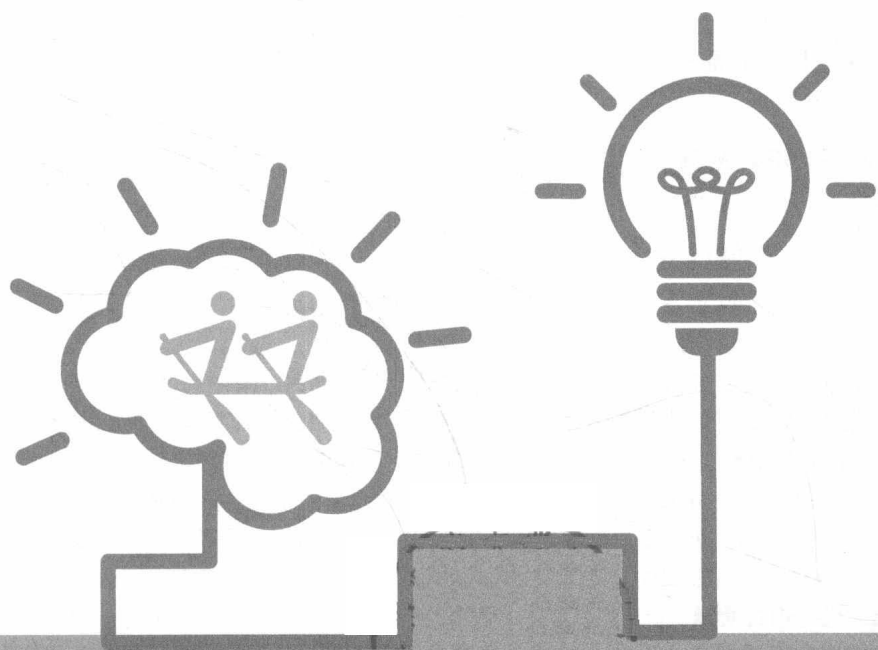


深度学习实战之 PaddlePaddle

潘志宏 王培彬 万智萍 邱泽敏◎编著



- ★ 国内计算机视觉知名教授推荐
- ★ 全书尽可能避开数学公式，以深度学习中的图像识别案例贯穿其中，内容由浅入深
- ★ 从简单的手写数字识别、CIFAR 彩色图像识别开始，到生活中广泛应用的验证码识别、车牌识别、场景文字识别，再到移动端、服务器端的深度学习应用。旨在提高读者的 PaddlePaddle 深度学习应用程序开发水平



深度学习实战之 PaddlePaddle

潘志宏 王培彬 编著
万智萍 邱泽敏

人民邮电出版社
北京

图书在版编目 (C I P) 数据

深度学习实战之PaddlePaddle / 潘志宏等编著. --
北京: 人民邮电出版社, 2019.6
ISBN 978-7-115-50332-9

I. ①深… II. ①潘… III. ①学习系统 IV.
①TP273

中国版本图书馆CIP数据核字(2018)第278866号

内 容 提 要

本书全面讲解了深度学习框架 PaddlePaddle, 并结合典型案例, 阐述了 PaddlePaddle 的具体应用。本书共 15 章。第 1 章介绍了深度学习及其主流框架; 第 2 章介绍了几种不同的 PaddlePaddle 安装方式; 第 3 章使用 MNIST 数据集实现手写数字识别; 第 4 章介绍 CIFAR 彩色图像识别; 第 5 章介绍了自定义数据集的识别; 第 6 章介绍了验证码的识别; 第 7 章介绍了场景文字的识别; 第 8 章实现了验证码的端到端的识别; 第 9~11 章讲解了车牌识别、使用 SSD 神经网络完成目标检测; 第 12 章和第 13 章介绍了 Fluid、可视化工具 VisualDL; 第 14 章和第 15 章介绍了如何在服务器端与 Android 移动终端使用 PaddlePaddle 进行项目实践。

本书适合机器学习爱好者、程序员、人工智能方面的从业人员阅读, 也可以作为大专院校相关专业的师生用书和培训学校的教材。

◆ 编 著 潘志宏 王培彬 万智萍 邱泽敏

责任编辑 张 涛

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>

固安县铭成印刷有限公司印刷

◆ 开本: 800×1000 1/16

印张: 17.75

字数: 325 千字

2019 年 6 月第 1 版

印数: 1-2 000 册

2019 年 6 月河北第 1 次印刷

定价: 69.00 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

推 荐 序

近年来，中国人工智能技术蓬勃发展，并且在某些领域上已经取得了非常出色的成绩。在语言识别技术中，科大讯飞语音识别已经应用于很多行业和场景；在图像识别技术中，腾讯优图、旷视科技 Face++ 和广州像素已经在医疗、安防和无人零售等行业广泛应用；在无人驾驶技术中，百度 Apollo 开放平台已经实现 L4 级别无人车的量产。虽然从技术研发层面看中国与美国相比还有一定差距，但是从应用层面看中国安防企业和互联网公司拥有超大规模的数据、广泛的应用场景，有些企业在人工智能方面已经实现了一系列成功的案例，同时，这也为我们的技术研究提供了丰富的数据。

就国家发展层面而言，中国政府已把人工智能列为国家发展战略，2017 年 7 月国务院印发了《新一代人工智能发展规划》，提出了 2030 年我国新一代人工智能发展的指导思想、战略目标、重点任务和保障措施。从人才培养方面，人才缺口迫使高校必须承担人才培养的工作，2018 年 4 月国家教育部印发了《高等学校人工智能创新行动计划》，提出中国高校要分“三步走”，2030 年要成为建设世界主要人工智能创新的核心力量和人才高地，国内很多知名高校也纷纷做出相应行动，从 2017 年到 2018 年短短 1 年多内，各大高校整合资源成立人工智能学院承担起人工智能人才培养和科研工作。

随着人工智能技术在各行各业落地开花，除了需要研究型人才之外，应用型人才也非常急缺。研究型人才主要由知名高校和研究院培养，而应用型人才或许更多的应该由应用型本科院校来承担。但是对于应用型本科院校的 IT 类专业学生，要快速学好人工智能（特别是深度学习知识）是不容易的，往往会因为大量数学公式而望而却步。本书作者是在应用型本科院校从教多年的教师，有丰富的教学经验，在内容的编排上尽可能避开了数学公式，并通过大量实际案例来教初学者如何解决实际的问题，从 MNIST 数据集中手写数字的识别、CIFAR 数据集中彩色图像的识别到日常生活实用性较强的验证码识别、场景文字识别、车牌识别，最后再到在移动端、服务器端实现深度学习应用。无论是对于深度学习的初学者还是已经具备一些项目经验但是想使用 PaddlePaddle 来进行项目开发的读者来说，本书都是很好的读物，我也衷心希望本书能够帮助更多读者掌

握深度学习方法，应用深度学习方法解决实际项目中的问题，也希望中国的人工智能事业蓬勃发展。

赖剑煌

中山大学数据科学与计算机学院教授
中国计算机学会计算机视觉专业组副主任
中国图象图形学会副理事长
广东省图象图形学会理事长

前 言

深度学习技术是目前非常热门的技术。随着大数据技术的发展和数据平台计算能力的提升，深度学习在行业中的应用越来越多。目前我国在人工智能技术的应用方面走在世界前列，应用领域从语音识别、图像识别到自然语言处理，再到自动驾驶等。在这些应用领域中，深度学习都发挥着重要的作用。同时，由于企业需要一批人工智能方面的应用型人才，因此很多知名高校纷纷成立人工智能学院培养人才来适应市场需求。对于普通信息类专业的学生，如果想快速进入深度学习领域，并利用深度学习解决实际问题，就需要了解大量的数学知识及公式。为了更好地进行学习，本书在讲解基础知识的同时，通过大量的实际案例展示如何解决实际的问题。

我在接触 PaddlePaddle 这个深度学习框架时，是通过官方教程开始学习的，但是随着学习的深入，发现 PaddlePaddle 的文档不是很完善，网络上的相关教程也非常少，于是我在 CSDN 网站上开设了“我的 PaddlePaddle 学习之路”系列教程。在编写这个系列教程的过程中，我遇到了不少的困难，好在 GitHub 社区的 PaddlePaddle 工程师们非常热心，帮助我解决了一个又一个的难题，最终完成了这个教程。这个教程发布之后，很多读者通过这个教程学习了如何使用 PaddlePaddle，并应用到自己的项目中。该教程还得到了读者的一致好评。有些读者联系到我，希望我能够出版一本关于 PaddlePaddle 的图书，让他们可以更系统地学习 PaddlePaddle。应读者的要求，我和我的导师潘志宏经过几个月的努力，终于完成了本书的编写工作。

本书一共有 15 章，由浅入深，从基础知识、环境安装到项目实战，一步步带领读者从零开始使用深度学习技术和 PaddlePaddle 解决实际问题。另外，我们在 GitHub 平台上提供了本书中所有项目的代码。相关代码都经过本书编写团队编程实现和运行验证，读者在实际开发中，只需要做少量修改就可以迁移到自己的项目中。

第 1 章介绍了深度学习和深度学习的一些主流框架，同时还简单介绍了深度学习中比较常用的数学知识。

第 2 章介绍了 PaddlePaddle 的安装。该章介绍了在 Windows 和 Ubuntu 两大主流操作系统上 PaddlePaddle 的安装方式，其中包括 Docker、原生 pip、源码编译的安装方式，以满足不同读者的需求。

从第 3 章开始介绍 PaddlePaddle 的使用方法。

第 3 章和第 4 章介绍了深度学习中比较常见的两个例子，即使用神经网络实现 MNIST 数据集中手写数字的识别和 CIFAR 数据集中彩色图像的识别。

为了让读者明白如何训练数据集，第 5 章介绍了自定义数据集的识别，并在第 6 章讨论了验证码的识别。

第 7 章介绍了比较常见的一种需求场景——文字识别，在这个技术基础之上，在第 8 章中实现了验证码端到端的识别，并且在第 9 章中实现了常见的另一种场景——车牌识别。

在第 10 章和第 11 章中，使用 SSD 神经网络在 VOC 数据集和自定义数据集上分别完成了目标检测。

为了让读者适应 PaddlePaddle 之后的版本，第 12 章介绍了 Fluid 版本的使用方法。

第 13 章介绍了 PaddlePaddle 推出的一个深度学习可视化工具 VisualDL，使用这个工具，可以更加方便地观察训练的情况。

第 14 章和第 15 章分别介绍了如何在服务器端与 Android 移动端使用 PaddlePaddle，通过这两章的学习，读者可以更好地把 PaddlePaddle 应用到自己的项目中。

本书由中山大学新华学院潘志宏担任主编，并负责全书内容的组织和编审，王培彬（网名为夜雨飘零）负责全书大部分章节的编写。中山大学新华学院万智萍、邱泽敏参与了部分章节的编写。本书的编写得到以下基金项目的支持：2017 年第二批教育部产学研合作协同育人项目（201702071078），2016 年广东省普通高校重大平台与重大科研项目——青年创新人才项目（自然科学）（2016KQNCX222），2016 年度广东省本科高校高等教育教学改革项目（2016SGJ002），2017 年度中山大学新华学院教学质量与教学改革工程项目（2017JC001）。

本书在编写过程中参考了百度 PaddlePaddle 官网和其他人分享的相关技术文献，在此对相关人士表示衷心感谢。另外，感谢 GitHub 社区的 PaddlePaddle 工程师帮助我解决了一些问题。限于篇幅，不能够在本书中介绍更多的 PaddlePaddle 知识，之后更多的教程会在博客上继续更新，欢迎读者继续关注本人的博客，在 CSDN 网站上搜索“夜雨飘零”即可查看更多教程。编辑联系邮箱为 zhangtao@ptpress.com.cn。

由于本人的水平和认知有限，书中难免会有不妥之处，恳请各位读者批评指正。

王培彬

服务与支持

本书由异步社区出品，社区 (<https://www.epubit.com/>) 为您提供相关资源和后续服务。

提交勘误

作者和编辑尽最大努力来确保书中内容的准确性，但难免会存在疏漏。欢迎您将发现的问题反馈给我们，帮助我们提升图书的质量。

当您发现错误时，请登录异步社区，按书名搜索，进入本书页面，单击“提交勘误”，输入勘误信息，单击“提交”按钮即可。本书的作者和编辑会对您提交的勘误进行审核，确认并接受后，您将获赠异步社区的 100 积分。积分可用于在异步社区兑换优惠券、样书或奖品。



The screenshot shows a web form for submitting勘误 (勘误). At the top, there are three tabs: '详细信息' (Detailed Information), '写书评' (Write a Review), and '提交勘误' (Submit勘误), with the latter being the active tab. Below the tabs, there are three input fields: '页码:' (Page Number), '页内位置 (行数):' (Page Position (Line Number)), and '勘误印次:' (勘误次数). Below these fields is a rich text editor with a toolbar containing icons for bold (B), italic (I), underline (U), bulleted list, numbered list, link, and unlink. At the bottom right of the form, there is a '字数统计' (Character Count) label and a '提交' (Submit) button.

扫码关注本书

扫描下方二维码，您将会在异步社区微信服务号中看到本书信息及相关的服务提示。



与我们联系

我们的联系邮箱是 contact@epubit.com.cn。

如果您对本书有任何疑问或建议，请您发邮件给我们，并请在邮件标题中注明本书书名，以便我们更高效地做出反馈。

如果您有兴趣出版图书、录制教学视频，或者参与图书翻译、技术审校等工作，可以发邮件给我们；有意出版图书的作者也可以到异步社区在线提交投稿（直接访问 www.epubit.com/selfpublish/submission 即可）。

如果您是学校、培训机构或企业，想批量购买本书或异步社区出版的其他图书，也可以发邮件给我们。

如果您在网上发现有针对异步社区出品图书的各种形式的盗版行为，包括对图书全部或部分内容的非授权传播，请您将怀疑有侵权行为的链接发邮件给我们。您的这一举动是对作者权益的保护，也是我们持续为您提供有价值的内容的动力之源。

关于异步社区和异步图书

“异步社区”是人民邮电出版社旗下 IT 专业图书社区，致力于出版精品 IT 技术图书和相关学习产品，为作译者提供优质出版服务。异步社区创办于 2015 年 8 月，提供大量精品 IT 技术图书和电子书，以及高品质技术文章和视频课程。更多详情请访问异步社区官网 <https://www.epubit.com>。

“异步图书”是由异步社区编辑团队策划出版的精品 IT 专业图书的品牌，依托于人民邮电出版社近 30 年的计算机图书出版积累和专业编辑团队，相关图书在封面上印有异步图书的 LOGO。异步图书的出版领域包括软件开发、大数据、AI、测试、前端、网络技术等。



异步社区



微信服务号

目 录

第 1 章 深度学习	1	2.9 测试安装效果	43
1.1 引言	1	2.10 小结	45
1.2 深度学习框架简介	1	第 3 章 使用 MNIST 数据集实现手写 数字识别	46
1.3 数学基础知识	3	3.1 引言	46
1.3.1 线性代数相关知识	3	3.2 数据集	46
1.3.2 概率论相关知识	10	3.3 定义神经网络模型	47
1.3.3 导数相关知识	13	3.4 开始训练模型	50
1.4 简单的深度学习理论知识	14	3.4.1 导入依赖包	50
1.5 小结	19	3.4.2 初始化 Paddle	51
第 2 章 PaddlePaddle 的安装	20	3.4.3 获取训练器	51
2.1 引言	20	3.4.4 开始训练	52
2.2 计算机配置	20	3.5 使用参数预测	54
2.3 安装前的检查	20	3.5.1 初始化 PaddlePaddle	54
2.4 使用 pip 安装	21	3.5.2 获取训练好的参数	54
2.5 使用 Docker 安装	23	3.5.3 读取图片	54
2.6 从源码编译并生成安装包	25	3.5.4 开始预测	55
2.6.1 在本地编译并生成安装包	25	3.6 小结	56
2.6.2 在 Docker 中编译并生成 安装包	28	第 4 章 CIFAR 数据集中彩色图像的 识别	57
2.7 编译 Docker 镜像	29	4.1 引言	57
2.8 在 Windows 操作系统中安装 PaddlePaddle 的方法	30	4.2 数据集	57
2.8.1 在 Windows 系统中安装 Docker 容器	30	4.3 定义神经网络模型	59
2.8.2 在 Windows 系统中 安装 Ubuntu	35	4.4 开始训练模型	61
2.8.3 在 Windows 10 中安装 Linux 子系统	41	4.4.1 导入依赖包	62
		4.4.2 初始化 Paddle	62
		4.4.3 获取参数	62
		4.4.4 创建训练器	63

4.4.5	开始训练	64	6.2.4	生成图像列表	110
4.5	使用参数预测	67	6.3	读取数据	111
4.6	使用其他神经模型	69	6.4	使用 PaddlePaddle 开始训练	112
4.7	小结	70	6.5	使用 PaddlePaddle 预测	118
第 5 章	自定义图像数据集的识别	72	6.5.1	裁剪验证码	118
5.1	引言	72	6.5.2	预测图像	119
5.2	网络爬虫技术	72	6.5.3	标签转成字符	120
5.2.1	网络爬虫的整体框架	72	6.6	小结	121
5.2.2	URL 管理器	74	第 7 章	场景文字识别	122
5.2.3	网页下载器	75	7.1	引言	122
5.2.4	网页解析器	76	7.2	数据集	122
5.3	网络爬虫实例	77	7.3	定义神经网络模型	123
5.3.1	调度器的使用	79	7.4	数据的读取	128
5.3.2	URL 管理器的使用	80	7.4.1	读取图像列表	128
5.3.3	网页下载器的使用	81	7.4.2	生成标签字典	129
5.3.4	网页解析器的使用	82	7.4.3	读取训练数据	131
5.3.5	数据收集器的使用	83	7.5	训练模型	133
5.3.6	运行代码	84	7.5.1	训练准备	133
5.4	数据集	88	7.5.2	安装 libwarpctc.so 库	135
5.4.1	生成图像列表	89	7.5.3	开始训练	136
5.4.2	读取数据	92	7.6	开始预测	137
5.5	定义神经网络	96	7.7	小结	140
5.6	使用 PaddlePaddle 开始训练	97	第 8 章	验证码端到端的识别	141
5.6.1	创建训练器	98	8.1	引言	141
5.6.2	开始训练	99	8.2	数据集	141
5.7	使用 PaddlePaddle 预测	102	8.3	生成图像列表文件	143
5.8	小结	104	8.4	数据的读取	144
第 6 章	验证码的识别	105	8.4.1	读取数据并存储成列表	144
6.1	引言	105	8.4.2	生成和读取标签字典	145
6.2	数据集的获取	105	8.4.3	读取训练和测试的数据	146
6.2.1	下载验证码	106	8.5	定义网络模型	147
6.2.2	修改验证码的文件名	107	8.6	生成训练器	150
6.2.3	裁剪验证码	108	8.7	定义训练	151
			8.8	启动训练	152

8.9	开始预测	153	第 11 章	通过自定义图像数据集实现	
8.10	小结	156		目标检测	196
第 9 章	车牌端到端的识别	157	11.1	引言	196
9.1	引言	157	11.2	数据集	196
9.2	车牌数据的采集	157	11.2.1	下载车牌数据	196
9.2.1	车牌数据的下载	157	11.2.2	重命名图像	197
9.2.2	命名车牌图像	159	11.3	标注数据集	198
9.2.3	车牌定位	159	11.3.1	安装 LabelImg	198
9.2.4	灰度化图像	163	11.3.2	使用 LabelImg	198
9.3	数据的读取	164	11.3.3	生成图像列表	201
9.3.1	生成列表文件	164	11.4	训练模型	202
9.3.2	以列表方式读取数据	165	11.4.1	预训练模型处理	202
9.3.3	生成和读取标签字典	166	11.4.2	开始训练	203
9.3.4	训练数据和测试数据的		11.5	评估模型	204
	读取	167	11.6	预测图片	205
9.4	定义神经网络	169	11.6.1	获取预测结果	205
9.5	开始训练	171	11.6.2	显示预测结果	206
9.6	开始预测	173	11.7	小结	208
9.7	小结	176	第 12 章	使用 PaddlePaddle Fluid	209
第 10 章	使用 VOC 数据集实现目标		12.1	引言	209
	检测	177	12.2	Fluid 版本	209
10.1	引言	177	12.3	定义神经网络	210
10.2	VOC 数据集	177	12.4	训练程序	212
10.2.1	下载 VOC 数据集	178	12.4.1	定义数据	213
10.2.2	生成图像列表	179	12.4.2	定义平均正确率	213
10.3	数据预处理	180	12.4.3	定义测试程序	213
10.4	SSD 神经网络	182	12.4.4	定义优化方法	214
10.5	训练模型	186	12.5	训练模型	214
10.6	评估模型	189	12.5.1	定义调试器	215
10.7	预测数据	191	12.5.2	获取数据	215
10.7.1	预测并保存预测		12.5.3	开始训练	216
	结果	191	12.5.4	保存预测模型	217
10.7.2	显示画出的框	193	12.6	预测模型	217
10.8	小结	195			

12.7	小结	219	14.3.2	测试 Flask 框架是否安装成功	234
第 13 章	可视化工具 VisualDL 的使用	220	14.3.3	文件上传	235
13.1	引言	220	14.4	使用 PaddlePaddle 预测	237
13.2	VisualDL 的介绍	220	14.4.1	获取预测模型	237
13.3	VisualDL 的安装	222	14.4.2	部署 PaddlePaddle	238
13.3.1	使用 pip 安装	223	14.5	小结	242
13.3.2	使用源码安装	224	第 15 章	把 PaddlePaddle 应用到 Android 手机	244
13.4	简单使用 VisualDL	224	15.1	引言	244
13.5	在 PaddlePaddle 中使用 VisualDL	226	15.2	编译 PaddlePaddle 库	244
13.5.1	定义 VisualDL 组件	226	15.2.1	使用 Docker 编译 PaddlePaddle 库	244
13.5.2	编写 PaddlePaddle 代码	227	15.2.2	使用 Linux 编译 PaddlePaddle 库	247
13.5.3	把数据添加到 VisualDL 中	229	15.3	MobileNet 神经网络	250
13.6	小结	232	15.4	训练模型	254
第 14 章	把 PaddlePaddle 部署到网站服务器上	233	15.5	编写预测代码	258
14.1	引言	233	15.6	合并模型	261
14.2	开发环境	233	15.7	移植到 Android	262
14.3	Flask 的使用	234	15.7.1	加载 PaddlePaddle 库	262
14.3.1	安装 Flask	234	15.7.2	加载合并的模型	263
			15.7.3	开发 Android 程序	263
			15.8	小结	272

第1章 深度学习

1.1 引言

2016年3月，AlphaGo与围棋世界冠军、职业九段棋手李世石进行围棋人机大战，最终AlphaGo以4:1的总比分获胜。这场比赛引起社会的高度关注，同时也把人工智能又一次推到顶峰，这可能意味着人工智能真正进入了我们的生活。次年，小度机器人在《最强大脑》节目中与人类选手共同获得了该节目的最高荣誉——“脑王”。这两件事情同时说明了一个问题，人工智能在很多领域中已经可以与人类媲美，部分领域甚至已经超过了人类。AlphaGo与小度机器人能具有如此强大的功能，共同之处都是它们使用了深度学习技术。

最近这几年，深度学习技术之所以发展这么快，一方面是因为深度算法的突破，另一方面是因为计算能力的提升。这些都是推动深度学习不断发展的动力。在这个恰当的时机下，如果你怀着一个人工智能的梦，不妨就从现在开始接触深度学习，让人工智能的发展道路上也能留下你的足迹。当你对此蠢蠢欲动但又不知道如何入手时，通常的建议是从学习一个深度学习框架开始。那么，什么是深度学习框架呢？继续往下看。

本章代码参见GitHub的yeyupiaoling主页里BookSource中的chapter1。测试环境是Python 2.7。

1.2 深度学习框架简介

为什么要使用深度学习框架呢？因为使用深度学习框架可以大大降低学习深度学习的门槛。例如，我们可以使用这些积木快速搭建各种各样的房子。使用积木搭建房子，不仅快速而且门槛低，使用起来也非常灵活，同时可以随意搭建我们想要的房子。深度

学习框架提供类似积木的高级 API，通过使用这些 API 搭建我们想实现的网络。同时，深度学习框架还可以使得我们在 CPU 和 GPU 的使用上无缝迁移。

正是因为如此，使用深度学习框架之后，我们可以利用更多的时间研究更好的模型，而不用在一个小小的语法错误上纠结。那么，有哪些深度学习框架？我们又应该选择哪些呢？现在开源的深度学习框架有很多，如 TensorFlow、PaddlePaddle、PyTorch、Caffe2 和 Keras 等，而本书为什么选择 PaddlePaddle 呢？因为 PaddlePaddle 是百度公司在 2016 年 9 月 27 日发布的开源框架，也是国内首个开源深度学习平台。图 1-1 即为 PaddlePaddle 官网。

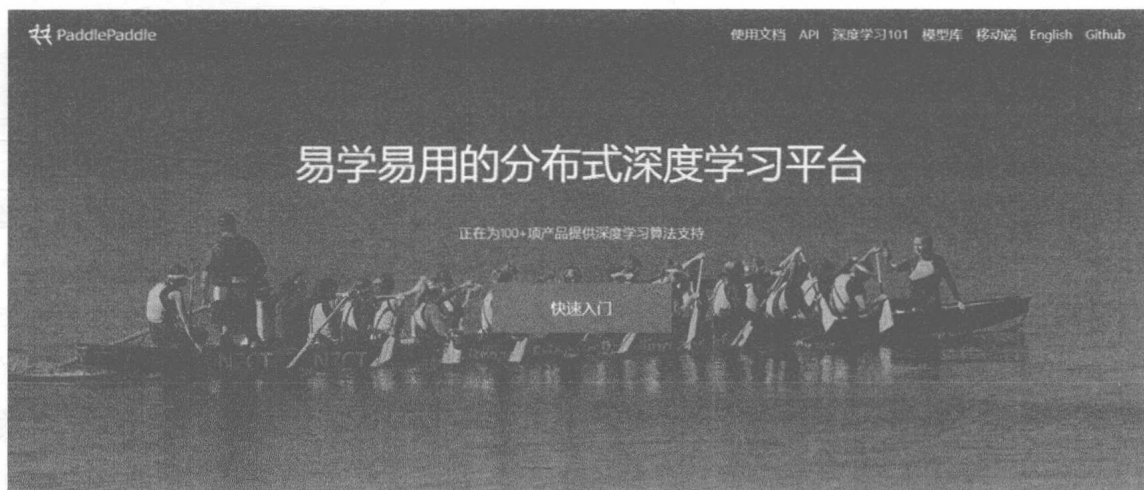


图 1-1 PaddlePaddle 官网

从官方标题中可以看出，这个框架是非常易用的，同时对我们入门深度学习也是非常友好的。如果能够快速入门，那么会增强我们继续学习的信心。同时，PaddlePaddle 官方也提出：要把 PaddlePaddle 发展成最符合中国开发者的深度学习框架之一。为什么要强调符合中国开发者呢？因为 PaddlePaddle 是由百度开源的一个企业级的深度学习平台，在国内的企业中，PaddlePaddle 首先惠及国内的开发者。在文档方面，PaddlePaddle 在中文文档上要比其他的深度学习框架更加完善，对于国内一些英文不太好的开发者来说，这也是非常方便的。另外，由 PaddlePaddle 官方创建的技术社区和录制的公开课都是针对国内开发者而设计的，因此在国内选择 PaddlePaddle 能够接触更多的官方资源，这是其他深度学习框架很难做到的。

PaddlePaddle 的前身是百度公司于 2013 年自主研发的深度学习平台，且一直为百度内部工程师研发使用。我们用的百度的图像识别、语音语义识别理解、情感分析、

机器翻译接口，其开发部门也会应用到这个平台。在 2016 年 9 月 1 日的百度世界大会上，百度首席科学家吴恩达首次宣布将百度深度学习平台对外开放，并命名为 PaddlePaddle。PaddlePaddle 在此之前已经在内部稳定运行了 3 年多时间，可以说是相当稳定了。同时百度官方也说明，PaddlePaddle 是完全开源的，不会存在内部版本和开源版本的情况。

使用 PaddlePaddle 深度学习框架到底有多么简单呢？下面以在 PaddlePaddle 中使用一个经典的卷积神经网络 VGG16 为例进行说明。在 PaddlePaddle 中，只需要下面的一行代码就可以完成 VGG16 的定义和调用，`input_image` 是指输入的数据类型，`num_channels` 是指输入图像的通道数，`num_classes` 是指输入数据集的分类总数。

```
paddle.v2.networks.vgg_16_network(input_image=img,
                                   num_channels=3,
                                   num_classes=10)
```

通过上面的例子，你是不是觉得使用 PaddlePaddle 框架非常简单并且已经忍不住想要马上尝试一下？在使用之前，希望读者能够简单了解一下深度学习的相关基础知识。接下来，本章就针对这两方面进行讲解。

1.3 数学基础知识

1.3.1 线性代数相关知识

1. 标量、向量、矩阵和张量

- **标量 (scalar)**: 标量就是一个单独的数，它不同于线性代数中大部分的研究对象（通常是多个数的数组）。我们用斜体表示标量。通常赋予标量小写的变量名称，如 x 。
- **向量 (vector)**: 向量是一列数。这些数都是有序排列的。通过次序中的索引，可以确定每个单独的数。通常赋予向量粗体的小写变量名称，如 \mathbf{x} 。

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

- **矩阵 (matrix)**: 矩阵是二维数组, 其中的每一个元素由两个索引 (而非一个) 确定。通常会赋予矩阵粗体的大写变量名称, 如 A 。

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$$

- **张量 (tensor)**: 在某种情况下, 我们会讨论坐标超过两维的数组。一般地, 若一个数组中的元素分布在若干维坐标的规则网格中, 则它称为张量, 并使用 A 表示。张量 A 中坐标为 (x,y,z) 的元素记作 A_{xyz} 。

下面使用 Python 代码实现上面所提到的矩阵。

1) 创建普通二维矩阵。

```
import numpy as np

m = np.mat([[1,2,3],[4,5,6]])
print m
```

输出为:

```
[[1 2 3]
 [4 5 6]]
```

2) 使用 `zeros` 创建一个 3×2 的 **0** 矩阵, 还可以使用 `ones` 函数创建元素全是 1 的矩阵。

```
from numpy import *
import numpy as np

m = np.mat(zeros((3,2)))
print m
```

输出为:

```
[[0. 0.]
 [0. 0.]
 [0. 0.]]
```

2. 转置

转置 (transpose) 是矩阵的重要操作之一。矩阵的转置是以对角线为轴的镜像, 这条从左上角到右下角的对角线称为主对角线。将矩阵 A 的转置表示为 A^T , 定义如下。