

DEEP LEARNING

深度学习

[美]伊恩·古德费洛 (Ian Goodfellow) [加]约书亚·本吉奥 (Yoshua Bengio)

[加]亚伦·库维尔 (Aaron Courville) 著
赵申剑 黎彧君 符天凡 李凯 译 张志华等 审校



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS

DEEP LEARNING

深度学习

[美]伊恩·古德费洛 (Ian Goodfellow) [加]约书亚·本吉奥 (Yoshua Bengio)

[加]亚伦·库维尔 (Aaron Courville) 著
赵申剑 黎彧君 符天凡 李凯 译 张志华等 审校

人民邮电出版社
北京

图书在版编目 (C I P) 数据

深度学习 / (美) 伊恩·古德费洛
(Ian Goodfellow), (加) 约书亚·本吉奥
(Yoshua Bengio), (加) 亚伦·库维尔
(Aaron Courville) 著 ; 赵申剑等译. -- 北京 : 人民
邮电出版社, 2017.8

ISBN 978-7-115-46147-6

I. ①深… II. ①伊… ②约… ③亚… ④赵… III.
①机器学习 IV. ①TP181

中国版本图书馆CIP数据核字(2017)第153811号

版权声明

Deep Learning by Ian Goodfellow, Yoshua Bengio, Aaron Courville

© 2016 Massachusetts Institute of Technology

Simplified Chinese translation copyright © 2017 by Posts & Telecom Press.

This edition published by arrangement with MIT Press through Bardon- Chinese Media Agency. All rights reserved.

本书简体中文翻译版由 Bardon-Chinese Media Agency 代理 MIT Press 授权人民邮电出版社独家出版发行。未经出版者书面许可，不得以任何方式复制或节录本书中的任何内容。

版权所有，侵权必究。

◆ 著 [美] Ian Goodfellow [加] Yoshua Bengio
[加] Aaron Courville
译 赵申剑 黎彧君 符天凡 李 凯
审 校 张志华 等
责任编辑 王峰松
责任印制 焦志炜

◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京画中画印刷有限公司印刷

◆ 开本：787×1092 1/16
印张：33
字数：805 千字 2017 年 8 月第 1 版
印数：1-4 000 册 2017 年 8 月北京第 1 次印刷
著作权合同登记号 图字：01-2016-1194 号

定价：168.00 元

读者服务热线：(010) 81055410 印装质量热线：(010) 81055316

反盗版热线：(010) 81055315

广告经营许可证：京东工商广登字 20170147 号

内容提要

《深度学习》由全球知名的三位专家 Ian Goodfellow、Yoshua Bengio 和 Aaron Courville 撰写，是深度学习领域奠基性的经典教材。全书的内容包括 3 个部分：第 1 部分介绍基本的数学工具和机器学习的概念，它们是深度学习的预备知识；第 2 部分系统深入地讲解现今已成熟的深度学习方法和技术；第 3 部分讨论某些具有前瞻性的方向和想法，它们被公认为是深度学习未来的研究重点。

《深度学习》适合各类读者阅读，包括相关专业的大学生或研究生，以及不具有机器学习或统计背景、但是想要快速补充深度学习知识，以便在实际产品或平台中应用的软件工程师。

作者简介

Ian Goodfellow，谷歌公司 (Google) 的研究科学家，2014 年蒙特利尔大学机器学习博士。他的研究兴趣涵盖大多数深度学习主题，特别是生成模型以及机器学习的安全和隐私。Ian Goodfellow 在研究对抗样本方面是一位有影响力的早期研究者，他发明了生成式对抗网络，在深度学习领域贡献卓越。

Yoshua Bengio，蒙特利尔大学计算机科学与运筹学系 (DIRO) 的教授，蒙特利尔学习算法研究所 (MILA) 的负责人，CIFAR 项目的共同负责人，加拿大统计学习算法研究主席。Yoshua Bengio 的主要研究目标是了解产生智力的学习原则。他还教授“机器学习”研究生课程 (IFT6266)，并培养了一大批研究生和博士后。

Aaron Courville，蒙特利尔大学计算机科学与运筹学系的助理教授，也是 LISA 实验室的成员。目前他的研究兴趣集中在发展深度学习模型和方法，特别是开发概率模型和新颖的推断方法。Aaron Courville 主要专注于计算机视觉应用，在其他领域，如自然语言处理、音频信号处理、语音理解和其他 AI 相关任务方面也有所研究。

中文版审校者简介

张志华，北京大学数学科学学院统计学教授，北京大学大数据研究中心和北京大数据研究院数据科学教授，主要从事机器学习和应用统计学的教学与研究工作。

译者简介

赵申剑，上海交通大学计算机系硕士研究生，研究方向为数值优化和自然语言处理。

黎彧君，上海交通大学计算机系博士研究生，研究方向为数值优化和强化学习。

符天凡，上海交通大学计算机系硕士研究生，研究方向为贝叶斯推断。

李凯，上海交通大学计算机系博士研究生，研究方向为博弈论和强化学习。

中文版推荐语(按姓氏拼音排序)

《深度学习》的中文译本忠实客观地表述了英文原稿的内容。本书的三位共同作者是一个老中青三代结合的整体，既有深度学习领域的奠基人，也有处于研究生涯中期的领域中坚，更有领域里近年涌现的新星。所以，本书的结构行文很好地考虑到了处于研究生涯各个不同阶段的学生和研究人员的需求，是一本非常好的关于深度学习的教科书。

深度学习近年来在学术界和产业界都取得了极大的成功，但诚如本书作者所说，深度学习是创建人工智能系统的一个重要的方法，但不是全部的方法。期望在人工智能领域有所作为的研究人员，可以通过本书充分思考深度学习和传统机器学习、人工智能算法的联系和区别，共同推进本领域的发展。

——微软研究院首席研究员华刚博士

这是一本还在写作阶段就被开发、研究和工程人员极大关注的深度学习教科书。它的出版表明我们进入了一个系统化理解和组织深度学习框架的新时代。这本书从浅入深介绍了基础数学、机器学习经验，以及现阶段深度学习的理论和发展。它能帮助 AI 技术爱好者和从业人员在三位专家学者的思维带领下全方位了解深度学习。

——腾讯优图杰出科学家、香港中文大学教授贾佳亚

深度学习代表了我们这个时代的人工智能技术。这部由该领域最权威的几位学者 Goodfellow、Bengio、Courville 撰写的题为《深度学习》的著作，涵盖了深度学习的基础与应用、理论与实践等各个方面的主要技术，观点鲜明，论述深刻，讲解详尽，内容充实。相信这是每一位关注深度学习人士的必读书目和必备宝典。感谢张志华教授等的辛勤审校，使这部大作能够这么快与中文读者见面。

——华为诺亚方舟实验室主任，北京大学、南京大学客座教授，IEEE Fellow 李航

从基础前馈神经网络到深度生成模型，从数学模型到最佳实践，这本书覆盖了深度学习的各个方面。《深度学习》是当下最适合的入门书籍，强烈推荐给此领域的研究者和从业人员。

——亚马逊主任科学家、Apache MXNet 发起人之一李沐

出自三位深度学习最前沿权威学者的教科书一定要在案前放一本。本书的第二部分是精华，对深度学习的基本技术进行了深入浅出的精彩阐述。

——ResNet 作者之一、Face++ 首席科学家孙剑

过去十年里，深度学习的广泛应用开创了人工智能的新时代。这本教材是深度学习领域有重要影响的几位学者共同撰写。它涵盖了深度学习的主要方向，为想进入该领域的研究人员、工程师以及初学者提供了一个很好的系统性教材。

——香港中文大学信息工程系主任汤晓鸥教授

这是一本教科书，又不只是一本教科书。任何对深度学习感兴趣的读者，本书在很长一段时间里，都将是你能获得的最全面系统的资料，以及思考并真正推进深度学习产业应用、构建智能化社会框架的绝佳理论起点。

——新智元创始人兼 CEO 杨静

www.epubit.com.cn
异步社区

译者序

青山遮不住，毕竟东流去

深度学习这个术语自 2006 年被正式提出后，在最近 10 年得到了巨大发展。它使人工智能 (AI) 产生了革命性的突破，让我们切实地领略到人工智能给人类生活带来改变的潜力。2016 年 12 月，MIT 出版社出版了 Ian Goodfellow、Yoshua Bengio 和 Aaron Courville 三位学者撰写的《Deep Learning》一书。三位作者一直耕耘于机器学习领域的前沿，引领了深度学习的发展潮流，是深度学习众多方法的主要贡献者。该书正应其时，一经出版就风靡全球。

该书包括 3 个部分，第 1 部分介绍基本的数学工具和机器学习的概念，它们是深度学习的预备知识。第 2 部分系统深入地讲解现今已成熟的深度学习方法和技术。第 3 部分讨论某些具有前瞻性的方向和想法，它们被公认为是深度学习未来的研究重点。因此，该书适用于不同层次的读者。我本人在阅读该书时受到启发良多，大有裨益，并采用该书作为教材在北京大学讲授深度学习课程。

这是一本涵盖深度学习技术细节的教科书，它告诉我们深度学习集技术、科学与艺术于一体，牵涉统计、优化、矩阵、算法、编程、分布式计算等多个领域。书中同时也蕴含了作者对深度学习的理解和思考，处处闪烁着深刻的思想，耐人回味。第 1 章关于深度学习的思想、历史发展等论述尤为透彻而精辟。

作者在书中写到：“人工智能的真正挑战在于解决那些对人来说很容易执行、但很难形式化描述的任务，比如识别人们所说的话或图像中的脸。对于这些问题，我们人类往往可以凭直觉轻易地解决”。为了应对这些挑战，他们提出让计算机从经验中学习，并根据层次化的概念体系来理解世界，而每个概念通过与某些相对简单的概念之间的关系来定义。由此，作者给出了深度学习的定义：“层次化的概念让计算机构建较简单的概念来学习复杂概念。如果绘制出表示这些概念如何建立在彼此之上的一幅图，我们将得到一张‘深’(层次很多) 的图。由此，我们称这种方法为 AI 深度学习 (deep learning)”。

作者指出：“一般认为，到目前为止深度学习已经经历了三次发展浪潮：20 世纪 40 年代到 60 年代深度学习的雏形出现在控制论 (cybernetics) 中，20 世纪 80 年代到 90 年代深度学习以联结主义 (connectionism) 为代表，而从 2006 年开始，以深度学习之名复兴”。

谈到深度学习与脑科学或者神经科学的关系，作者强调：“如今神经科学在深度学习研究中的作用被削弱，主要原因是我们根本没有足够的关于大脑的信息作为指导去使用它。要获得对被大脑实际使用算法的深刻理解，我们需要有能力同时监测 (至少是) 数千相连神经元的活动。我们不能够做到这一点，所以我们甚至连大脑最简单、最深入研究的部分都还远远没有理解”。值得注意的是，我国有些专家热衷倡导人工智能与脑科学或认知学科的交叉研究，推动国家在所谓的“类脑智能”等领域投入大量资源。且不论我国是否真有同时精通人工智能和脑科学或认知心理学的学者，至少对交叉领域，我们都应该怀着务实、理性的求是态度。唯有如此，我们才有可能在这一波人工智能发展浪潮中有所作为，而不是又成为一群观潮人。

作者进一步指出：“媒体报道经常强调深度学习与大脑的相似性。的确，深度学习研究者比其他机器学习领域（如核方法或贝叶斯统计）的研究者更可能地引用大脑作为参考，但大家不应该认为深度学习在尝试模拟大脑。现代深度学习从许多领域获取灵感，特别是应用数学的基本内容如线性代数、概率论、信息论和数值优化。尽管一些深度学习的研究人员引用神经科学作为重要的灵感来源，然而其他学者完全不关心神经科学”。的确，对于广大青年学者和一线的工程师来说，我们是可以完全不用因为不懂神经（或脑）科学而对深度学习、人工智能躊躇不前。数学模型、计算方法和应用驱动才是我们研究人工智能的可行之道。深度学习和人工智能不是飘悬在我们头顶的框架，而是立足于我们脚下的技术。我们诚然可以从哲学层面或角度来欣赏科学与技术，但过度地从哲学层面来研究科学问题只会导致一些空洞的名词。

关于人工神经网络在 20 世纪 90 年代中期的衰落，作者分析到：“基于神经网络和其他 AI 技术的创业公司开始寻求投资，其做法野心勃勃但不切实际。当 AI 研究不能实现这些不合理的期望时，投资者感到失望。同时，机器学习的其他领域取得了进步。比如，核方法和图模型都在很多重要任务上实现了很好的效果。这两个因素导致了神经网络热潮的第二次衰退，并一直持续到 2007 年”。“其兴也悖焉，其亡也忽焉”。这个教训也同样值得当今基于深度学习的创业界、工业界和学术界等警醒。

我非常荣幸获得人民邮电出版社王峰松先生的邀请来负责该书的中文翻译。我是 2016 年 7 月收到王先生的邀请，但那时我正忙于找工作，无暇顾及。然而，当我和我的学生讨论翻译事宜时，他们一致认为这是一件非常有意义的事情，表达愿意来承担。译稿是由我的四位学生赵申剑、黎彧君、符天凡和李凯独立完成的。申剑和天凡是二年级的硕士生，而李凯和彧君则分别是二年级和三年级的直博生。虽然他们在机器学习领域都还是新人，其知识结构还不全面，但是他们热情高涨、勤于学习、工作专注、执行力极强。他们通过重现书中的算法代码和阅读相关文献来加强理解，在不到三个月的时间就拿出了译著的初稿，之后又经过自校对、交叉校对等环节力图使译著保持正确性和一致性。他们自我协调、主动揽责、相互谦让，他们的责任心和独立工作能力让我倍感欣慰，因而得以从容。

由于我们无论是中文还是英文能力都深感有限，译文恐怕还是有些生硬，我们特别担心未能完整地传达出原作者的真实思想和观点。因此，我们强烈地建议有条件的读者去阅读英文原著，也非常期待大家继续指正译著，以便今后进一步修订完善。我恳请大家多给予 4 位译者以鼓励。请把你们对译著的批评留给我，这是我作为他们的导师必须要承担的，也是我对王峰松先生的信任做出的承诺。

当初译稿基本完成时，我们决定把它公开在 GitHub 上，希望通过广大读者的参与来完善译稿。令人惊喜的是，有上百位热心读者给予了大量富有建设性的修改意见，其中有 20 多位热心读者直接帮助润色校对（详见中文版致谢名单）。可以说，这本译著是大家共同努力的结晶。这些读者来自一线的工程师和在校的学生，从中我领略到了他们对深度学习和机器学习领域的挚爱。更重要的是，我感受到了他们开放、合作和奉献的精神，而这也是推动人工智能发展不可或缺的。因此，我更加坚定地认为中国人工智能发展的希望在于年青学者，唯有他们才能让我国人工智能学科在世界有竞争力和影响力。

江山代有人才出，各领风骚数十年！

张志华代笔

2017 年 5 月 12 日于北大静园六院

中文版致谢

首先，我们要感谢原书作者在本书翻译时给予我们的大力帮助。特别是，原书作者和我们分享了书中的原图和参考文献库，这极大节省了我们的时间和精力。

本书涉及的内容博大且思想深刻，如果没有众多同学和网友的帮助，我们不可能顺利完成翻译。

我们才疏学浅而受此重任，深知自身水平难以将本书翻译得很准确。因此我们完成初稿后，将书稿公开于 GitHub，及早接受网友的批评和建议。以下网友为本书的翻译初稿提供了很多及时的反馈和宝贵的修改意见：@tttwwy、@tankeco、@fairmiracle、@GageGao、@huangpingchun、@MaHongP、@acgtyrant、@yanhuibin315、@Buttonwood、@titicacafz、@weijy026a、@RuiZhang1993、@zymiboxpay、@xingkongliang、@oisc、@tielei、@yuduowu、@Qingmu、@HC-2016、@xiaomingabc、@bengordai、@Bojian、@JoyFYan、@minoriwww、@khty2000、@gump88、@zdx3578、@PassStory、@imwebson、@wlbksy、@roachsinai、@Elvinczp、@endymecy、@9578577、@linzhp、@cnscottzheng、@germany-zhu、@zhangyafeikimi、@showgood163、@kangqf、@NeutronT、@badpoem、@kkpoker、@Seaball、@wheao、@angrymidiao、@ZhiweiYang、@corenel、@zhaoyu611、@SiriusXDJ、@dfcv24、@EmisXXY、@FlyingFire、@vsooda、@friskit-china、@poerin、@ninesunqian、@JiaqiYao、@Sofring、@wenlei、@wizyoung、@imageslr、@indam、@XuLYC、@zhouqingping、@freedomRen、@runPenguin 和 @piantou。

在此期间，我们 4 位译者再次进行了校对并且相互之间也校对了一遍。然而仅仅通过我们的校对，实在难以发现翻译中存在的所有问题。因此，我们邀请一些同学和网友帮助我们校对。经过他们的校对，本书的翻译质量得到了极大的提升。在此我们一一列出，以表示我们由衷的感谢！

- 第 1 章 (引言)：刘畅、许丁杰、潘雨粟和 NeutronT 阅读了本章，并对很多语句提出了不少修改建议。林中鹏进行了校对，他提出了很多独到的修改建议。
- 第 2 章 (线性代数)：许丁杰和骆徐圣阅读了本章，并修改语句。李若愚进行了校对，提出了很多细心的建议。蒋武轩阅读并润色了部分内容，提升了译文准确性和可读性。
- 第 3 章 (概率与信息论)：许丁杰阅读了本章，并修改语句。李培炎和何翊卓进行了校对，并修改了很多中文用词，使翻译更加准确。
- 第 4 章 (数值计算)：张亚霏阅读了本章，并对其他章节也提出了一些修改建议。张源进行了校对，并指出了原文可能存在的问题，非常仔细。
- 第 5 章 (机器学习基础)：郭浩和黄平春阅读了本章，并修改语句。李东和林中鹏进行了校对。本章篇幅较长，能够有现在的翻译质量离不开这 4 位的贡献。
- 第 6 章 (深度前馈网络)：周卫林、林中鹏和张远航阅读了本章，并提出修改意见。

- 第 7 章 (深度学习中的正则化): 周柏村进行了非常细心的校对, 指出了大量问题, 令翻译更加准确。
- 第 8 章 (深度模型中的优化): 房晓宇和吴翔阅读了本章。黄平春进行了校对, 他提出的很多建议让行文更加流畅易懂。
- 第 9 章 (卷积网络): 赵雨和潘雨粟阅读了本章, 并润色语句。丁志铭进行了非常仔细的校对, 并指出很多翻译问题。
- 第 10 章 (序列建模: 循环和递归网络): 刘畅阅读了本章。赵雨提供了详细的校对建议, 尹瑞清根据他的翻译版本, 给我们的版本提出了很多建议。虽然仍存在一些分歧, 但我们两个版本的整合, 让翻译质量提升很多。
- 第 12 章 (应用): 潘雨粟进行了校对。在他的校对之前, 本章阅读起来比较困难。他提供的修改建议, 不仅提高了行文流畅度, 还提升了译文的准确度。
- 第 13 章 (线性因子模型): 贺天行阅读了本章, 修改语句。杨志伟校对了本章, 润色大量语句。
- 第 14 章 (自编码器): 李雨慧和黄平春进行了校对。李雨慧提升了语言的流畅度, 黄平春纠正了不少错误, 提高了准确性。
- 第 15 章 (表示学习): cnscottzheng 阅读了本章, 并修改语句。
- 第 17 章 (蒙特卡罗方法): 张远航提供了非常细致的校对, 后续又校对了一遍, 使译文质量大大提升。
- 第 18 章 (直面配分函数): 吴家楠进行了校对, 提升了译文准确性和可读性。
- 第 19 章 (近似推断): 黄浩军、张远航和张源源进行了校对。本章虽然篇幅不大, 但内容有深度, 译文在 3 位的帮助下提高了准确度。

所有校对的修改建议都保存在 GitHub 上, 再次感谢以上同学和网友的付出。经过这 5 个多月的修改, 初稿慢慢变成了最终提交给出版社的稿件。尽管还有很多问题, 但大部分内容是可读的, 并且是准确的。当然目前的译文仍存在一些没有及时发现的问题, 因此修订工作也将持续更新, 不断修改。我们非常希望读者能到 GitHub 提建议, 并且非常欢迎, 无论多么小的修改建议, 都是非常宝贵的。

此外, 我们还要感谢魏太云学长, 他帮助我们与出版社沟通交流, 并给予了我们很多排版上的指导。

最后, 感谢我们的导师张志华教授, 没有老师的 support, 我们难以完成翻译。

英文原书致谢

如果没有他人的贡献，这本书将不可能完成。我们感谢为本书提出建议和帮助组织内容结构的人：Guillaume Alain、Kyunghyun Cho、Çağlar Gülc̄ehre、David Krueger、Hugo Larochelle、Razvan Pascanu 和 Thomas Rohée。

我们感谢为本书内容提供反馈的人。其中一些人对许多章都给出了建议：Martín Abadi、Guillaume Alain、Ion Androutsopoulos、Fred Bertsch、Olexa Bilaniuk、Ufuk Can Biçici、Matko Bošnjak、John Boersma、Greg Brockman、Alexandre de Brébisson、Pierre Luc Carrier、Sarah Chandar、Pawel Chilinski、Mark Daoust、Oleg Dashevskii、Laurent Dinh、Stephan Dreseitl、Jim Fan、Miao Fan、Meire Fortunato、Frédéric Francis、Nando de Freitas、Çağlar Gülc̄ehre、Jurgen Van Gael、Javier Alonso García、Jonathan Hunt、Gopi Jeyaram、Chingiz Kabytayev、Lukasz Kaiser、Varun Kanade、Asifullah Khan、Akiel Khan、John King、Diederik P. Kingma、Yann Le-Cun、Rudolf Mathey、Matías Mattamala、Abhinav Maurya、Kevin Murphy、Oleg Mürk、Roman Novak、Augustus Q. Odena、Simon Pavlik、Karl Pichotta、Eddie Pierce、Kari Pulli、Roussel Rahman、Tapani Raiko、Anurag Ranjan、Johannes Roith、Mihaela Rosca、Halis Sak、César Salgado、Grigory Sapunov、Yoshinori Sasaki、Mike Schuster、Julian Serban、Nir Shabat、Ken Shirriff、Andre Simpelo、Scott Stanley、David Sussillo、Ilya Sutskever、Carles Gelada Sáez、Graham Taylor、Valentin Tolmer、Massimiliano Tomassoli、An Tran、Shubhendu Trivedi、Alexey Umnov、Vincent Vanhoucke、Marco Visentini-Scarzanella、Martin Vita、David Warde-Farley、Dustin Webb、Kelvin Xu、Wei Xue、Ke Yang、Li Yao、Zygmunt Zajac 和 Ozan Çağlayan。

我们也要感谢对单个章节提供有效反馈的人。

- 数学符号：Zhang Yuanhang。
- 第 1 章（引言）：Yusuf Akgul、Sebastien Bratieres、Samira Ebrahimi、Charlie Gorichanaz、Brendan Loudermilk、Eric Morris、Cosmin Parvulescu 和 Alfredo Solano。
- 第 2 章（线性代数）：Amjad Almahairi、Nikola Banić、Kevin Bennett、Philippe Castonguay、Oscar Chang、Eric Fosler-Lussier、Andrey Khalyavin、Sergey Oreshkov、István Petrás、Dennis Prangle、Thomas Rohée、Gitanjali Gulve Sehgal、Colby Toland、Alessandro Vitale 和 Bob Welland。
- 第 3 章（概率与信息论）：John Philip Anderson、Kai Arulkumaran、Vincent Dumoulin、Rui Fa、Stephan Gouws、Artem Oboturov、Antti Rasmus、Alexey Surkov 和 Volker Tresp。
- 第 4 章（数值计算）：Tran Lam An Ian Fischer 和 Hu Yuhuang。
- 第 5 章（机器学习基础）：Dzmitry Bahdanau、Justin Domingue、Nikhil Garg、Makoto Otsuka、Bob Pepin、Philip Popien、Emmanuel Rayner、Peter Shepard、Kee-Bong

Song、Zheng Sun 和 Andy Wu。

- 第 6 章 (深度前馈网络): Uriel Berdugo、Fabrizio Bottarel、Elizabeth Burl、Ishan Durugkar、Jeff Hlywa、Jong Wook Kim、David Krueger 和 Aditya Kumar Praharaj。
- 第 7 章 (深度学习中的正则化): Morten Kolbæk、Kshitij Lauria、Inkyu Lee、Sunil Mohan、Hai Phong Phan 和 Joshua Salisbury。
- 第 8 章 (深度模型中的优化): Marcel Ackermann、Peter Armitage、Rowel Atienza、Andrew Brock、Tegan Maharaj、James Martens、Kashif Rasul、Klaus Strobl 和 Nicholas Turner。
- 第 9 章 (卷积网络): Martín Arjovsky、Eugene Brevdo、Konstantin Divilov、Eric Jensen、Mehdi Mirza、Alex Paino、Marjorie Sayer、Ryan Stout 和 Wentao Wu。
- 第 10 章 (序列建模: 循环和递归网络): Gökçen Eraslan、Steven Hickson、Razvan Pascanu、Lorenzo von Ritter、Rui Rodrigues、Dmitriy Serdyuk、Dongyu Shi 和 Kaiyu Yang。
- 第 11 章 (实践方法论): Daniel Beckstein。
- 第 12 章 (应用): George Dahl、Vladimir Nekrasov 和 Ribana Roscher。
- 第 13 章 (线性因子模型): Jayanth Koushik。
- 第 15 章 (表示学习): Kunal Ghosh。
- 第 16 章 (深度学习中的结构化概率模型): Minh Lê 和 Anton Varfolom。
- 第 18 章 (直面配分函数): Sam Bowman。
- 第 19 章 (近似推断): Yujia Bao。
- 第 20 章 (深度生成模型): Nicolas Chapados、Daniel Galvez、Wenming Ma、Fady Medhat、Shakir Mohamed 和 Grégoire Montavon。
- 参考文献: Lukas Michelbacher 和 Leslie N. Smith。

我们还要感谢那些允许我们引用他们的出版物中的图片、数据的人。我们在图片标题的文字中注明了他们的贡献。

我们还要感谢 Lu Wang 为我们写了 pdf2htmlEX, 我们用它来制作这本书的网页版本, Lu Wang 还帮助我们改进了生成的 HTML 的质量。

我们还要感谢 Ian 的妻子 Daniela Flori Goodfellow 在 Ian 的写作过程中的耐心支持和检查。

我们还要感谢 Google Brain 团队提供了学术环境, 从而使得 Ian 能够花费大量时间写作本书并接受同行的反馈和指导。我们特别感谢 Ian 的前任经理 Greg Corrado 和他的现任经理 Samy Bengio 对这项工作的支持。最后我们还要感谢 Geoffrey Hinton 在写作困难时的鼓励。

数学符号

下面简要介绍本书所使用的数学符号。我们在第 2 ~ 4 章中描述大多数数学概念，如果你不熟悉任何相应的数学概念，可以参考对应的章节。

数和数组

a	标量 (整数或实数)
\mathbf{a}	向量
A	矩阵
\mathbf{A}	张量
I_n	n 行 n 列的单位矩阵
I	维度蕴含于上下文的单位矩阵
$e^{(i)}$	标准基向量 $[0, \dots, 0, 1, 0, \dots, 0]$, 其中索引 i 处值为 1
$\text{diag}(\mathbf{a})$	对角方阵, 其中对角元素由 \mathbf{a} 给定
a	标量随机变量
\mathbf{a}	向量随机变量
\mathbf{A}	矩阵随机变量

集合和图

\mathbb{A}	集合
\mathbb{R}	实数集
$\{0, 1\}$	包含 0 和 1 的集合
$\{0, 1, \dots, n\}$	包含 0 和 n 之间所有整数的集合
$[a, b]$	包含 a 和 b 的实数区间
$(a, b]$	不包含 a 但包含 b 的实数区间
$\mathbb{A} \setminus \mathbb{B}$	差集, 即其元素包含于 \mathbb{A} 但不包含于 \mathbb{B}
\mathcal{G}	图
$P_{\mathcal{G}}(\mathbf{x}_i)$	图 \mathcal{G} 中 \mathbf{x}_i 的父节点

索引

a_i	向量 \mathbf{a} 的第 i 个元素, 其中索引从 1 开始
a_{-i}	除了第 i 个元素, \mathbf{a} 的所有元素
$A_{i,j}$	矩阵 \mathbf{A} 的 i,j 元素
$\mathbf{A}_{i,:}$	矩阵 \mathbf{A} 的第 i 行
$\mathbf{A}_{:,i}$	矩阵 \mathbf{A} 的第 i 列
$A_{i,j,k}$	3 维张量 \mathbf{A} 的 (i,j,k) 元素
$\mathbf{A}_{:,:,i}$	3 维张量的 2 维切片
a_i	随机向量 \mathbf{a} 的第 i 个元素

线性代数中的操作

\mathbf{A}^\top	矩阵 \mathbf{A} 的转置
\mathbf{A}^+	\mathbf{A} 的 Moore-Penrose 伪逆
$\mathbf{A} \odot \mathbf{B}$	\mathbf{A} 和 \mathbf{B} 的逐元素乘积 (Hadamard 乘积)
$\det(\mathbf{A})$	\mathbf{A} 的行列式

微积分

$\frac{dy}{dx}$	y 关于 x 的导数
$\frac{\partial y}{\partial x}$	y 关于 x 的偏导
$\nabla_x y$	y 关于 x 的梯度
$\nabla_X y$	y 关于 X 的矩阵导数
$\nabla_{\mathbf{X}} y$	y 关于 \mathbf{X} 求导后的张量
$\frac{\partial f}{\partial x}$	$f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ 的 Jacobian 矩阵 $\mathbf{J} \in \mathbb{R}^{m \times n}$
$\nabla_x^2 f(x)$ or $\mathbf{H}(f)(x)$	f 在点 x 处的 Hessian 矩阵
$\int f(x) dx$	x 整个域上的定积分
$\int_S f(x) dx$	集合 S 上关于 x 的定积分

概率和信息论

$a \perp b$	a 和 b 相互独立的随机变量
$a \perp b \mid c$	给定 c 后条件独立
$P(a)$	离散变量上的概率分布
$p(a)$	连续变量 (或变量类型未指定时) 上的概率分布
$a \sim P$	具有分布 P 的随机变量 a
$\mathbb{E}_{x \sim P}[f(x)]$ or $\mathbb{E}f(x)$	$f(x)$ 关于 $P(x)$ 的期望
$\text{Var}(f(x))$	$f(x)$ 在分布 $P(x)$ 下的方差
$\text{Cov}(f(x), g(x))$	$f(x)$ 和 $g(x)$ 在分布 $P(x)$ 下的协方差
$H(x)$	随机变量 x 的香浓熵
$D_{\text{KL}}(P \parallel Q)$	P 和 Q 的 KL 散度
$\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$	均值为 $\boldsymbol{\mu}$, 协方差为 $\boldsymbol{\Sigma}$, x 上的高斯分布

函数

$f : \mathbb{A} \rightarrow \mathbb{B}$	定义域为 \mathbb{A} 值域为 \mathbb{B} 的函数 f
$f \circ g$	f 和 g 的组合
$f(\mathbf{x}; \boldsymbol{\theta})$	由 $\boldsymbol{\theta}$ 参数化, 关于 \mathbf{x} 的函数 (有时为简化表示, 我们忽略 $\boldsymbol{\theta}$ 而记为 $f(\mathbf{x})$)
$\log x$	x 的自然对数
$\sigma(x)$	Logistic sigmoid, $\frac{1}{1 + \exp(-x)}$
$\zeta(x)$	Softplus, $\log(1 + \exp(x))$
$\ \mathbf{x}\ _p$	\mathbf{x} 的 L^p 范数
$\ \mathbf{x}\ $	\mathbf{x} 的 L^2 范数
x^+	x 的正数部分, 即 $\max(0, x)$
$\mathbf{1}_{\text{condition}}$	如果条件为真则为 1, 否则为 0

有时候我们使用函数 f , 它的参数是一个标量, 但应用到一个向量、矩阵或张量: $f(\mathbf{x})$ 、 $f(\mathbf{X})$ 或 $f(\mathbf{X})$ 。这表示逐元素地将 f 应用于数组。例如, $\mathbf{C} = \sigma(\mathbf{X})$, 则对于所有合法的 i, j 和 k , $C_{i,j,k} = \sigma(X_{i,j,k})$ 。

数据集和分布

p_{data}	数据生成分布
\hat{p}_{train}	由训练集定义的经验分布
\mathbb{X}	训练样本的集合
$\mathbf{x}^{(i)}$	数据集的第 i 个样本 (输入)
$y^{(i)}$ 或 $\mathbf{y}^{(i)}$	监督学习中与 $\mathbf{x}^{(i)}$ 关联的目标
\mathbf{X}	$m \times n$ 的矩阵, 其中行 $\mathbf{X}_{i,:}$ 为输入样本 $\mathbf{x}^{(i)}$

目 录

第 1 章 引言	1
1.1 本书面向的读者	7
1.2 深度学习的历史趋势	8
1.2.1 神经网络的众多名称和命运变迁	8
1.2.2 与日俱增的数据量	12
1.2.3 与日俱增的模型规模	13
1.2.4 与日俱增的精度、复杂度和对现实世界的冲击	15

第 1 部分 应用数学与机器学习基础

第 2 章 线性代数	19
2.1 标量、向量、矩阵和张量	19
2.2 矩阵和向量相乘	21
2.3 单位矩阵和逆矩阵	22
2.4 线性相关和生成子空间	23
2.5 范数	24
2.6 特殊类型的矩阵和向量	25
2.7 特征分解	26
2.8 奇异值分解	28
2.9 Moore-Penrose 伪逆	28
2.10 迹运算	29
2.11 行列式	30
2.12 实例：主成分分析	30
第 3 章 概率与信息论	34
3.1 为什么要使用概率	34
3.2 随机变量	35
3.3 概率分布	36
3.3.1 离散型变量和概率质量函数	36
3.3.2 连续型变量和概率密度函数	36
3.4 边缘概率	37
3.5 条件概率	37
3.6 条件概率的链式法则	38

3.7 独立性和条件独立性	38
3.8 期望、方差和协方差	38
3.9 常用概率分布	39
3.9.1 Bernoulli 分布	40
3.9.2 Multinoulli 分布	40
3.9.3 高斯分布	40
3.9.4 指数分布和 Laplace 分布	41
3.9.5 Dirac 分布和经验分布	42
3.9.6 分布的混合	42
3.10 常用函数的有用性质	43
3.11 贝叶斯规则	45
3.12 连续型变量的技术细节	45
3.13 信息论	47
3.14 结构化概率模型	49
第 4 章 数值计算	52
4.1 上溢和下溢	52
4.2 病态条件	53
4.3 基于梯度的优化方法	53
4.3.1 梯度之上: Jacobian 和 Hessian 矩阵	56
4.4 约束优化	60
4.5 实例: 线性最小二乘	61
第 5 章 机器学习基础	63
5.1 学习算法	63
5.1.1 任务 T	63
5.1.2 性能度量 P	66
5.1.3 经验 E	66
5.1.4 示例: 线性回归	68
5.2 容量、过拟合和欠拟合	70
5.2.1 没有免费午餐定理	73
5.2.2 正则化	74
5.3 超参数和验证集	76
5.3.1 交叉验证	76
5.4 估计、偏差和方差	77
5.4.1 点估计	77
5.4.2 偏差	78
5.4.3 方差和标准差	80
5.4.4 权衡偏差和方差以最小化均方误差	81
5.4.5 一致性	82
5.5 最大似然估计	82
5.5.1 条件对数似然和均方误差	84

5.5.2 最大似然的性质	84
5.6 贝叶斯统计	85
5.6.1 最大后验 (MAP) 估计	87
5.7 监督学习算法	88
5.7.1 概率监督学习	88
5.7.2 支持向量机	88
5.7.3 其他简单的监督学习算法	90
5.8 无监督学习算法	91
5.8.1 主成分分析	92
5.8.2 k -均值聚类	94
5.9 随机梯度下降	94
5.10 构建机器学习算法	96
5.11 促使深度学习发展的挑战	96
5.11.1 维数灾难	97
5.11.2 局部不变性和平滑正则化	97
5.11.3 流形学习	99

第 2 部分 深度网络：现代实践

第 6 章 深度前馈网络	105
6.1 实例：学习 XOR	107
6.2 基于梯度的学习	110
6.2.1 代价函数	111
6.2.2 输出单元	113
6.3 隐藏单元	119
6.3.1 整流线性单元及其扩展	120
6.3.2 logistic sigmoid 与双曲正切函数	121
6.3.3 其他隐藏单元	122
6.4 架构设计	123
6.4.1 万能近似性质和深度	123
6.4.2 其他架构上的考虑	126
6.5 反向传播和其他的微分算法	126
6.5.1 计算图	127
6.5.2 微积分中的链式法则	128
6.5.3 递归地使用链式法则来实现反向传播	128
6.5.4 全连接 MLP 中的反向传播计算	131
6.5.5 符号到符号的导数	131
6.5.6 一般化的反向传播	133
6.5.7 实例：用于 MLP 训练的反向传播	135
6.5.8 复杂化	137

6.5.9 深度学习界以外的微分	137
6.5.10 高阶微分	138
6.6 历史小记	139
第 7 章 深度学习中的正则化	141
7.1 参数范数惩罚	142
7.1.1 L^2 参数正则化	142
7.1.2 L^1 正则化	144
7.2 作为约束的范数惩罚	146
7.3 正则化和欠约束问题	147
7.4 数据集增强	148
7.5 噪声鲁棒性	149
7.5.1 向输出目标注入噪声	150
7.6 半监督学习	150
7.7 多任务学习	150
7.8 提前终止	151
7.9 参数绑定和参数共享	156
7.9.1 卷积神经网络	156
7.10 稀疏表示	157
7.11 Bagging 和其他集成方法	158
7.12 Dropout	159
7.13 对抗训练	165
7.14 切面距离、正切传播和流形正切分类器	167
第 8 章 深度模型中的优化	169
8.1 学习和纯优化有什么不同	169
8.1.1 经验风险最小化	169
8.1.2 代理损失函数和提前终止	170
8.1.3 批量算法和小批量算法	170
8.2 神经网络优化中的挑战	173
8.2.1 病态	173
8.2.2 局部极小值	174
8.2.3 高原、鞍点和其他平坦区域	175
8.2.4 悬崖和梯度爆炸	177
8.2.5 长期依赖	177
8.2.6 非精确梯度	178
8.2.7 局部和全局结构间的弱对应	178
8.2.8 优化的理论限制	179
8.3 基本算法	180
8.3.1 随机梯度下降	180
8.3.2 动量	181
8.3.3 Nesterov 动量	183

8.4	参数初始化策略	184
8.5	自适应学习率算法	187
8.5.1	AdaGrad	187
8.5.2	RMSProp	188
8.5.3	Adam	189
8.5.4	选择正确的优化算法	190
8.6	二阶近似方法	190
8.6.1	牛顿法	190
8.6.2	共轭梯度	191
8.6.3	BFGS	193
8.7	优化策略和元算法	194
8.7.1	批标准化	194
8.7.2	坐标下降	196
8.7.3	Polyak 平均	197
8.7.4	监督预训练	197
8.7.5	设计有助于优化的模型	199
8.7.6	延拓法和课程学习	199
第 9 章	卷积网络	201
9.1	卷积运算	201
9.2	动机	203
9.3	池化	207
9.4	卷积与池化作为一种无限强的先验	210
9.5	基本卷积函数的变体	211
9.6	结构化输出	218
9.7	数据类型	219
9.8	高效的卷积算法	220
9.9	随机或无监督的特征	220
9.10	卷积网络的神经科学基础	221
9.11	卷积网络与深度学习的历史	226
第 10 章	序列建模：循环和递归网络	227
10.1	展开计算图	228
10.2	循环神经网络	230
10.2.1	导师驱动过程和输出循环网络	232
10.2.2	计算循环神经网络的梯度	233
10.2.3	作为有向图模型的循环网络	235
10.2.4	基于上下文的 RNN 序列建模	237
10.3	双向 RNN	239
10.4	基于编码-解码的序列到序列架构	240
10.5	深度循环网络	242
10.6	递归神经网络	243

10.7 长期依赖的挑战	244
10.8 回声状态网络	245
10.9 渗漏单元和其他多时间尺度的策略	247
10.9.1 时间维度的跳跃连接	247
10.9.2 渗漏单元和一系列不同时间尺度	247
10.9.3 删除连接	248
10.10 长短期记忆和其他门控 RNN	248
10.10.1 LSTM	248
10.10.2 其他门控 RNN	250
10.11 优化长期依赖	251
10.11.1 截断梯度	251
10.11.2 引导信息流的正则化	252
10.12 外显记忆	253
第 11 章 实践方法论	256
11.1 性能度量	256
11.2 默认的基准模型	258
11.3 决定是否收集更多数据	259
11.4 选择超参数	259
11.4.1 手动调整超参数	259
11.4.2 自动超参数优化算法	262
11.4.3 网格搜索	262
11.4.4 随机搜索	263
11.4.5 基于模型的超参数优化	264
11.5 调试策略	264
11.6 示例：多位数字识别	267
第 12 章 应用	269
12.1 大规模深度学习	269
12.1.1 快速的 CPU 实现	269
12.1.2 GPU 实现	269
12.1.3 大规模的分布式实现	271
12.1.4 模型压缩	271
12.1.5 动态结构	272
12.1.6 深度网络的专用硬件实现	273
12.2 计算机视觉	274
12.2.1 预处理	275
12.2.2 数据集增强	277
12.3 语音识别	278
12.4 自然语言处理	279
12.4.1 n -gram	280
12.4.2 神经语言模型	281

12.4.3 高维输出	282
12.4.4 结合 n -gram 和神经语言模型	286
12.4.5 神经机器翻译	287
12.4.6 历史展望	289
12.5 其他应用	290
12.5.1 推荐系统	290
12.5.2 知识表示、推理和回答	292

第 3 部分 深度学习研究

第 13 章 线性因子模型	297
13.1 概率 PCA 和因子分析	297
13.2 独立成分分析	298
13.3 慢特征分析	300
13.4 稀疏编码	301
13.5 PCA 的流形解释	304
第 14 章 自编码器	306
14.1 欠完备自编码器	306
14.2 正则自编码器	307
14.2.1 稀疏自编码器	307
14.2.2 去噪自编码器	309
14.2.3 惩罚导数作为正则	309
14.3 表示能力、层的大小和深度	310
14.4 随机编码器和解码器	310
14.5 去噪自编码器详解	311
14.5.1 得分估计	312
14.5.2 历史展望	314
14.6 使用自编码器学习流形	314
14.7 收缩自编码器	317
14.8 预测稀疏分解	319
14.9 自编码器的应用	319
第 15 章 表示学习	321
15.1 贪心逐层无监督预训练	322
15.1.1 何时以及为何无监督预训练有效有效	323
15.2 迁移学习和领域自适应	326
15.3 半监督解释因果关系	329
15.4 分布式表示	332
15.5 得益于深度的指数增益	336
15.6 提供发现潜在原因的线索	337

第 16 章 深度学习中的结构化概率模型	339
16.1 非结构化建模的挑战	339
16.2 使用图描述模型结构	342
16.2.1 有向模型	342
16.2.2 无向模型	344
16.2.3 配分函数	345
16.2.4 基于能量的模型	346
16.2.5 分离和 d-分离	347
16.2.6 在有向模型和无向模型中转换	350
16.2.7 因子图	352
16.3 从图模型中采样	353
16.4 结构化建模的优势	353
16.5 学习依赖关系	354
16.6 推断和近似推断	354
16.7 结构化概率模型的深度学习方法	355
16.7.1 实例：受限玻尔兹曼机	356
第 17 章 蒙特卡罗方法	359
17.1 采样和蒙特卡罗方法	359
17.1.1 为什么需要采样	359
17.1.2 蒙特卡罗采样的基础	359
17.2 重要采样	360
17.3 马尔可夫链蒙特卡罗方法	362
17.4 Gibbs 采样	365
17.5 不同的峰值之间的混合挑战	365
17.5.1 不同峰值之间通过回火来混合	367
17.5.2 深度也许会有助于混合	368
第 18 章 直面配分函数	369
18.1 对数似然梯度	369
18.2 随机最大似然和对比散度	370
18.3 伪似然	375
18.4 得分匹配和比率匹配	376
18.5 去噪得分匹配	378
18.6 噪声对比估计	378
18.7 估计配分函数	380
18.7.1 退火重要采样	382
18.7.2 桥式采样	384
第 19 章 近似推断	385
19.1 把推断视作优化问题	385
19.2 期望最大化	386
19.3 最大后验推断和稀疏编码	387

19.4 变分推断和变分学习	389
19.4.1 离散型潜变量	390
19.4.2 变分法	394
19.4.3 连续型潜变量	396
19.4.4 学习和推断之间的相互作用	397
19.5 学成近似推断	397
19.5.1 醒眠算法	398
19.5.2 学成推断的其他形式	398
第 20 章 深度生成模型	399
20.1 玻尔兹曼机	399
20.2 受限玻尔兹曼机	400
20.2.1 条件分布	401
20.2.2 训练受限玻尔兹曼机	402
20.3 深度信念网络	402
20.4 深度玻尔兹曼机	404
20.4.1 有趣的性质	406
20.4.2 DBM 均匀场推断	406
20.4.3 DBM 的参数学习	408
20.4.4 逐层预训练	408
20.4.5 联合训练深度玻尔兹曼机	410
20.5 实值数据上的玻尔兹曼机	413
20.5.1 Gaussian-Bernoulli RBM	413
20.5.2 条件协方差的无向模型	414
20.6 卷积玻尔兹曼机	417
20.7 用于结构化或序列输出的玻尔兹曼机	418
20.8 其他玻尔兹曼机	419
20.9 通过随机操作的反向传播	419
20.9.1 通过离散随机操作的反向传播	420
20.10 有向生成网络	422
20.10.1 sigmoid 信念网络	422
20.10.2 可微生成器网络	423
20.10.3 变分自编码器	425
20.10.4 生成式对抗网络	427
20.10.5 生成矩匹配网络	429
20.10.6 卷积生成网络	430
20.10.7 自回归网络	430
20.10.8 线性自回归网络	430
20.10.9 神经自回归网络	431
20.10.10 NADE	432
20.11 从自编码器采样	433

20.11.1	与任意去噪自编码器相关的马尔可夫链	434
20.11.2	夹合与条件采样	434
20.11.3	回退训练过程	435
20.12	生成随机网络	435
20.12.1	判别性 GSN	436
20.13	其他生成方案	436
20.14	评估生成模型	437
20.15	结论	438
参考文献		439
索引		486

www.epubit.com.cn

第1章 引言

远在古希腊时期，发明家就梦想着创造能自主思考的机器。神话人物皮格马利翁 (Pygmalion)、代达罗斯 (Daedalus) 和赫淮斯托斯 (Hephaestus) 可以被看作传说中的发明家，而加拉蒂亚 (Galatea)、塔洛斯 (Talos) 和潘多拉 (Pandora) 则可以被视为人造生命 (Ovid and Martin, 2004; Sparkes, 1996; Tandy, 1997)。

当人类第一次构思可编程计算机时，就已经在思考计算机能否变得智能（尽管这距造出第一台计算机还有一百多年）(Lovelace, 1842)。如今，人工智能 (artificial intelligence, AI) 已经成为一个具有众多实际应用和活跃研究课题的领域，并且正在蓬勃发展。我们期望通过智能软件自动地处理常规劳动、理解语音或图像、帮助医学诊断和支持基础科学的研究。

在人工智能的早期，那些对人类智力来说非常困难、但对计算机来说相对简单的问题得到迅速解决，比如，那些可以通过一系列形式化的数学规则来描述的问题。人工智能的真正挑战在于解决那些对人来说很容易执行、但很难形式化描述的任务，如识别人们所说的话或图像中的脸。对于这些问题，我们人类往往可以凭借直觉轻易地解决。

针对这些比较直观的问题，本书讨论一种解决方案。该方案可以让计算机从经验中学习，并根据层次化的概念体系来理解世界，而每个概念则通过与某些相对简单的概念之间的关系来定义。让计算机从经验获取知识，可以避免由人类来给计算机形式化地指定它需要的所有知识。层次化的概念让计算机构建较简单的概念来学习复杂概念。如果绘制出表示这些概念如何建立在彼此之上的图，我们将得到一张“深”（层次很多）的图。基于这个原因，我们称这种方法为 **AI 深度学习** (deep learning)。

AI 许多早期的成功发生在相对朴素且形式化的环境中，而且不要求计算机具备很多关于世界的知识。例如，IBM 的深蓝 (Deep Blue) 国际象棋系统在 1997 年击败了世界冠军 Garry Kasparov (Hsu, 2002)。显然国际象棋是一个非常简单的领域，因为它仅含有 64 个位置并只能以严格限制的方式移动 32 个棋子。设计一种成功的国际象棋策略是巨大的成就，但向计算机描述棋子及其允许的走法并不是这一挑战的困难所在。国际象棋完全可以由一个非常简短的、完全形式化的规则列表来描述，并可以容易地由程序员事先准备好。

具有讽刺意义的是，抽象和形式化的任务对人类而言是最困难的脑力任务之一，但对计算机而言却属于最容易的。计算机早就能够打败人类最好的国际象棋选手，但直到最近计算机才在识别对象或语音任务中达到人类平均水平。一个人的日常生活需要关于世界的巨量知识。很多这方面的知识是主观的、直观的，因此很难通过形式化的方式表达清楚。计算机需要获取同样的知识才能表现出智能。人工智能的一个关键挑战就是如何将这些非形式化的知识传达给计算机。

一些人工智能项目力求将关于世界的知识用形式化的语言进行硬编码 (hard-code)。计算机可以使用逻辑推理规则来自动地理解这些形式化语言中的声明。这就是众所周知的人工智能的知识库 (knowledge base) 方法。然而，这些项目最终都没有取得重大的成功。其中最著名的项目是 Cyc (Lenat and Guha, 1989)。Cyc 包括一个推断引擎和一个使用 CycL 语言描述的声明数据库。这些声明是由人类监督者输入的。这是一个笨拙的过程。人们设法设

计出足够复杂的形式化规则来精确地描述世界。例如，Cyc 不能理解一个关于名为 Fred 的人在早上剃须的故事 (Linde, 1992)。它的推理引擎检测到故事中的不一致性：它知道人体的构成不包含电气零件，但由于 Fred 正拿着一个电动剃须刀，它认为实体——“正在剃须的 Fred”(“FredWhileShaving”) 含有电气部件。因此，它产生了这样的疑问——Fred 在刮胡子的时候是否仍然是一个人。

依靠硬编码的知识体系面临的困难表明，AI 系统需要具备自己获取知识的能力，即从原始数据中提取模式的能力。这种能力称为机器学习(machine learning)。引入机器学习使计算机能够解决涉及现实世界知识的问题，并能做出看似主观的决策。比如，一个称为逻辑回归(logistic regression)的简单机器学习算法可以决定是否建议剖腹产 (Mor-Yosef *et al.*, 1990)。而同样是简单机器学习算法的朴素贝叶斯(naive Bayes)则可以区分垃圾电子邮件和合法电子邮件。

这些简单的机器学习算法的性能在很大程度上依赖于给定数据的表示(representation)。例如，当逻辑回归用于判断产妇是否适合剖腹产时，AI 系统不会直接检查患者。相反，医生需要告诉系统几条相关的信息，诸如是否存在子宫疤痕。表示患者的每条信息称为一个特征。逻辑回归学习病人的这些特征如何与各种结果相关联。然而，它丝毫不能影响该特征定义的方式。如果将病人的 MRI(核磁共振) 扫描而不是医生正式的报告作为逻辑回归的输入，它将无法做出有用的预测。MRI 扫描的单一像素与分娩过程中并发症之间的相关性微乎其微。

在整个计算机科学乃至日常生活中，对表示的依赖都是一个普遍现象。在计算机科学中，如果数据集合被精巧地结构化并被智能地索引，那么诸如搜索之类的操作的处理速度就可以成指级地加快。人们可以很容易地在阿拉伯数字的表示下进行算术运算，但在罗马数字的表示下，运算会比较耗时。因此，毫不奇怪，表示的选择会对机器学习算法的性能产生巨大的影响。图 1.1 展示了一个简单的可视化例子。

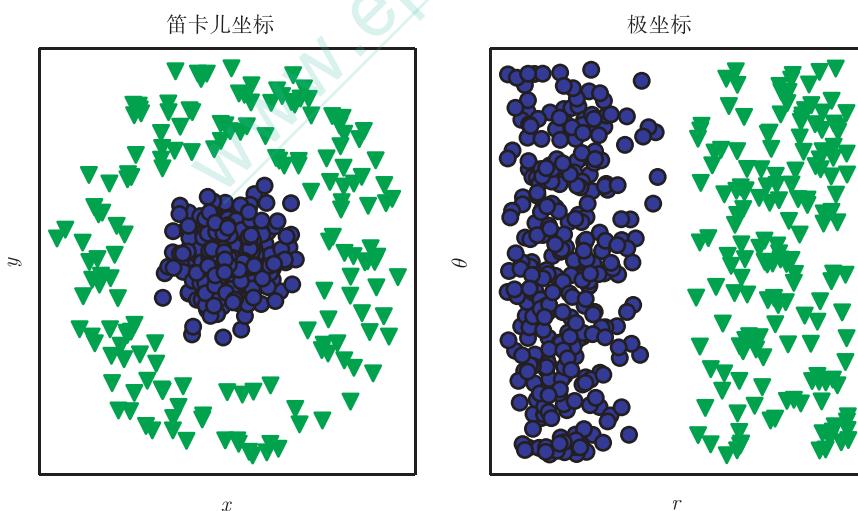


图 1.1 不同表示的例子：假设我们想在散点图中画一条线来分隔两类数据。在左图中，我们使用笛卡儿坐标表示数据，这个任务是不可能的。在右图中，我们用极坐标表示数据，可以用垂直线简单地解决这个任务 (与 David Warde-Farley 合作绘制此图)

许多人工智能任务都可以通过以下方式解决：先提取一个合适的特征集，然后将这些特

征提供给简单的机器学习算法。例如，对于通过声音鉴别说话者的任务来说，一个有用的特点是对其声道大小的估计。这个特征为判断说话者是男性、女性还是儿童提供了有力线索。

然而，对于许多任务来说，我们很难知道应该提取哪些特征。例如，假设我们想编写一个程序来检测照片中的车。我们知道，汽车有轮子，所以我们可能会想用车轮的存在与否作为特征。遗憾的是，我们难以准确地根据像素值来描述车轮看上去像什么。虽然车轮具有简单的几何形状，但它的图像可能会因场景而异，如落在车轮上的阴影、太阳照亮的车轮的金属零件、汽车的挡泥板或者遮挡的车轮一部分的前景物体等。

解决这个问题的途径之一是使用机器学习来发掘表示本身，而不仅仅把表示映射到输出。这种方法我们称之为**表示学习**(representation learning)。学习到的表示往往比手动设计的表示表现得更好。并且它们只需最少的人工干预，就能让AI系统迅速适应新的任务。表示学习算法只需几分钟就可以为简单的任务发现一个很好的特征集，对于复杂任务则需要几小时到几个月。手动为一个复杂的任务设计特征需要耗费大量的人工、时间和精力，甚至需要花费整个社群研究人员几十年的时间。

表示学习算法的典型例子是**自编码器**(autoencoder)。自编码器由一个**编码器**(encoder)函数和一个**解码器**(decoder)函数组合而成。编码器函数将输入数据转换为一种不同的表示，而解码器函数则将这个新的表示转换回原来的形式。我们期望当输入数据经过编码器和解码器之后尽可能多地保留信息，同时希望新的表示有各种好的特性，这也是自编码器的训练目标。为了实现不同的特性，我们可以设计不同形式的自编码器。

当设计特征或设计用于学习特征的算法时，我们的目标通常是分离出能解释观察数据的**变差因素**(factors of variation)。在此背景下，“因素”这个词仅指代影响的不同来源；因素通常不是乘性组合。这些因素通常是不能被直接观察到的量。相反，它们可能是现实世界中观察不到的物体或者不可观测的力，但会影响可观测的量。为了对观察到的数据提供有用的简化解释或推断其原因，它们还可能以概念的形式存在于人类的思维中。它们可以被看作数据的概念或者抽象，帮助我们了解这些数据的丰富多样性。当分析语音记录时，变差因素包括说话者的年龄、性别、他们的口音和他们正在说的词语。当分析汽车的图像时，变差因素包括汽车的位置、它的颜色、太阳的角度和亮度。

在许多现实的人工智能应用中，困难主要源于多个变差因素同时影响着我们能够观察到的每一个数据。比如，在一张包含红色汽车的图片中，其单个像素在夜间可能会非常接近黑色。汽车轮廓的形状取决于视角。大多数应用需要我们理清变差因素并忽略我们不关心的因素。

显然，从原始数据中提取如此高层次、抽象的特征是非常困难的。许多诸如说话口音这样的变差因素，只能通过对数据进行复杂的、接近人类水平的理解来辨识。这几乎与获得原问题的表示一样困难，因此，乍一看，表示学习似乎并不能帮助我们。

深度学习(deep learning) 通过其他较简单的表示来表达复杂表示，解决了表示学习中的核心问题。

深度学习让计算机通过较简单的概念构建复杂的概念。图 1.2 展示了深度学习系统如何通过组合较简单的概念（例如角和轮廓，它们反过来由边线定义）来表示图像中人的概念。深度学习模型的典型例子是前馈深度网络或或多层感知机(multilayer perceptron, MLP)。多层感知机仅仅是一个将一组输入值映射到输出值的数学函数。该函数由许多较简单的函数复合而成。我们可以认为不同数学函数的每一次应用都为输入提供了新的表示。

学习数据的正确表示的想法是解释深度学习的一个视角。另一个视角是深度促使计算机学习一个多步骤的计算机程序。每一层表示都可以被认为是并行执行另一组指令之后计算机的存储器状态。更深的网络可以按顺序执行更多的指令。顺序指令提供了极大的能力，因为后面的指令可以参考早期指令的结果。从这个角度上看，在某层激活函数里，并非所有信息都蕴涵着解释输入的变差因素。表示还存储着状态信息，用于帮助程序理解输入。这里的状态信息类似于传统计算机程序中的计数器或指针。它与具体的输入内容无关，但有助于模型组织其处理过程。

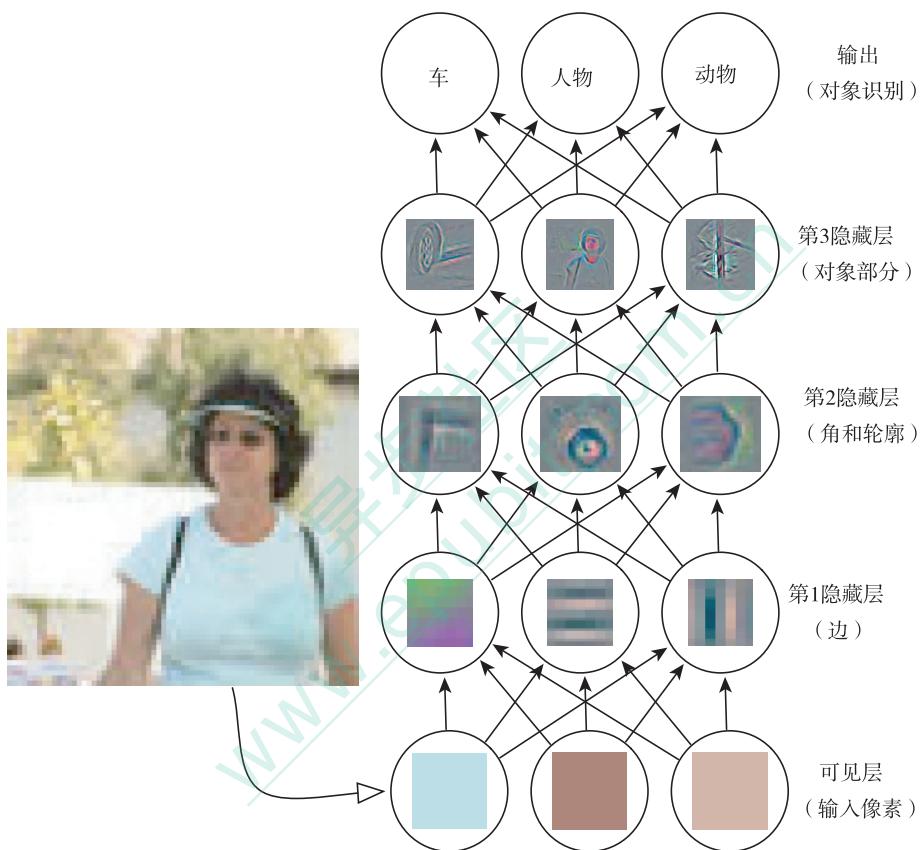


图 1.2 深度学习模型的示意图。计算机难以理解原始感观输入数据的含义，如表示为像素值集合的图像。将一组像素映射到对象标识的函数非常复杂。如果直接处理，学习或评估此映射似乎是不可能的。深度学习将所需的复杂映射分解为一系列嵌套的简单映射（每个由模型的不同层描述）来解决这一难题。输入展示在可见层(visible layer)，这样命名的原因是因为它包含我们能观察到的变量。然后是一系列从图像中提取越来越多抽象特征的隐藏层(hidden layer)。因为它们的值不在数据中给出，所以将这些层称为“隐藏层”；模型必须确定哪些概念有利于解释观察数据中的关系。这里的图像是每个隐藏单元表示的特征的可视化。给定像素，第 1 层可以轻易地通过比较相邻像素的亮度来识别边缘。有了第 1 隐藏层描述的边缘，第 2 隐藏层可以容易地搜索可识别为角和扩展轮廓的边集合。给定第 2 隐藏层中关于角和轮廓的图像描述，第 3 隐藏层可以找到轮廓和角的特定集合来检测特定对象的整个部分。最后，根据图像描述中包含的对象部分，可以识别图像中存在的对象（经 Zeiler and Fergus (2014) 许可引用此图）

目前主要有两种度量模型深度的方式。一种方式是基于评估架构所需执行的顺序指令的

数目。假设我们将模型表示为给定输入后，计算对应输出的流程图，则可以将这张流程图中的最长路径视为模型的深度。正如两个使用不同语言编写的等价程序将具有不同的长度，相同的函数可以被绘制为具有不同深度的流程图，其深度取决于我们可以用来作为一个步骤的函数。图 1.3 说明了语言的选择如何给相同的架构两个不同的衡量。

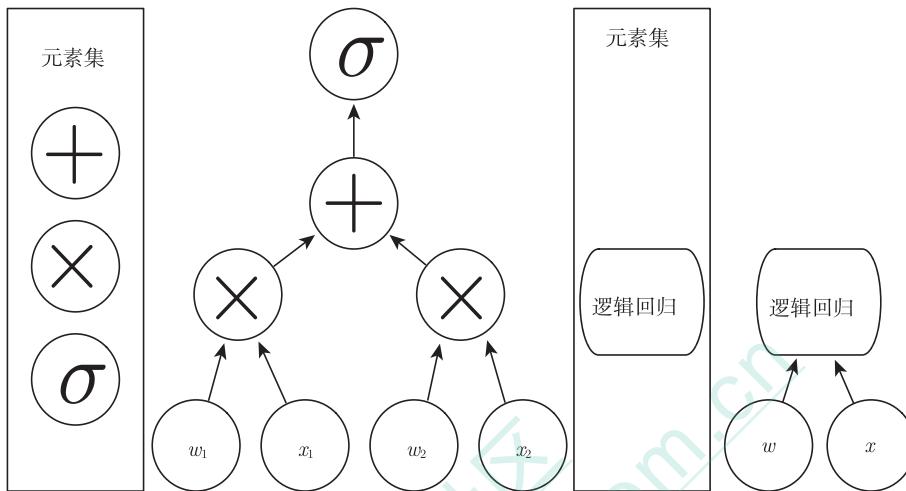


图 1.3 将输入映射到输出的计算图表的示意图，其中每个节点执行一个操作。深度是从输入到输出的最长路径的长度，但这取决于可能的计算步骤的定义。这些图中所示的计算是逻辑回归模型的输出， $\sigma(w^T x)$ ，其中 σ 是 logistic sigmoid 函数。如果使用加法、乘法和 logistic sigmoid 作为计算机语言的元素，那么这个模型深度为 3；如果将逻辑回归视为元素本身，那么这个模型深度为 1

另一种是在深度概率模型中使用的方法，它不是将计算图的深度视为模型深度，而是将描述概念彼此如何关联的图的深度视为模型深度。在这种情况下，计算每个概念表示的计算流程图的深度可能比概念本身的图更深。这是因为系统对较简单概念的理解在给出更复杂概念的信息后可以进一步精细化。例如，一个 AI 系统观察其中一只眼睛在阴影中的脸部图像时，它最初可能只看到一只眼睛。但当检测到脸部的存在后，系统可以推断第二只眼睛也可能是存在的。在这种情况下，概念的图仅包括两层（关于眼睛的层和关于脸的层），但如果我们将每个概念的估计将需要额外的 n 次计算，那么计算的图将包含 $2n$ 层。

由于并不总是清楚计算图的深度和概率模型图的深度哪一个是最重要的，并且由于不同的人选择不同的最小元素集来构建相应的图，所以就像计算机程序的长度不存在单一的正确值一样，架构的深度也不存在单一的正确值。另外，也不存在模型多么深才能被修饰为“深”的共识。但相比传统机器学习，深度学习研究的模型涉及更多学到功能或学到概念的组合，这点毋庸置疑。

总之，这本书的主题——深度学习是通向人工智能的途径之一。具体来说，它是机器学习的一种，一种能够使计算机系统从经验和数据中得到提高的技术。我们坚信机器学习可以构建出在复杂实际环境下运行的 AI 系统，并且是唯一切实可行的方法。深度学习是一种特定类型的机器学习，具有强大的能力和灵活性，它将大千世界表示为嵌套的层次概念体系（由较简单概念间的联系定义复杂概念、从一般抽象概括到高级抽象表示）。图 1.4 说明了这些不同的 AI 学科之间的关系。图 1.5 展示了每个学科如何工作的高层次原理。

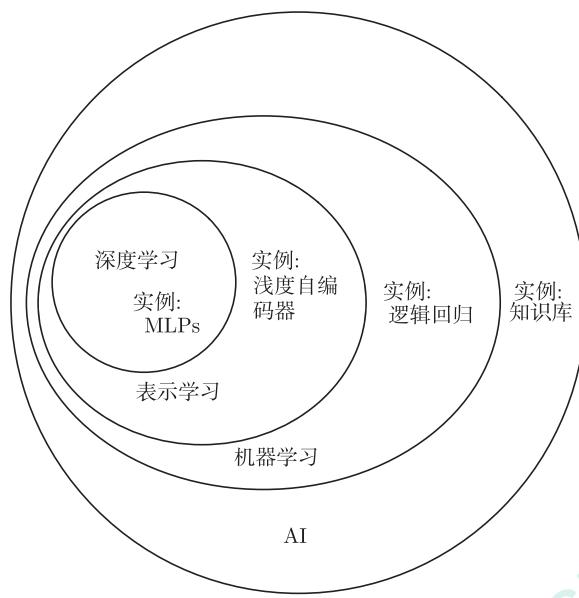


图 1.4 维恩图展示了深度学习既是一种表示学习, 也是一种机器学习, 可以用于许多 (但不是全部)AI 方法。维恩图的每个部分包括一个 AI 技术的实例

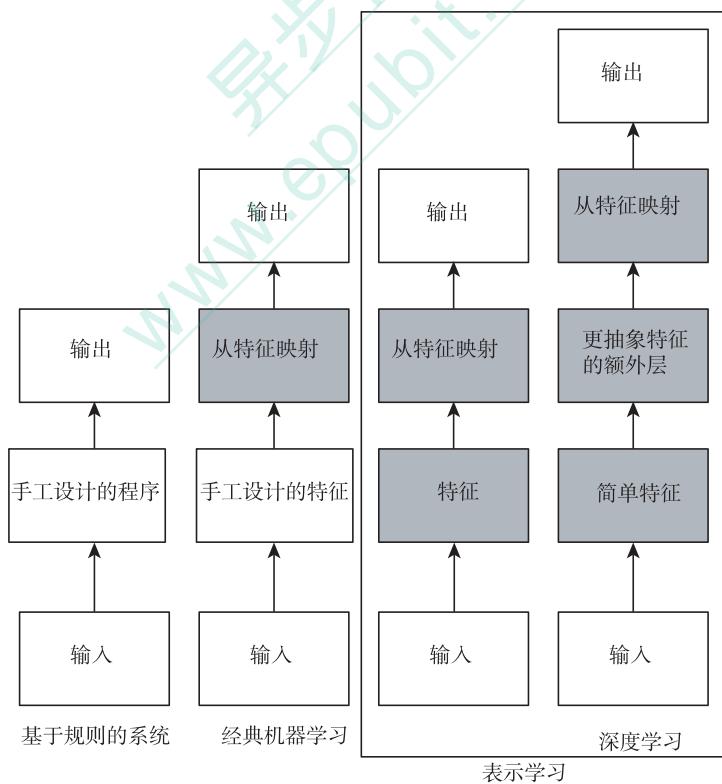


图 1.5 流程图展示了 AI 系统的不同部分如何在不同的 AI 学科中彼此相关。阴影框表示能从数据中学习的组件

1.1 本书面向的读者

本书对各类读者都有一定的用处，但主要是为两类受众而写的。其中，一类受众是学习机器学习的大学生（本科或研究生），包括那些已经开始职业生涯的深度学习和人工智能研究者。另一类受众是没有机器学习或统计背景，但希望能快速地掌握这方面知识，并在他们的产品或平台中使用深度学习的软件工程师。现已证明，深度学习在许多软件领域都是有用的，包括计算机视觉、语音和音频处理、自然语言处理、机器人技术、生物信息学和化学、电子游戏、搜索引擎、网络广告和金融。

为了更好地服务各类读者，我们将本书组织为 3 个部分。第 1 部分介绍基本的数学工具和机器学习的概念。第 2 部分介绍最成熟的深度学习算法，这些技术基本上已经得到解决。第 3 部分讨论某些具有展望性的想法，它们被广泛地认为是深度学习未来的研究重点。

读者可以随意跳过不感兴趣或与自己背景不相关的部分。熟悉线性代数、概率和基本机器学习概念的读者可以跳过第 1 部分。若读者只是想实现一个能工作的系统，则不需要阅读超出第 2 部分的内容。为了帮助读者选择章节，图 1.6 给出了本书高层组织结构的流程图。

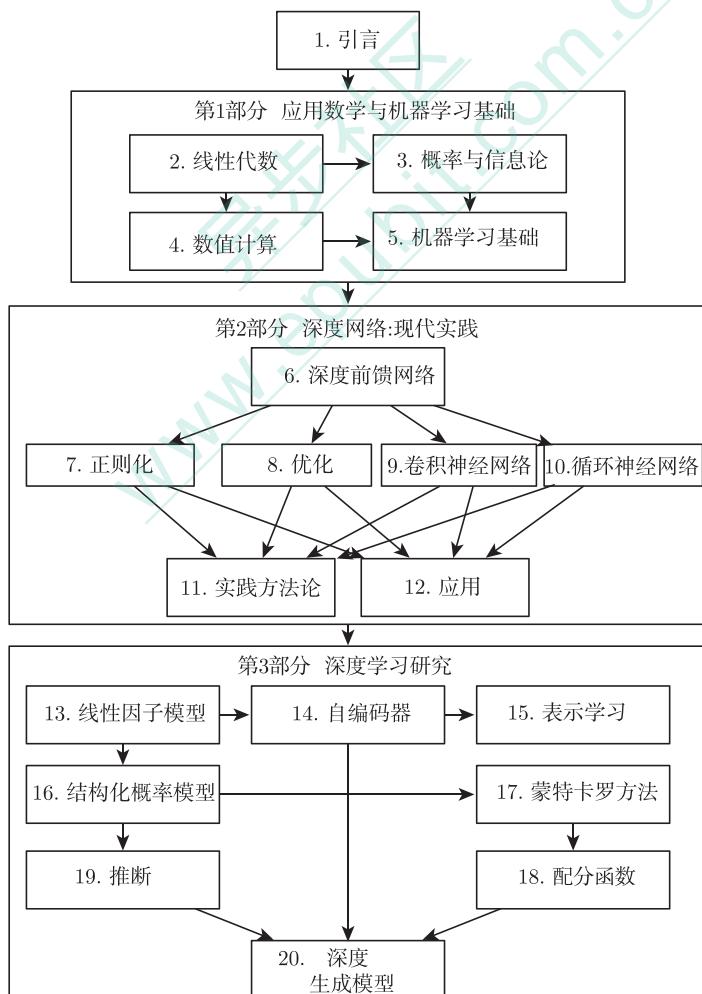


图 1.6 本书的高层组织结构的流程图。从一章到另一章的箭头表示前一章是理解后一章的必备内容

我们假设所有读者都具备计算机科学背景。也假设读者熟悉编程，并且对计算的性能问题、复杂性理论、入门级微积分和一些图论术语有基本的了解。

《深度学习》英文版配套网站是 www.deeplearningbook.org。网站上提供了各种补充材料，包括练习、讲义幻灯片、错误更正以及其他应该对读者和讲师有用的资源。

《深度学习》中文版的读者，可访问人民邮电出版社异步社区网站 www.epubit.com.cn，获取更多图书信息。

1.2 深度学习的历史趋势

通过历史背景了解深度学习是最简单的方式。这里我们仅指出深度学习的几个关键趋势，而不是提供其详细的历史：

- 深度学习有着悠久而丰富的历史，但随着许多不同哲学观点的渐渐消逝，与之对应的名称也渐渐尘封。
- 随着可用的训练数据量不断增加，深度学习变得更加有用。
- 随着时间的推移，针对深度学习的计算机软硬件基础设施都有所改善，深度学习模型的规模也随之增长。
- 随着时间的推移，深度学习已经解决日益复杂的应用，并且精度不断提高。

1.2.1 神经网络的众多名称和命运变迁

我们期待这本书的许多读者都听说过深度学习这一激动人心的新技术，并对一本书提及一个新兴领域的“历史”而感到惊讶。事实上，深度学习的历史可以追溯到 20 世纪 40 年代。深度学习看似是一个全新的领域，只不过因为在目前流行的前几年它还是相对冷门的，同时也因为它被赋予了许多不同的名称（其中大部分已经不再使用），最近才成为众所周知的“深度学习”。这个领域已经更换了很多名称，它反映了不同的研究人员和不同观点的影响。

全面地讲述深度学习的历史超出了本书的范围。然而，一些基本的背景对理解深度学习是有用的。一般认为，迄今为止深度学习已经经历了 3 次发展浪潮：20 世纪 40 年代到 60 年代，深度学习的雏形出现在控制论(cybernetics) 中；20 世纪 80 年代到 90 年代，深度学习表现为联结主义(connectionism)；直到 2006 年，才真正以深度学习之名复兴。图 1.7 给出了定量的展示。

我们今天知道的一些最早的学习算法，旨在模拟生物学习的计算模型，即大脑怎样学习或为什么能学习的模型。其结果是深度学习以人工神经网络(artificial neural network, ANN) 之名而淡去。彼时，深度学习模型被认为是受生物大脑（无论人类大脑或其他动物的大脑）所启发而设计出来的系统。尽管有些机器学习的神经网络有时被用来理解大脑功能 (Hinton and Shallice, 1991)，但它们一般都没有设计成生物功能的真实模型。深度学习的神经观点受两个主要思想启发：一个想法是，大脑作为例子证明智能行为是可能的，因此，概念上，建立智能的直接途径是逆向大脑背后的计算原理，并复制其功能；另一种看法是，理解大脑和人类智能背后的原理也非常有趣，因此机器学习模型除了解决工程应用的能力，如果能让人类对这些基本的科学问题有进一步的认识，也将会很有用。

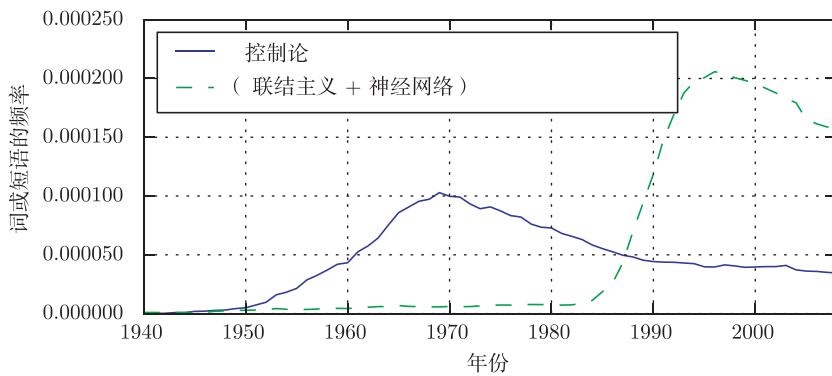


图 1.7 根据 Google 图书中短语“控制论”“联结主义”或“神经网络”频率衡量的人工神经网络研究的历史浪潮（图中展示了 3 次浪潮的前两次，第 3 次最近才出现）。第 1 次浪潮开始于 20 世纪 40 年代到 20 世纪 60 年代的控制论，随着生物学习理论的发展 (McCulloch and Pitts, 1943; Hebb, 1949) 和第一个模型的实现 (如感知机 (Rosenblatt, 1958))，能实现单个神经元的训练。第 2 次浪潮开始于 1980—1995 年间的联结主义方法，可以使用反向传播 (Rumelhart *et al.*, 1986a) 训练具有一个或两个隐藏层的神经网络。当前第 3 次浪潮，也就是深度学习，大约始于 2006 年 (Hinton *et al.*, 2006a; Bengio *et al.*, 2007a; Ranzato *et al.*, 2007a)，并且于 2016 年以图书的形式出现。另外，前两次浪潮类似地出现在书中的时间比相应的科学活动晚得多

现代术语“深度学习”超越了目前机器学习模型的神经科学观点。它诉诸于学习多层次组合这一更普遍的原理，这一原理也可以应用于那些并非受神经科学启发的机器学习框架。

现代深度学习最早的前身是从神经科学的角度出发的简单线性模型。这些模型设计为使用一组 n 个输入 x_1, \dots, x_n ，并将它们与一个输出 y 相关联。这些模型希望学习一组权重 w_1, \dots, w_n ，并计算它们的输出 $f(\mathbf{x}, \mathbf{w}) = x_1 w_1 + \dots + x_n w_n$ 。如图 1.7 所示，第一次神经网络研究浪潮称为控制论。

McCulloch-Pitts 神经元 (McCulloch and Pitts, 1943) 是脑功能的早期模型。该线性模型通过检验函数 $f(\mathbf{x}, \mathbf{w})$ 的正负来识别两种不同类别的输入。显然，模型的权重需要正确设置后才能使模型的输出对应于期望的类别。这些权重可以由操作人员设定。20 世纪 50 年代，感知机 (Rosenblatt, 1956, 1958) 成为第一个能根据每个类别的输入样本来学习权重的模型。大约在同一时期，自适应线性单元 (adaptive linear element, ADALINE) 简单地返回函数 $f(\mathbf{x})$ 本身的值来预测一个实数 (Widrow and Hoff, 1960)，并且它还可以学习从数据预测这些数。

这些简单的学习算法大大影响了机器学习的现代景象。用于调节 ADALINE 权重的训练算法是被称为随机梯度下降 (stochastic gradient descent) 的一种特例。稍加改进后的随机梯度下降算法仍然是当今深度学习的主要训练算法。

基于感知机和 ADALINE 中使用的函数 $f(\mathbf{x}, \mathbf{w})$ 的模型称为线性模型 (linear model)。尽管在许多情况下，这些模型以不同于原始模型的方式进行训练，但仍是目前最广泛使用的机器学习模型。

线性模型有很多局限性。最著名的是，它们无法学习异或 (XOR) 函数，即 $f([0, 1], \mathbf{w}) = 1$ 和 $f([1, 0], \mathbf{w}) = 1$ ，但 $f([1, 1], \mathbf{w}) = 0$ 和 $f([0, 0], \mathbf{w}) = 0$ 。观察到线性模型这个缺陷的批评者对受生物学启发的学习普遍地产生了抵触 (Minsky and Papert, 1969)。这导致了神经网络热潮的第一次大衰退。

现在, 神经科学被视为深度学习研究的一个重要灵感来源, 但它已不再是该领域的主要指导。

如今神经科学在深度学习研究中的作用被削弱, 主要原因是根本没有足够的关于大脑的信息来作为指导去使用它。要获得对被大脑实际使用算法的深刻理解, 我们需要有能力同时监测(至少是)数千相连神经元的活动。我们不能够做到这一点, 所以我们甚至连大脑最简单、最深入研究的部分都还远远没有理解(Olshausen and Field, 2005)。

神经科学已经给了我们依靠单一深度学习算法解决许多不同任务的理由。神经学家们发现, 如果将雪貂的大脑重新连接, 使视觉信号传送到听觉区域, 它们可以学会用大脑的听觉处理区域去“看”(Von Melchner *et al.*, 2000)。这暗示着大多数哺乳动物的大脑使用单一的算法就可以解决其大脑可以解决的大部分不同任务。在这个假设之前, 机器学习研究是比较分散的, 研究人员在不同的社群研究自然语言处理、计算机视觉、运动规划和语音识别。如今, 这些应用社群仍然是独立的, 但是对于深度学习研究团体来说, 同时研究许多甚至所有这些应用领域是很常见的。

我们能够从神经科学得到一些粗略的指南。仅通过计算单元之间的相互作用而变得智能的基本思想是受大脑启发的。新认知机(Fukushima, 1980)受哺乳动物视觉系统的结构启发, 引入了一个处理图片的强大模型架构, 它后来成为了现代卷积网络的基础(LeCun *et al.*, 1998c)(参见第 9.10 节)。目前大多数神经网络是基于一个称为整流线性单元(rectified linear unit)的神经单元模型。原始认知机(Fukushima, 1975)受我们关于大脑功能知识的启发, 引入了一个更复杂的版本。简化的现代版通过吸收来自不同观点的思想而形成, Nair and Hinton(2010b) 和 Glorot *et al.*(2011a)援引神经科学作为影响, Jarrett *et al.*(2009a)援引更多面向工程的影响。虽然神经科学是灵感的重要来源, 但它不需要被视为刚性指导。我们知道, 真实的神经元计算着与现代整流线性单元非常不同的函数, 但更接近真实神经网络的系统并没有导致机器学习性能的提升。此外, 虽然神经科学已经成功地启发了一些神经网络架构, 但我们对用于神经科学的生物学习还没有足够多的了解, 因此也就不能为训练这些架构用的学习算法提供太多的借鉴。

媒体报道经常强调深度学习与大脑的相似性。的确, 深度学习研究者比其他机器学习领域(如核方法或贝叶斯统计)的研究者更可能地引用大脑作为影响, 但是大家不应该认为深度学习在尝试模拟大脑。现代深度学习从许多领域获取灵感, 特别是应用数学的基本内容, 如线性代数、概率论、信息论和数值优化。尽管一些深度学习的研究人员引用神经科学作为灵感的重要来源, 然而其他学者完全不关心神经科学。

值得注意的是, 了解大脑是如何在算法层面上工作的尝试确实存在且发展良好。这项尝试主要被称为“计算神经科学”, 并且是独立于深度学习的领域。研究人员在两个领域之间来回研究是很常见的。深度学习领域主要关注如何构建计算机系统, 从而成功解决需要智能才能解决的任务, 而计算神经科学领域主要关注构建大脑如何真实工作的、比较精确的模型。

20世纪80年代, 神经网络研究的第二次浪潮在很大程度上是伴随一个被称为联结主义(connectionism)或并行分布处理(parallel distributed processing)潮流而出现的(Rumelhart *et al.*, 1986d; McClelland *et al.*, 1995)。联结主义是在认知科学的背景下出现的。认知科学是理解思维的跨学科途径, 即它融合多个不同的分析层次。20世纪80年代初期, 大多数认知科学家研究符号推理模型。尽管这很流行, 但符号模型很难解释大脑如何真正使用神经元实现推理功能。联结主义者开始研究真正基于神经系统实现的认知模型(Touretzky and Minton, 1985), 其中

很多复苏的想法可以追溯到心理学家 Donald Hebb 在 20 世纪 40 年代的工作 (Hebb, 1949)。

联结主义的中心思想是, 当网络将大量简单的计算单元连接在一起时可以实现智能行为。这种见解同样适用于生物神经系统中的神经元, 因为它和计算模型中隐藏单元起着类似的作用。

在 20 世纪 80 年代的联结主义期间形成的几个关键概念在今天的深度学习中仍然是非常重要的。

其中一个概念是**分布式表示**(distributed representation)(Hinton *et al.*, 1986)。其思想是: 系统的每一个输入都应该由多个特征表示, 并且每一个特征都应该参与到多个可能输入的表示。例如, 假设我们有一个能够识别红色、绿色或蓝色的汽车、卡车和鸟类的视觉系统, 表示这些输入的其中一个方法是将 9 个可能的组合: 红卡车、红汽车、红鸟、绿卡车等使用单独的神经元或隐藏单元激活。这需要 9 个不同的神经元, 并且每个神经元必须独立地学习颜色和对象身份的概念。改善这种情况的方法之一是使用分布式表示, 即用 3 个神经元描述颜色, 3 个神经元描述对象身份。这仅仅需要 6 个神经元而不是 9 个, 并且描述红色的神经元能够从汽车、卡车和鸟类的图像中学习红色, 而不仅仅是从一个特定类别的图像中学习。分布式表示的概念是本书的核心, 我们将在第 15 章中更加详细地描述。

联结主义潮流的另一个重要成就是反向传播在训练具有内部表示的深度神经网络中的成功使用以及反向传播算法的普及 (Rumelhart *et al.*, 1986c; LeCun, 1987)。这个算法虽然曾黯然失色且不再流行, 但截至写书之时, 它仍是训练深度模型的主导方法。

20 世纪 90 年代, 研究人员在使用神经网络进行序列建模的方面取得了重要进展。Hochreiter (1991b) 和 Bengio *et al.* (1994b) 指出了对长序列进行建模的一些根本性数学难题, 这将在第 10.7 节中描述。Hochreiter 和 Schmidhuber(1997) 引入**长短期记忆**(long short-term memory, LSTM) 网络来解决这些难题。如今, LSTM 在许多序列建模任务中广泛应用, 包括 Google 的许多自然语言处理任务。

神经网络研究的第二次浪潮一直持续到 20 世纪 90 年代中期。基于神经网络和其他 AI 技术的创业公司开始寻求投资, 其做法野心勃勃但不切实际。当 AI 研究不能实现这些不合理的期望时, 投资者感到失望。同时, 机器学习的其他领域取得了进步。比如, 核方法 (Boser *et al.*, 1992; Cortes and Vapnik, 1995; Schölkopf *et al.*, 1999) 和图模型 (Jordan, 1998) 都在很多重要任务上实现了很好的效果。这两个因素导致了神经网络热潮的第二次衰退, 并一直持续到 2007 年。

在此期间, 神经网络继续在某些任务上获得令人印象深刻的表现 (LeCun *et al.*, 1998c; Bengio *et al.*, 2001a)。加拿大高级研究所 (CIFAR) 通过其神经计算和自适应感知 (NCAP) 研究计划帮助维持神经网络研究。该计划联合了分别由 Geoffrey Hinton、Yoshua Bengio 和 Yann LeCun 领导的多伦多大学、蒙特利尔大学和纽约大学的机器学习研究小组。这个多学科的 CIFAR NCAP 研究计划还包括了神经科学家、人类和计算机视觉专家。

在那个时候, 人们普遍认为深度网络是难以训练的。现在我们知道, 20 世纪 80 年代就存在的算法能工作得非常好, 但是直到 2006 年前后都没有体现出来。这可能仅仅由于其计算代价太高, 而以当时可用的硬件难以进行足够的实验。

神经网络研究的第三次浪潮始于 2006 年的突破。Geoffrey Hinton 表明名为“深度信念网络”的神经网络可以使用一种称为“贪婪逐层预训练”的策略来有效地训练 (Hinton *et al.*, 2006a), 我们将在第 15.1 节中更详细地描述。其他 CIFAR 附属研究小组很快表明, 同样的策

略可以被用来训练许多其他类型的深度网络 (Bengio and LeCun, 2007a; Ranzato *et al.*, 2007b), 并能系统地帮助提高在测试样例上的泛化能力。神经网络研究的这一次浪潮普及了“深度学习”这一术语, 强调研究者现在有能力训练以前不可能训练的比较深的神经网络, 并着力于深度的理论重要性上 (Bengio and LeCun, 2007b; Delalleau and Bengio, 2011; Pascanu *et al.*, 2014a; Montufar *et al.*, 2014)。此时, 深度神经网络已经优于与之竞争的基于其他机器学习技术以及手工设计功能的 AI 系统。在写这本书的时候, 神经网络的第三次发展浪潮仍在继续, 尽管深度学习的研究重点在这一段时间内发生了巨大变化。第三次浪潮已开始着眼于新的无监督学习技术和深度模型在小数据集的泛化能力, 但目前更多的兴趣点仍是比较传统的监督学习算法和深度模型充分利用大型标注数据集的能力。

1.2.2 与日俱增的数据量

人们可能想问, 既然人工神经网络的第一个实验在 20 世纪 50 年代就完成了, 但为什么深度学习直到最近才被认为是关键技术? 自 20 世纪 90 年代以来, 深度学习就已经成功用于商业应用, 但通常被视为一种只有专家才可以使用的艺术而不是一种技术, 这种观点一直持续到最近。确实, 要从一个深度学习算法获得良好的性能需要一些技巧。幸运的是, 随着训练数据的增加, 所需的技巧正在减少。目前在复杂的任务中达到人类水平的学习算法, 与 20 世纪 80 年代努力解决玩具问题 (toy problem) 的学习算法几乎是一样的, 尽管我们使用这些算法训练的模型经历了变革, 即简化了极深架构的训练。最重要的新进展是, 现在我们有了这些算法得以成功训练所需的资源。图 1.8 展示了基准数据集的大小如何随着时间的推移而

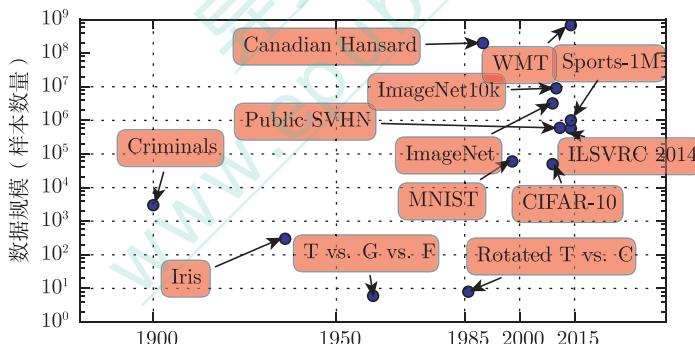


图 1.8 与日俱增的数据量。20 世纪初, 统计学家使用数百或数千的手工制作的度量来研究数据集 (Garson, 1900; Gosset, 1908; Anderson, 1935; Fisher, 1936)。20 世纪 50 年代到 80 年代, 受生物启发的机器学习开拓者通常使用小的合成数据集, 如低分辨率的字母位图, 设计为在低计算成本下表明神经网络能够学习特定功能 (Widrow and Hoff, 1960; Rumelhart *et al.*, 1986b)。20 世纪 80 年代和 90 年代, 机器学习变得更偏统计, 并开始利用包含成千上万个样本的更大数据集, 如手写扫描数字的 MNIST 数据集 (如图 1.9 所示)(LeCun *et al.*, 1998c)。在 21 世纪的第一个 10 年里, 相同大小更复杂的数据集持续出现, 如 CIFAR-10 数据集 (Krizhevsky and Hinton, 2009)。在这 10 年结束和接下来的 5 年, 明显更大的数据集 (包含数万到数千万的样例) 完全改变了深度学习可能实现的事。这些数据集包括公共 Street View House Numbers 数据集 (Netzer *et al.*, 2011)、各种版本的 ImageNet 数据集 (Deng *et al.*, 2009, 2010a; Russakovsky *et al.*, 2014a) 以及 Sports-1M 数据集 (Karpathy *et al.*, 2014)。在图顶部, 我们看到翻译句子的数据集通常远大于其他数据集, 如根据 Canadian Hansard 制作的 IBM 数据集 (Brown *et al.*, 1990) 和 WMT 2014 英法数据集 (Schwenk, 2014)。

显著增加。这种趋势是由社会日益数字化驱动的。由于我们的活动越来越多地发生在计算机上，我们做什么也越来越多地被记录。由于计算机越来越多地联网在一起，这些记录更容易集中管理，并更容易将它们整理成适于机器学习应用的数据集。因为统计估计的主要负担（观察少量数据以在新数据上泛化）已经减轻，“大数据”时代使机器学习更加容易。截至 2016 年，一个粗略的经验法则是，监督深度学习算法在每类给定约 5000 个标注样本情况下一般将达到可以接受的性能，当至少有 1000 万个标注样本的数据集用于训练时，它将达到或超过人类表现。此外，在更小的数据集上获得成功是一个重要的研究领域，为此我们应特别侧重于如何通过无监督或半监督学习充分利用大量的未标注样本。

8	9	0	1	2	3	4	7	8	9	0	1	2	3	4	5	6	7	8	6
4	2	6	4	7	5	5	4	7	8	9	2	9	3	9	3	8	2	0	5
0	1	0	4	2	6	5	3	5	3	8	0	0	3	4	1	5	3	0	8
3	0	6	2	7	1	1	8	1	7	1	3	8	9	7	6	7	4	1	6
7	5	1	7	1	9	8	0	6	9	4	9	9	3	7	1	9	2	2	5
3	7	8	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	0
1	2	3	4	5	6	7	8	9	8	1	0	5	5	1	9	0	4	1	9
3	8	4	7	7	8	5	0	6	5	5	3	3	3	9	8	1	4	0	6
1	0	0	6	2	1	1	3	2	8	8	7	8	4	6	0	2	0	3	6
8	7	1	5	9	9	3	2	4	9	4	6	5	3	2	8	5	9	4	1
6	5	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7
8	9	0	1	2	3	4	5	6	7	8	9	6	4	2	6	4	7	5	5
4	7	8	9	2	9	3	9	3	8	2	0	9	8	0	5	6	0	1	0
4	2	6	5	5	5	4	3	4	1	5	3	0	8	3	0	6	2	7	1
1	8	1	7	1	3	8	5	4	2	0	9	7	6	7	4	1	6	8	4
7	5	1	2	6	7	1	9	8	0	6	9	4	9	9	6	2	3	7	1
9	2	2	5	3	7	8	0	1	2	3	4	5	6	7	8	0	1	2	3
4	5	6	7	8	0	1	2	3	4	5	6	7	8	9	2	1	2	1	3
9	9	8	5	3	7	0	7	7	5	7	9	9	4	7	0	3	4	1	4
4	7	5	8	1	4	8	4	1	8	6	4	4	6	3	5	7	2	5	9

图 1.9 MNIST 数据集的输入样例。“NIST”代表国家标准和技术研究所 (National Institute of Standards and Technology)，是最初收集这些数据的机构。“M”代表“修改的 (Modified)”，为更容易地与机器学习算法一起使用，数据已经过预处理。MNIST 数据集包括手写数字的扫描和相关标签 (描述每个图像中包含 0~9 中哪个数字)。这个简单的分类问题是深度学习研究中最简单和最广泛使用的测试之一。尽管现代技术很容易解决这个问题，它仍然很受欢迎。Geoffrey Hinton 将其描述为“机器学习的果蝇”，这意味着机器学习研究人员可以在受控的实验室条件下研究他们的算法，就像生物学家经常研究果蝇一样。

1.2.3 与日俱增的模型规模

20 世纪 80 年代，神经网络只能取得相对较小的成功，而现在神经网络非常成功的另一个重要原因是我们现在拥有的计算资源可以运行更大的模型。联结主义的主要见解之一是，当动物的许多神经元一起工作时会变得聪明。单独神经元或小集合的神经元不是特别有用。

生物神经元不是特别稠密地连接在一起。如图 1.10 所示，几十年来，我们的机器学习模型中每个神经元的连接数量已经与哺乳动物的大脑在同一数量级上。

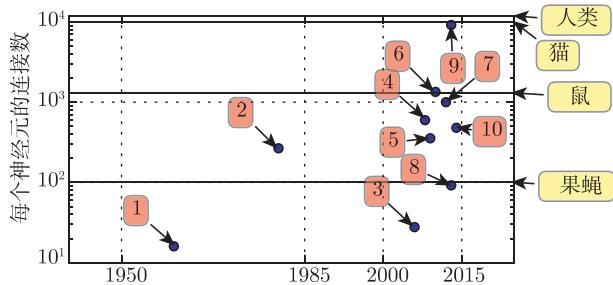


图 1.10 与日俱增的每个神经元的连接数。最初，人工神经网络中神经元之间的连接数受限于硬件能力。而现在，神经元之间的连接数大多是出于设计考虑。一些人工神经网络中每个神经元的连接数与猫一样多，并且对于其他神经网络来说，每个神经元的连接数与较小哺乳动物（如小鼠）一样多，这种情况是非常普遍的。甚至人类大脑每个神经元的连接数也没有过高的数量。生物神经网络规模来自 Wikipedia (2015)

1. 自适应线性单元 (Widrow and Hoff, 1960); 2. 神经认知机 (Fukushima, 1980); 3. GPU- 加速卷积网络 (Chellapilla et al., 2006); 4. 深度玻尔兹曼机 (Salakhutdinov and Hinton, 2009a); 5. 无监督卷积网络 (Jarrett et al., 2009b); 6. GPU- 加速多层感知机 (Ciresan et al., 2010); 7. 分布式自编码器 (Le et al., 2012); 8. Multi-GPU 卷积网络 (Krizhevsky et al., 2012a); 9. COTS HPC 无监督卷积网络 (Coates et al., 2013); 10. GoogLeNet (Szegedy et al., 2014a)

如图 1.11 所示，就神经元的总数而言，直到最近神经网络都是惊人的小。自从隐藏单元引入以来，人工神经网络的规模大约每 2.4 年扩大一倍。这种增长是由更大内存、更快的计算机和更大的可用数据集驱动的。更大的网络能够在更复杂的任务中实现更高的精度。这种趋势看起来将持续数十年。除非有能力迅速扩展新技术，否则至少要到 21 世纪 50 年代，人工神经网络才能具备与人脑相同数量级的神经元。生物神经元表示的功能可能比目前的人工神经元所表示的更复杂，因此生物神经网络可能比图中描绘的甚至要更大。

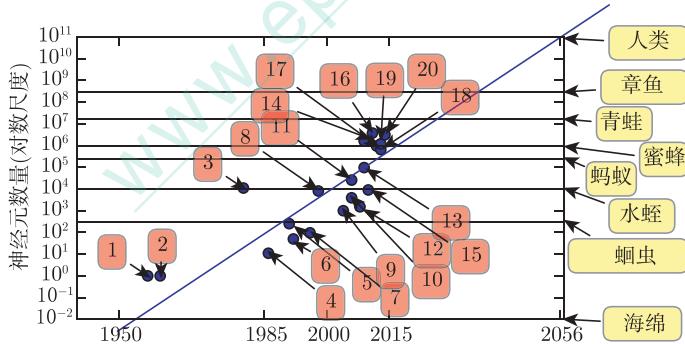


图 1.11 与日俱增的神经网络规模。自从引入隐藏单元，人工神经网络的规模大约每 2.4 年翻一倍。生物神经网络规模来自 Wikipedia (2015)

1. 感知机 (Rosenblatt, 1958, 1962); 2. 自适应线性单元 (Widrow and Hoff, 1960); 3. 神经认知机 (Fukushima, 1980); 4. 早期后向传播网络 (Rumelhart et al., 1986b); 5. 用于语音识别的循环神经网络 (Robinson and Fallside, 1991); 6. 用于语音识别的多层感知机 (Bengio et al., 1991); 7. 均匀场 sigmoid 信念网络 (Saul et al., 1996); 8. LeNet-5 (LeCun et al., 1998c); 9. 回声状态网络 (Jaeger and Haas, 2004); 10. 深度信念网络 (Hinton et al., 2006a); 11. GPU- 加速卷积网络 (Chellapilla et al., 2006); 12. 深度玻尔兹曼机 (Salakhutdinov and Hinton, 2009a); 13. GPU- 加速深度信念网络 (Raina et al., 2009a); 14. 无监督卷积网络 (Jarrett et al., 2009b); 15. GPU- 加速多层感知机 (Ciresan et al., 2010); 16. OMP-1 网络 (Coates and Ng, 2011); 17. 分布式自编码器 (Le et al., 2012); 18. Multi-GPU 卷积网络 (Krizhevsky et al., 2012a); 19. COTS HPC 无监督卷积网络 (Coates et al., 2013); 20. GoogLeNet (Szegedy et al., 2014a)

现在看来，神经元数量比一个水蛭还少的神经网络不能解决复杂的人工智能问题，这是不足为奇的。即使现在的网络，从计算系统角度来看它可能相当大，但实际上它比相对原始的脊椎动物（如青蛙）的神经系统还要小。

由于更快的 CPU、通用 GPU 的出现（在第 12.1.2 节中讨论）、更快的网络连接和更好的分布式计算的软件基础设施，模型规模随着时间的推移不断增加是深度学习历史中最重要的趋势之一。人们普遍预计这种趋势将很好地持续到未来。

1.2.4 与日俱增的精度、复杂度和对现实世界的冲击

20 世纪 80 年代以来，深度学习提供精确识别和预测的能力一直在提高。而且，深度学习持续成功地应用于越来越广泛的实际问题中。

最早的深度模型被用来识别裁剪紧凑且非常小的图像中的单个对象（Rumelhart *et al.*, 1986d）。此后，神经网络可以处理的图像尺寸逐渐增加。现代对象识别网络能处理丰富的高分辨率照片，并且不需要在被识别的对象附近进行裁剪（Krizhevsky *et al.*, 2012b）。类似地，最早的网络只能识别两种对象（或在某些情况下，单类对象的存在与否），而这些现代网络通常能够识别至少 1000 个不同类别的对象。对象识别中最大的比赛是每年举行的 ImageNet 大型视觉识别挑战（ILSVRC）。深度学习迅速崛起的激动人心的一幕是卷积网络第一次大幅赢得这一挑战，它将最高水准的前 5 错误率从 26.1% 降到 15.3%（Krizhevsky *et al.*, 2012b），这意味着该卷积网络针对每个图像的可能类别生成一个顺序列表，除了 15.3% 的测试样本，其他测试样本的正确类标都出现在此列表中的前 5 项里。此后，深度卷积网络连续地赢得这些比赛，截至写作本书时，深度学习的最新结果将这个比赛中的前 5 错误率降到了 3.6%，如图 1.12 所示。

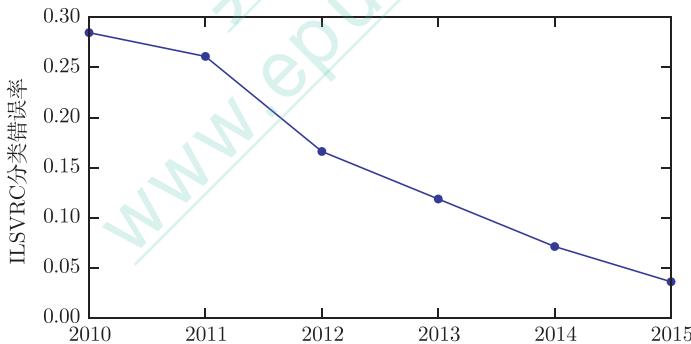


图 1.12 日益降低的错误率。由于深度网络达到了在 ImageNet 大规模视觉识别挑战中竞争所必需的规模，它们每年都能赢得胜利，并且产生越来越低的错误率。数据来源于 Russakovsky *et al.* (2014b) 和 He *et al.* (2015)

深度学习也对语音识别产生了巨大影响。语音识别在 20 世纪 90 年代得到提高后，直到约 2000 年都停滞不前。深度学习的引入（Dahl *et al.*, 2010; Deng *et al.*, 2010b; Seide *et al.*, 2011; Hinton *et al.*, 2012a）使得语音识别错误率陡然下降，有些错误率甚至降低了一半。我们将在第 12.3 节更详细地探讨这个历史。

深度网络在行人检测和图像分割中也取得了引人注目的成功（Sermanet *et al.*, 2013; Farabet *et al.*, 2013; Couprie *et al.*, 2013），并且在交通标志分类上取得了超越人类的表现（Ciresan *et al.*, 2012）。

在深度网络的规模和精度有所提高的同时，它们可以解决的任务也日益复杂。Goodfellow *et al.* (2014d) 表明，神经网络可以学习输出描述图像的整个字符序列，而不是仅仅识别单个对象。此前，人们普遍认为，这种学习需要对序列中的单个元素进行标注 (Gulcehre and Bengio, 2013)。循环神经网络，如之前提到的 LSTM 序列模型，现在用于对序列和其他序列之间的关系进行建模，而不是仅仅固定输入之间的关系。这种序列到序列的学习似乎引领着另一个应用的颠覆性发展，即机器翻译 (Sutskever *et al.*, 2014; Bahdanau *et al.*, 2015)。

这种复杂性日益增加的趋势已将其推向逻辑结论，即神经图灵机 (Graves *et al.*, 2014) 的引入，它能学习读取存储单元和向存储单元写入任意内容。这样的神经网络可以从期望行为的样本中学习简单的程序。例如，从杂乱和排好序的样本中学习对一系列数进行排序。这种自我编程技术正处于起步阶段，但原则上未来可以适用于几乎所有的任务。

深度学习的另一个最大的成就是其在强化学习(reinforcement learning)领域的扩展。在强化学习中，一个自主的智能体必须在没有人类操作者指导下，通过试错来学习执行任务。DeepMind 表明，基于深度学习的强化学习系统能够学会玩 Atari 视频游戏，并在多种任务中可与人类匹敌 (Mnih *et al.*, 2015)。深度学习也显著改善了机器人强化学习的性能 (Finn *et al.*, 2015)。

许多深度学习应用都是高利润的。现在深度学习被许多顶级的技术公司使用，包括 Google、Microsoft、Facebook、IBM、Baidu、Apple、Adobe、Netflix、NVIDIA 和 NEC 等。

深度学习的进步也严重依赖于软件基础架构的进展。软件库如 Theano (Bergstra *et al.*, 2010a; Bastien *et al.*, 2012a)、PyLearn2 (Goodfellow *et al.*, 2013e)、Torch (Collobert *et al.*, 2011b)、DistBelief (Dean *et al.*, 2012)、Caffe (Jia, 2013)、MXNet (Chen *et al.*, 2015) 和 TensorFlow (Abadi *et al.*, 2015) 都能支持重要的研究项目或商业产品。

深度学习也为其他科学做出了贡献。用于对象识别的现代卷积网络为神经科学家们提供了可以研究的视觉处理模型 (DiCarlo, 2013)。深度学习也为处理海量数据以及在科学领域做出有效的预测提供了非常有用的工具。它已成功地用于预测分子如何相互作用、从而帮助制药公司设计新的药物 (Dahl *et al.*, 2014)，搜索亚原子粒子 (Baldi *et al.*, 2014)，以及自动解析用于构建人脑三维图的显微镜图像 (Knowles-Barley *et al.*, 2014) 等多个场合。我们期待深度学习未来能够出现在越来越多的科学领域中。

总之，深度学习是机器学习的一种方法。在过去几十年的发展中，它大量借鉴了我们关于人脑、统计学和应用数学的知识。近年来，得益于更强大的计算机、更大的数据集和能够训练更深网络的技术，深度学习的普及性和实用性都有了极大的发展。未来几年，深度学习更是充满了进一步提高并应用到新领域的挑战和机遇。