌握创建LLMs所使用的概念和技术。

这本书与众不同的地方在于它全面涵盖了构建LLMs的整个过程，从处理数据集到实现模型架构，在未标记数据上进行预训练，以及针对特定任务进行微调。截至本文撰写时，还没有

other resource provides such a complete and hands-on approach to building LLMs from the ground up.

To understand the code examples in this book, you should have a solid grasp of Python programming. While some familiarity with machine learning, deep learning, and artificial intelligence can be beneficial, an extensive background in these areas is not required. LLMs are a unique subset of AI, so even if you're relatively new to the field, you'll be able to follow along.

If you have some experience with deep neural networks, you may find certain concepts more familiar, as LLMs are built upon these architectures. However, proficiency in PyTorch is not a prerequisite. Appendix A provides a concise introduction to PyTorch, equipping you with the necessary skills to comprehend the code examples throughout the book.

A high school–level understanding of mathematics, particularly working with vectors and matrices, can be helpful as we explore the inner workings of LLMs. However, advanced mathematical knowledge is not necessary to grasp the key concepts and ideas presented in this book.

The most important prerequisite is a strong foundation in Python programming. With this knowledge, you'll be well prepared to explore the fascinating world of LLMs and understand the concepts and code examples presented in this book.

How this book is organized: A roadmap

This book is designed to be read sequentially, as each chapter builds upon the concepts and techniques introduced in the previous ones. The book is divided into seven chapters that cover the essential aspects of LLMs and their implementation.

Chapter 1 provides a high-level introduction to the fundamental concepts behind LLMs. It explores the transformer architecture, which forms the basis for LLMs such as those used on the ChatGPT platform.

Chapter 2 lays out a plan for building an LLM from scratch. It covers the process of preparing text for LLM training, including splitting text into word and subword tokens, using byte pair encoding for advanced tokenization, sampling training examples with a sliding window approach, and converting tokens into vectors that feed into the LLM.

Chapter 3 focuses on the attention mechanisms used in LLMs. It introduces a basic self-attention framework and progresses to an enhanced self-attention mechanism. The chapter also covers the implementation of a causal attention module that enables LLMs to generate one token at a time, masking randomly selected attention weights with dropout to reduce overfitting and stacking multiple causal attention modules into a multihead attention module.

Chapter 4 focuses on coding a GPT-like LLM that can be trained to generate human-like text. It covers techniques such as normalizing layer activations to stabilize neural network training, adding shortcut connections in deep neural networks to train models more effectively, implementing transformer blocks to create GPT models

没有其他资源能提供如此全面和实战性的从零开始构建LLMs的方法。

为了理解本书中的代码示例，您应该对 Python 编程有扎实的掌握。虽然对机器学习、深度学习和人工智能有一定的了解可能有益，但在这方面的广泛背景并非必需。LLMs 是人工智能的一个独特子集，因此即使您对这个领域相对较新，也能跟上进度。

如果您对深度神经网络有一定经验，您可能会觉得某些概念更加熟悉，因为LLMs是基于这些架构构建的。然而，熟练掌握 PyTorch 并非先决条件。附录 A 提供了对 PyTorch 的简要介绍，使您具备理解本书中代码示例所需的知识。

高中水平的数学理解，尤其是处理向量和矩阵，在探索LLMs的内部运作时可能很有帮助。然而，掌握本书中提出的关键概念和思想并不需要高级数学知识。

最重要的先决条件是扎实的 Python 编程基础。

凭借这些知识，你将充分准备好去探索LLMs这个迷人的世界，并理解本书中提出的概念和代码示例。

这本书的组织结构：路线图

这本书旨在按顺序阅读，因为每一章都是在前一章介绍的概念和技术基础上构建的。本书分为七个章节，涵盖了LLMs的必要方面及其实现。

第一章提供了对LLMs背后基本概念的概述。它探讨了 Transformer 架构，这是 ChatGPT 平台等使用的LLMs的基础。

第二章概述了从头开始构建LLM的计划。它涵盖了为LLM训练准备文本的过程，包括将文本分割成单词和子词标记，使用字节对编码进行高级标记化，使用滑动窗口方法采样训练示例，以及将标记转换为输入到LLM的向量。

第三章重点介绍了在LLMs中使用的注意力机制。它介绍了一个基本的自注意力框架，并逐步发展到增强的自注意力机制。本章还涵盖了因果注意力模块的实现，该模块使LLMs能够一次生成一个标记，通过随机丢弃选择的注意力权重来减少过拟合，并将多个因果注意力模块堆叠成多头注意力模块。

第四章专注于编写一个类似 GPT 的LLM，可以训练生成类似人类的文本。它涵盖了诸如将层激活归一化以稳定神经网络训练、在深度神经网络中添加快捷连接以更有效地训练模型、实现 Transformer 块以创建 GPT 模型等技术

of various sizes, and computing the number of parameters and storage requirements of GPT models.

Chapter 5 implements the pretraining process of LLMs. It covers computing the training and validation set losses to assess the quality of LLM-generated text, implementing a training function and pretraining the LLM, saving and loading model weights to continue training an LLM, and loading pretrained weights from OpenAI.

Chapter 6 introduces different LLM fine-tuning approaches. It covers preparing a dataset for text classification, modifying a pretrained LLM for fine-tuning, fine-tuning an LLM to identify spam messages, and evaluating the accuracy of a fine-tuned LLM classifier.

Chapter 7 explores the instruction fine-tuning process of LLMs. It covers preparing a dataset for supervised instruction fine-tuning, organizing instruction data in training batches, loading a pretrained LLM and fine-tuning it to follow human instructions, extracting LLM-generated instruction responses for evaluation, and evaluating an instruction-fine-tuned LLM.

About the code

To make it as easy as possible to follow along, all code examples in this book are conveniently available on the Manning website at https://www.manning.com/books/build-a-large-language-model-from-scratch, as well as in Jupyter notebook format on GitHub at https://github.com/rasbt/LLMs-from-scratch. And don't worry about getting stuck—solutions to all the code exercises can be found in appendix C.

This book contains many examples of source code both in numbered listings and in line with normal text. In both cases, source code is formatted in a `fixed-width font like this` to separate it from ordinary text.

In many cases, the original source code has been reformatted; we've added line breaks and reworked indentation to accommodate the available page space in the book. In rare cases, even this was not enough, and listings include line-continuation markers (➥). Additionally, comments in the source code have often been removed from the listings when the code is described in the text. Code annotations accompany many of the listings, highlighting important concepts.

One of the key goals of this book is accessibility, so the code examples have been carefully designed to run efficiently on a regular laptop, without the need for any special hardware. But if you do have access to a GPU, certain sections provide helpful tips on scaling up the datasets and models to take advantage of that extra power.

Throughout the book, we'll be using PyTorch as our go-to tensor and a deep learning library to implement LLMs from the ground up. If PyTorch is new to you, I recommend you start with appendix A, which provides an in-depth introduction, complete with setup recommendations.

实现各种尺寸的 GPT 模型，并计算 GPT 模型的参数数量和存储需求。

第五章实现了LLMs的预训练过程。它涵盖了计算训练集和验证集损失以评估LLM生成文本的质量，实现训练函数，预训练LLM，保存和加载模型权重以继续训练LLM，以及从 OpenAI 加载预训练权重。

第六章介绍了不同的LLM微调方法。它涵盖了为文本分类准备数据集、修改预训练的LLM进行微调、微调LLM以识别垃圾邮件以及评估微调的LLM分类器的准确率。

第七章探讨了LLMs的指令微调过程。它涵盖了为监督指令微调准备数据集、组织训练批次中的指令数据、加载预训练的LLM并将其微调以遵循人类指令、提取LLM生成的指令响应以进行评估，以及评估指令微调的LLM。

关于代码

为了尽可能方便地跟随，本书中的所有代码示例都方便地可在 Manning 网站 https://www.manning.com/books/build-a-large-language-model-from-scratch 上找到，以及 GitHub 上的 Jupyter 笔记本格式 https://github.com/rasbt/LLMs-from-scratch。而且不用担心卡住——所有代码练习的解决方案可以在附录 C 中找到。

这本书包含了许多源代码示例，既有编号列表，也有与普通文本混排的。在两种情况下，源代码都采用固定宽度格式。

字体像这样 将其与普通文本区分开来。

在很多情况下，原始源代码已被重新格式化；我们添加了换行符并重新调整了缩进，以适应书籍中的可用页面空间。在极少数情况下，即使这样也不够，列表中还包括了行续行标记（➥）。此外，当代码在文本中描述时，源代码中的注释通常已从列表中删除。许多列表旁边都有代码注释，突出显示重要概念。

本书的关键目标之一是易于访问，因此代码示例已被精心设计，以便在普通笔记本电脑上高效运行，无需任何特殊硬件。但如果你能访问到 GPU，某些部分提供了有关扩展数据集和模型以利用额外性能的有用提示。

全书我们将使用 PyTorch 作为我们的首选张量库和深度学习库，从零开始实现 LLMs。如果您对 PyTorch 不熟悉，我建议您从附录 A 开始，其中提供了深入的介绍，包括设置建议。

liveBook discussion forum

Purchase of *Build a Large Language Model (From Scratch)* includes free access to liveBook, Manning's online reading platform. Using liveBook's exclusive discussion features, you can attach comments to the book globally or to specific sections or paragraphs. It's a snap to make notes for yourself, ask and answer technical questions, and receive help from the author and other users. To access the forum, go to https://livebook.manning.com/book/build-a-large-language-model-from-scratch/discussion. You can also learn more about Manning's forums and the rules of conduct at https://livebook.manning.com/discussion.

Manning's commitment to readers is to provide a venue where a meaningful dialogue between individual readers and between readers and the author can take place. It is not a commitment to any specific amount of participation on the part of the author, whose contribution to the forum remains voluntary (and unpaid). We suggest you try asking the author some challenging questions lest his interest stray! The forum and the archives of previous discussions will be accessible from the publisher's website as long as the book is in print.

Other online resources

Interested in the latest AI and LLM research trends?

- Check out my blog at https://magazine.sebastianraschka.com, where I regularly discusses the latest AI research with a focus on LLMs.

Need help getting up to speed with deep learning and PyTorch?

- I offer several free courses on my website at https://sebastianraschka.com/teaching. These resources can help you quickly get up to speed with the latest techniques.

Looking for bonus materials related to the book?

- Visit the book's GitHub repository at https://github.com/rasbt/LLMs-from-scratch to find additional resources and examples to supplement your learning.

liveBook 讨论论坛

购买《从零开始构建大型语言模型》包括免费访问现场..

书籍，Manning 的在线阅读平台。使用 liveBook 的独特讨论功能，您可以在全球范围内或特定章节或段落中添加评论。为自己做笔记、提问和回答技术问题、从作者和其他用户那里获得帮助都非常简单。要访问论坛，请访问 https://livebook.manning.com/book/build-a-large-language-model-from-scratch/discussion。您还可以在 https://livebook.manning.com/discussion 了解有关 Manning 论坛和行为准则的更多信息。

曼宁对读者的承诺是提供一个平台，让读者之间以及读者与作者之间可以进行有意义的对话。这并不是对作者参与特定数量活动的承诺，作者对论坛的贡献仍然是自愿的（且未支付报酬）。我们建议您尝试向作者提出一些挑战性的问题，以免他的兴趣转移！只要书籍在印刷中，论坛和先前讨论的存档将可通过出版社网站访问。

其他在线资源

感兴趣于最新的 AI 和LLM研究趋势吗？

- 查看我的博客：https://magazine.sebastianraschka.com，我在那里定期讨论最新的 AI 研究，重点关注LLMs。

需要帮助快速掌握深度学习和 PyTorch 吗？

- 我在我的网站 https://sebastianraschka.com/上提供几门免费课程，教授相关内容。这些资源可以帮助您快速掌握最新技术。

寻找与本书相关的额外材料？

- 访问书籍的 GitHub 仓库 https://github.com/rasbt/LLMs-from -scratch 以获取更多资源和示例，以补充您的学习。

about the author

SEBASTIAN RASCHKA, PhD, has been working in machine learning and AI for more than a decade. In addition to being a researcher, Sebastian has a strong passion for education. He is known for his bestselling books on machine learning with Python and his contributions to open source.

Sebastian is a staff research engineer at Lightning AI, focusing on implementing and training LLMs. Before his industry experience, Sebastian was an assistant professor in the Department of Statistics at the University of Wisconsin-Madison, where he focused on deep learning research. You can learn more about Sebastian at https://sebastianraschka.com.

关于作者

S R，博士，在机器学习和人工智能领域工作了十多年。除了是一名研究人员，Sebastian 对教育也有着强烈的热情。他因其在 Python 机器学习方面的畅销书和对开源的贡献而闻名。

Sebastian 是 Lightning AI 的员工研究工程师，专注于实施和训练LLMs。在行业经验之前，Sebastian 是威斯康星大学麦迪逊分校统计学系的助理教授，在那里他专注于深度学习研究。您可以在 https://sebastianraschka.com 了解更多关于 Sebastian 的信息。

about the cover illustration

The figure on the cover of *Build a Large Language Model (From Scratch)*, titled "Le duchesse," or "The duchess," is taken from a book by Louis Curmer published in 1841. Each illustration is finely drawn and colored by hand.

In those days, it was easy to identify where people lived and what their trade or station in life was just by their dress. Manning celebrates the inventiveness and initiative of the computer business with book covers based on the rich diversity of regional culture centuries ago, brought back to life by pictures from collections such as this one.

关于封面插图

封面上的《从零开始构建大型语言模型》插图，标题为“Le duchesse”或“The duchess”，取自路易·库默尔于 1841 年出版的书籍。

每幅插图都是手工精心绘制和着色的。

在那些日子里，仅凭人们的服饰就能轻易识别他们住在哪里以及他们的职业或社会地位。曼宁通过以数百年前丰富多样的地域文化为背景的书籍封面，庆祝计算机行业的创新精神和主动性，这些文化通过如这一系列图片的收藏被重新赋予生命。