

DeepSeek 图解 10 页 PDF

作者：郭震

2025.2.3

目录

1 本地部署并运行 DeepSeek	2
1.1 为什么要在本地部署 DeepSeek	2
1.2 DeepSeek 本地部署三个步骤	2
1.3 DeepSeek 本地运行使用演示	4
2 DeepSeek 零基础必知	5
2.1 LLM 基础概念	5
2.2 Transformer 基础架构	6
2.3 LLM 基本训练方法	7
2.3.1 预训练 (Pretraining)	7
2.3.2 监督微调 (Supervised Fine-Tuning, SFT)	7
2.3.3 强化学习 (Reinforcement Learning, RL)	7
3 DeepSeek-R1 精华图解	7
3.1 DeepSeek-R1 完整训练过程	7
3.1.1 核心创新 1: 含 R1-Zero 的中间推理模型	8
3.1.2 核心创新 2: 通用强化学习	8
3.2 含 R1-Zero 的中间推理模型训练过程	9
3.3 通用强化学习训练过程	10
3.4 总结 DeepSeek-R1	11
4 参考文献	11

1 本地部署并运行 DeepSeek

1.1 为什么要在本本地部署 DeepSeek

在本本地搭建大模型（如 DeepSeek）具有多个重要的优势，比如：

1. **保护隐私与数据安全**。数据不外传：本本地运行模型可以完全避免数据上传至云端，确保敏感信息不被第三方访问。
2. **可定制化与优化**。支持微调（Fine-tuning）：可以根据特定业务需求对模型进行微调，以适应特定任务，如行业术语、企业内部知识库等。
3. **离线运行，适用于无网络环境**。可在离线环境下运行：适用于无互联网连接或网络受限的场景。提高系统稳定性：即使云服务宕机，本本地大模型依然可以正常工作，不受外部因素影响。

本教程搭建 DeepSeek 好处

本本地搭建 DeepSeek 三个比较实际的好处：

- 本教程接入的是 DeepSeek 推理模型 R1，开源免费，性能强劲
- 本教程搭建方法 **零成本，不需花一分钱**。
- 为了照顾到大部分读者，推荐的搭建方法已将电脑配置要求降到最低，普通电脑也能飞速运行。

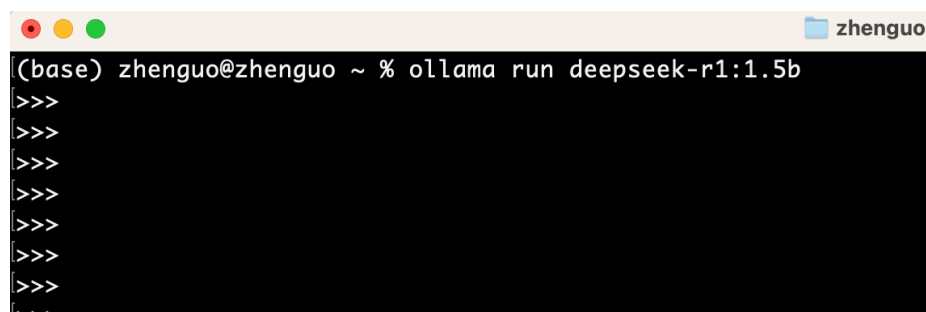
1.2 DeepSeek 本地部署三个步骤

一共只需要三步，就能做到 DeepSeek 在本本地运行并与它对话。

第一步，使用的是 ollama 管理各种不同大模型，ollama 比较直接、干净，一键下载后安装就行，安装过程基本都是下一步。

不知道去哪里下载的，可以直接在我的公众号后台回复：ollama，下载这个软件，然后装上，可以拿着手机扫码下图[1](#)直达我的公众号：

至此在我们本地电脑，DeepSeek 大模型就下载到我们本地电脑，接下来第三步就可以直接使用和它对话了。在 cmd(Windows 电脑) 或 terminal(苹果电脑) 执行命令：`ollama run deepseek-r1:1.5b`，很快就能进入对话界面，如下图4所示：

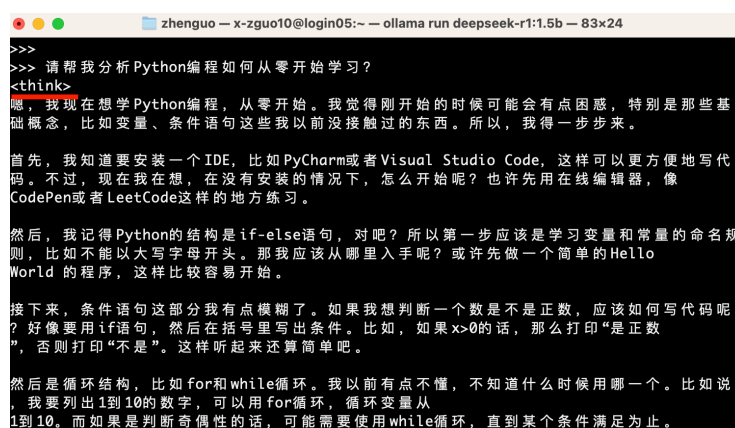


```
(base) zhenguo@zhenguo ~ % ollama run deepseek-r1:1.5b
>>>
>>>
>>>
>>>
>>>
>>>
>>>
```

图 4: Ollama 软件启动 deepseek-r1 界面

1.3 DeepSeek 本地运行使用演示

基于上面步骤搭建完成后，接下来提问 DeepSeek 一个问题：请帮我分析 *Python* 编程如何从零开始学习？，下面是它的回答，首先会有一个 `think` 标签，这里面嵌入的是它的思考过程，不是正式的回答：



```
zhenguo — x-zguo10@login05:~ — ollama run deepseek-r1:1.5b — 83x24
>>>
>>> 请帮我分析Python编程如何从零开始学习？
<think>
嗯，我现在想学Python编程，从零开始。我觉得刚开始的时候可能会有点困惑，特别是那些基础概念，比如变量、条件语句这些我以前没接触过的东西。所以，我得一步步来。

首先，我知道要安装一个IDE，比如PyCharm或者Visual Studio Code，这样可以更方便地写代码。不过，现在我在想，在没有安装的情况下，怎么开始呢？也许先用在线编辑器，像CodePen或者LeetCode这样的地方练习。

然后，我记得Python的结构是if-else语句，对吧？所以第一步应该是学习变量和常量的命名规则，比如不能以大写字母开头。那我应该从哪里入手呢？或许先做一个简单的Hello World的程序，这样比较容易开始。

接下来，条件语句这部分我有点模糊了。如果我想判断一个数是不是正数，应该如何写代码呢？好像要用if语句，然后在括号里写出条件。比如，如果x>0的话，那么打印“是正数”，否则打印“不是”。这样听起来还算简单吧。

然后是循环结构，比如for和while循环。我以前有点不懂，不知道什么时候用哪一个。比如说，我要列出1到10的数字，可以用for循环，循环变量从1到10。而如果是判断奇偶性的话，可能需要使用while循环，直到某个条件满足为止。
```

图 5: deepseek-r1 回复之思考部分

等我们看到另一个结束标签 `think` 后，表明它的思考已经结束，下面一行

就是正式回答，如下图6所示:



```
zhenguo — x-zguo10@login05:~ — ollama run deepseek-r1:1.5b — 83x24
下去，慢慢就能掌握Python了。可能还需要多做一些练习题，巩固所学的内容。
</think>

### 总结：Python编程入门步骤指南

1. **安装IDE**：
   - 使用在线编辑器如CodePen、LeetCode或Visual Studio Code。

2. **学习基础概念**：
   - 熟悉变量和常量命名规则（以小写字母开头）。
   - 掌握基本语法，如if-else语句。

3. **进行简单的练习**：
   - 实现Hello World的程序。
   - 判断数是否为正数或判断奇偶性。

4. **学习循环结构**：
   - 使用for和while循环完成列表和数字的生成。

5. **了解函数概念**：
   - 编写并调用简单的函数，理解参数使用。
```

图 6: deepseek-r1 回复之正式回答部分

2 DeepSeek 零基础必知

为了更深入理解 DeepSeek-R1，首先需要掌握 LLM 的基础知识，包括其工作原理、架构、训练方法。

近年来，人工智能（AI）技术的快速发展催生了大型语言模型（**Large Language Model, LLM**）的兴起。LLM 在自然语言处理（NLP）领域发挥着越来越重要的作用，广泛应用于智能问答、文本生成、代码编写、机器翻译等任务。LLM 是一种基于深度学习的人工智能模型，其核心目标是通过预测下一个单词来理解和生成自然语言。训练 LLM 需要大量的文本数据，使其能够掌握复杂的语言模式并应用于不同任务。

接下来，咱们先从较为基础的概念开始。

2.1 LLM 基础概念

模型参数。其中比较重要的比如`deepseek-r1:1.5b`, `qwen:7b`, `llama:8b`, 这里的 1.5b, 7b, 8b 代表什么? b 是英文的 billion, 意思是十亿, 7b 就是 70 亿, 8b 就是 80 亿, 70 亿、80 亿是指大模型的神经元参数 (权重参数 weight+bias) 的总量。目前大模型都是基于 Transformer 架构, 并且是很多层的 Transformer 结构, 最后还有全连接层等, 所有参数加起来 70 亿, 80 亿, 还有的上千亿。

教程作者：郭震，工作 8 年目前美国 AI 博士在读，公众号：郭震 AI，欢迎关注获取更多原创教程。资料用心打磨且开源，是为了帮助更多人了解获取 AI 知识，严禁拿此资料引流、出书、等形式的商业活动

通用性更强。大模型和我们自己基于某个特定数据集（如 ImageNet、20News-Group）训练的模型在本质上存在一些重要区别。主要区别之一，大模型更加通用，这是因为它们基于大量多样化的数据集进行训练，涵盖了不同领域和任务的数据。这种广泛的学习使得大模型具备了较强的知识迁移能力和多任务处理能力，从而展现出“无所不知、无所不晓”的特性。相比之下，我们基于单一数据集训练的模型通常具有较强的针对性，但其知识范围仅限于该数据集的领域或问题。因此，这类模型的应用范围较为局限，通常只能解决特定领域或单一任务的问题。

Scaling Laws 大家可能在很多场合都见到过。它是一个什么法则呢？大模型之所以能基于大量多样化的数据集进行训练，并最终“学得好”，核心原因之一是 Scaling Laws（扩展规律）的指导和模型自身架构的优势。Scaling Laws 指出参数越多，模型学习能力越强；训练数据规模越大、越多元化，模型最后就会越通用；即使包括噪声数据，模型仍能通过扩展规律提取出通用的知识。而 Transformer 这种架构正好完美做到了 Scaling Laws，Transformer 就是自然语言处理领域实现扩展规律的最好的网络结构。

2.2 Transformer 基础架构

LLM 依赖于 2017 年 Google 提出的 Transformer 模型，该架构相比传统的 RNN（递归神经网络）和 LSTM（长短时记忆网络）具有更高的训练效率和更强的长距离依赖建模能力。Transformer 由多个关键组件组成：1. 自注意力机制（Self-Attention）：模型在处理文本时，会自动关注句子中的重要单词，理解不同词语间的联系。2. 多头注意力（Multi-Head Attention）：使用多个注意力头同时分析不同的语义信息，使得模型的理解能力更强。3. 前馈神经网络（FFN）：非线性变换模块，提升模型的表达能力。4. 位置编码（Positional Encoding）：在没有循环结构的情况下，帮助模型理解单词的顺序信息。

Transformer 结构的优势

1. **高效的并行计算**：摒弃循环结构，使计算速度大幅提升。
2. **更好的上下文理解**：注意力机制可捕捉长文本中的远程依赖关系。
3. **良好的可扩展性**：可适配更大规模模型训练，增强 AI 泛化能力。

2.3 LLM 基本训练方法

2.3.1 预训练 (Pretraining)

LLM 训练通常采用大规模无监督学习，即：**1.** 从互联网上收集大量文本数据，如书籍、新闻、社交媒体等。**2.** 让模型学习词语之间的概率分布，理解句子结构。**3.** 训练目标是 최소화 预测误差，使其能更好地完成语言任务。

2.3.2 监督微调 (Supervised Fine-Tuning, SFT)

在预训练之后，通常需要对模型进行监督微调 (SFT)：使用人工标注的数据集，让模型在特定任务上优化表现。调整参数，使其更符合人类需求，如问答、对话生成等任务。

2.3.3 强化学习 (Reinforcement Learning, RL)

采用强化学习 (RL) 方法进行优化，主要通过人类反馈强化学习 (RLHF, Reinforcement Learning from Human Feedback)：

强化学习 (RLHF) 优化过程

- **步骤 1：** 人类标注者提供高质量回答。
- **步骤 2：** 模型学习人类评分标准，提高输出质量。
- **步骤 3：** 强化训练，使得生成的文本更符合人类偏好。

3 DeepSeek-R1 精华图解

3.1 DeepSeek-R1 完整训练过程

DeepSeek-R1 主要亮点在于出色的数学和逻辑推理能力，区别于一般的通用 AI 模型。其训练方式结合了强化学习 (RL) 与监督微调 (SFT)，创造了一种高效训练，高推理能力 AI 模型的方法。

整个训练过程分为核心两阶段，第一步训练基于 DeepSeek-V3 论文中的基础模型（而非最终版本），并经历了 SFT 和基于纯强化学习调优 + 通用性

偏好调整，如下图7所示：

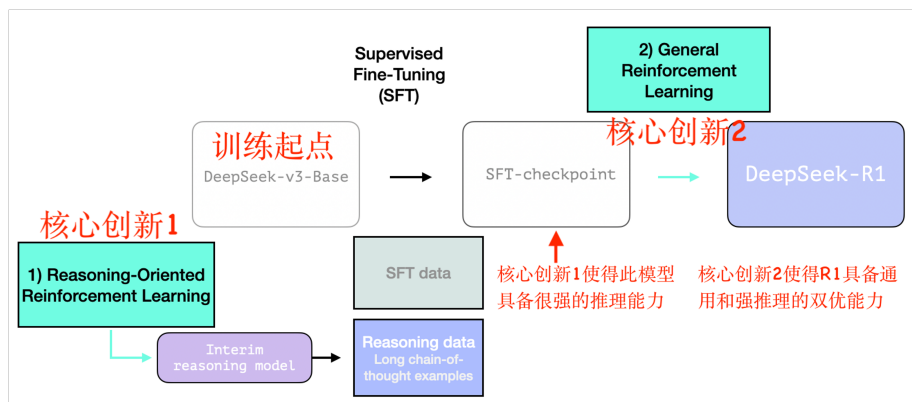


图 7: R1 完整训练过程

训练起点。DeepSeek-R1 的训练起点是 DeepSeek-v3-Base，作为基础模型进行训练，为后续的推理优化奠定基础。

3.1.1 核心创新 1：含 R1-Zero 的中间推理模型

如图7所示,推理导向的强化学习 (Reasoning-Oriented Reinforcement Learning) 得到中间推理模型 (Iterim reasoning model) ,图8会详细解释中间模型的训练过程。

DeepSeek-R1 核心贡献：首次验证了通过纯强化学习也能大幅提升大模型推理能力，开源纯强化学习推理模型 DeepSeek-R1-Zero

R1-Zero 能生成高质量的推理数据,包括大量长链式思维(Chain-of-Thought, CoT)示例,用于支持后续的 SFT 阶段,如图7所示。更加详细介绍参考3.2节。

3.1.2 核心创新 2：通用强化学习

第一阶段 R1-Zero 虽然展现出惊人的推理能力提升，但是也出现了回复时语言混合，非推理任务回复效果差的问题，为了解决这些问题，DeepSeek 提出通用强化学习训练框架。

如图7所示，通用强化学习 (General Reinforcement Learning) 基于 SFT-checkpoint，模型进行通用强化学习 (RL) 训练，优化其在推理任务和其他

通用任务上的表现。更加详细介绍参考3.3节。

3.2 含 R1-Zero 的中间推理模型训练过程

中间模型占据主要训练精力的阶段，实际上完全通过推理导向的强化学习直接训练而成，完全跳过了监督微调（SFT），如下图8所示，只在强化学习的冷启动阶段使用了 SFT。

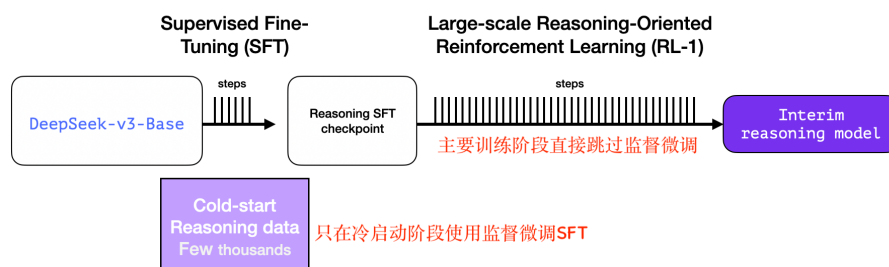


图 8: Interim reasoning model 训练方法

大规模推理导向的强化学习训练，必不可少的就是推理数据，手动标注就太繁琐了，成本昂贵，所以 DeepSeek 团队为了解决这个问题，训了一个 R1-Zero 模型，这是核心创新。

R1-Zero 完全跳过 SFT（监督微调）阶段，直接使用强化学习训练，如下图9所示，基于 V3，直接使用强化学习开训：

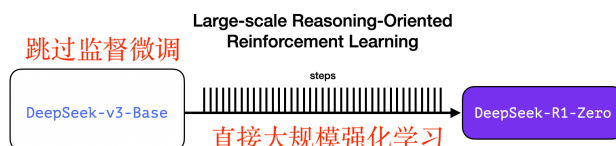


图 9: R1-Zero 完全跳过监督微调

这样做竟然达到了惊人的、意想不到的效果，推理超越 OpenAI O1，如下图10所示，蓝线表示单次推理（pass@1）的准确率，红线表示 16 次推理取一致性结果（cons@16）的准确率，可以看出一致性推理提高了最终性能。虚线代表 OpenAI O1 的基准表现，图中可以看到 DeepSeek-R1-Zero 的性能

逐步接近甚至超越了 OpenAI O1.

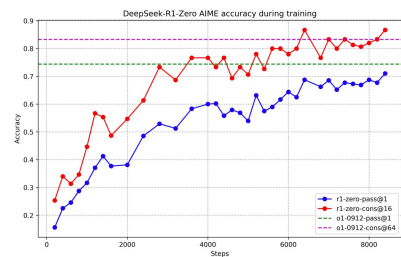


Figure 2 | AIME accuracy of DeepSeek-R1-Zero during training. For each question, we sample 16 responses and calculate the overall average accuracy to ensure a stable evaluation.

图 10: R1-Zero 惊人的推理能力

中间模型虽然推理能力很强，但存在可读性和多任务能力不足的问题，所以才有了第二个创新。

3.3 通用强化学习训练过程

最终偏好调整（Preference Tuning），如下图11所示。通用强化学习训练过程后，使得 R1 不仅在推理任务中表现卓越，同时也在非推理任务中也表现出色。但由于其能力拓展至非推理类应用，因此在这些应用中引入了帮助性（helpfulness）和安全性（safety）奖励模型（类似于 Llama 模型），以优化与这些应用相关的提示处理能力。

DeepSeek-R1 是训练流程的终点，结合了 R1-Zero 的推理能力和通用强化学习的任务适应能力，成为一个兼具强推理和通用能力的高效 AI 模型。

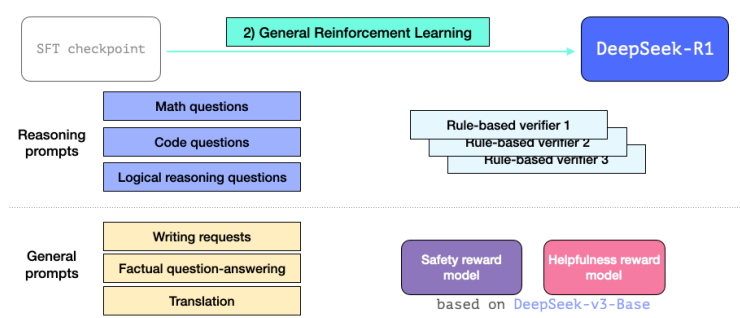


图 11: R1 通用性训练步骤

3.4 总结 DeepSeek-R1

中间推理模型生成：通过推理导向的强化学习（Reasoning-Oriented RL），直接生成高质量的推理数据（CoT 示例），减少人工标注依赖。通用强化学习优化：基于帮助性和安全性奖励模型，优化推理与非推理任务表现，构建通用性强的模型。最终，DeepSeek-R1 将 R1-Zero 的推理能力与通用强化学习的适应能力相结合，成为一个兼具强推理能力和任务广泛适应性的高效 AI 模型。

核心创新总结

中间推理模型生成：通过推理导向的强化学习（Reasoning-Oriented RL），直接生成高质量的推理数据（CoT 示例），减少人工标注依赖。

通用强化学习优化：基于帮助性和安全性奖励模型，优化推理与非推理任务表现，构建通用性强的模型。

最终成果：DeepSeek-R1 将 R1-Zero 的推理能力与通用强化学习的适应能力相结合，成为一个兼具强推理能力和任务广泛适应性的高效 AI 模型。

4 参考文献

<https://newsletter.languagemodels.co/p/the-illustrated-deepseek-r1>

<https://www.interconnects.ai/p/deepseek-r1-recipe-for-ol>

<https://newsletter.maartengrootendorst.com/p/a-visual-guide-to-mixture-of-experts>