HZ BOOKS 华章IT
[PACKT] PUBLISHING

智能系统与技术丛书

Natural Language Processing with Python Cookbook

自然语言处理Python进阶

克里希纳·巴夫萨（Krishna Bhavsar）
[印度] 纳雷什·库马尔（Naresh Kumar） 著
普拉塔普·丹蒂（Pratap Dangeti）

陈钰枫 译

机械工业出版社
China Machine Press

智能系统与技术丛书

Natural Language Processing with Python Cookbook

自然语言处理Python进阶

克里希纳·巴夫萨（Krishna Bhavsar）
[印度] 纳雷什·库马尔（Naresh Kumar） 著
普拉塔普·丹蒂（Pratap Dangeti）
陈钰枫 译

机械工业出版社
China Machine Press

图书在版编目（CIP）数据

自然语言处理 Python 进阶 /（印）克里希纳·巴夫萨（Krishna Bhavsar）等著；陈钰枫译．
—北京：机械工业出版社，2019.1
（智能系统与技术丛书）
书名原文：Natural Language Processing with Python Cookbook

ISBN 978-7-111-61643-6

I. 自… II. ①克… ②陈… III. 软件工具－程序设计 IV. TP311.561

中国版本图书馆 CIP 数据核字（2019）第 003846 号

本书版权登记号：图字 01-2018-1366

自然语言处理 Python 进阶

出版发行：机械工业出版社（北京市西城区百万庄大街 22 号 邮政编码：100037）
责任编辑：赵 静　　责任校对：李秋荣
印　刷：三河市宏图印务有限公司　　版　次：2019 年 2 月第 1 版第 1 次印刷
开　本：186mm×240mm 1/16　　印　张：13.75
书　号：ISBN 978-7-111-61643-6　　定　价：59.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换
客服热线：（010）88379426 88361066　　投稿热线：（010）88379604
购书热线：（010）68326294 88379649 68995259　　读者信箱：hzit@hzbook.com

THE TRANSLATOR'S WORDS

译 者 序

当第一次阅读本书的时候，我深感它就是目前寻求自然语言处理和深度学习入门及进阶方法的读者所需要的。

眼下自然语言处理在人工智能世界大放异彩，渴望徜徉其中的读者众多。我在自然语言处理方面有研究基础，此前也翻译过一些该领域的前沿文章，希望这份译本能帮助想通过 Python 工具深入钻研自然语言处理的读者。

本书是利用 Python 语言解决自然语言处理多种任务的应用指南，旨在帮助读者快速浏览自然语言处理的全貌，进而掌握自然语言处理各项任务的基本原理，最终快速踏入实践阶段。Python 编程语言以其清晰简洁的语法、易用性和可扩展性以及丰富庞大的库深受广大开发者的喜爱。Python 内置了非常强大的机器学习代码库和数学库，使它理所当然地成为自然语言处理的开发利器。而其中的 NLTK 是使用 Python 处理自然语言数据的领先平台。它为各种语言资源（比如 WordNet）提供了简便易用的界面，它还能用于高效完成文本预处理、词性标注、信息抽取和文本分类等多种自然语言处理任务。

值得一提的是，本书特别涵盖了自然语言处理的高阶任务，比如主题识别、指代消解和创建聊天机器人等。本书最后两章介绍了深度学习在自然语言处理中的应用。每一节都大致包含三个部分：准备工作、如何实现和工作原理，对自然语言处理各种任务的方方面面都有所涉及。这意味着，如果读者能够踏实地完成本书中给出的所有实例，就能切实掌握自然语言处理各种任务的基础技术和原理方法，为进一步学习和研究奠定基础。换言之，深入阅读本书，对读者探索深度学习技术在自然语言处理中的应用会有极大帮助。

最后，感谢机械工业出版社华章公司的编辑，是他们的鼓励和支持使得本书中文版能够与读者见面。感谢武文雅和李兴亚等多位研究生的辅助和校对，感谢我的家人的支持。尽管我们努力准确地表达出作者介绍的思想和方法，但仍难免有不当之处。若发现译文中的错误，敬请指出，我们将非常感激，请将相关意见发往 chenyf@bjtu.edu.cn。

陈钰枫

2018 年 11 月

PREFACE

前　言

亲爱的读者，感谢你选择本书来开启你的自然语言处理（Natural Language Processing, NLP）之路。本书将从实用的角度带领你由浅入深逐步理解并实现 NLP 解决方案。我们将从访问内置数据源和创建自己的数据源开始指引你踏上这段旅程。之后你将可以编写复杂的 NLP 解决方案，包括文本规范化、预处理、词性标注、句法分析等。

在本书中，我们将介绍在自然语言处理中应用深度学习所必需的各种基本原理，它们是目前最先进的技术。我们将使用 Keras 软件来讨论深度学习的应用。

本书的出发点如下：

- 内容设计上旨在通过细节分析来帮助新手迅速掌握基本原理。并且，对有经验的专业人员来说，它将更新各种概念，以便更清晰地应用算法来选择数据。
- 介绍了在 NLP 中深度学习应用的新趋势。

本书的组织结构

第 1 章教你使用内置的 NLTK 语料库和频率分布。我们还将学习什么是 WordNet，并探索其特点和用法。

第 2 章演示如何从各种格式的数据源中提取文本。我们还将学习如何从网络源提取原始文本。最后，我们将从这些异构数据源中对原始文本进行规范并构建语料库。

第 3 章介绍一些关键的预处理步骤，如分词、词干提取、词形还原和编辑距离。

第 4 章介绍正则表达式，它是最基本、最简单、最重要和最强大的工具之一。在本章中，你将学习模式匹配的概念，它是文本分析的一种方式，基于此概念，没有比正则表达式更方便的工具了。

第 5 章将学习如何使用和编写自己的词性标注器和文法规则。词性标注是进一步句法分析的基础，而通过使用词性标记和组块标记可以产生或改进文法规则。

第 6 章帮助你了解如何使用内置分块器以及训练或编写自己的分块器，即依存句法分析器。在本章中，你将学习评估自己训练的模型。

第 7 章介绍信息抽取和文本分类，告诉你关于命名实体识别的更多信息。我们将使用内置的命名实体识别工具，并使用字典创建自己的命名实体。我们将学会使用内置的文本

分类算法和一些简单的应用实例。

第 8 章介绍高阶自然语言处理方法，该方法将目前为止你所学的所有课程结合到一起，并创建应对你现实生活中各种问题的适用方法。我们将介绍诸如文本相似度、摘要、情感分析、回指消解等任务。

第 9 章介绍深度学习应用于自然语言处理所必需的各种基本原理，例如利用卷积神经网络（CNN）和长短型记忆网络（LSTM）进行邮件分类、情感分类等，最后在低维空间中可视化高维词汇。

第 10 章描述如何利用深度学习解决最前沿的问题，包括文本自动生成、情景数据问答，预测下一个最优词的语言模型以及生成式聊天机器人的开发。

本书需要你做什么

为了成功完成本书的实例，你需要在 Windows 或 Unix 操作系统上安装 Python 3.x 及以上版本，硬件要求：CPU 2.0GHz 以上，内存 4GB 以上。就 Python 开发的 IDE 而言，市场上有许多可用的 IDE，但我最喜欢的是 PyCharm 社区版。它是一款由 JetBrains 开发的免费开源工具，它的技术支持很强大，会定期发布该工具的升级和修正版本，你只要熟悉 IntelliJ 就能保持学习进度顺畅。

本书假设你已经了解 Keras 的基本知识和如何安装库。我们并不要求读者已经具备深度学习的知识和数学知识，比如线性代数等。

在本书中，我们使用了以下版本的软件，它们在最新的版本下都能很好地运行：

- Anaconda 3　4.3.1（Anaconda 中包括所有 Python 及相关包，Python　3.6.1, NumPy 1.12.1, pandas　0.19.2）
- Theano　0.9.0
- Keras　2.0.2
- feedparser　5.2.1
- bs4　4.6.0
- gensim　3.0.1

本书的读者对象

本书适用于想利用 NLP 提升现有技能来实现高阶文本分析的数据科学家、数据分析师和数据科学专业人员，建议读者具备自然语言处理的一些基本知识。

本书也适用于对自然语言处理知识毫无了解的新手，或是希望将自己的知识从传统的 NLP 技术扩展到最先进的深度学习应用技术的有经验的专业人士。

小节

在本书中，有几个标题经常出现（准备工作、如何实现、工作原理、更多、参见）。为了明确说明如何完成一个实例，本书使用了如下内容排布：

准备工作

本节介绍完成实例的预期结果，并说明如何安装所需的软件或初步设置。

如何实现

本节包含完成实例所需遵循的步骤。

工作原理

本节通常对上一节的操作进行详细的解释。

更多

本节包含实例的补充信息，以便读者对实例的实现方法有更多的了解。

参见

本节为实例的其他有用信息提供有效的链接。

约定

在本书中，你会发现许多不同类型信息的文本格式。下面是这些格式的一些范例和它们的含义解释。

任何命令行的输入或输出格式如下：

```
# Deep Learning modules
>>> import numpy as np
>>> from keras.models import Sequential
```

新术语和重要词以加粗字体显示。

警告或重要提示将跟在这样的符号后面。

提示或小技巧将跟在这样的符号后面。

读者反馈

欢迎读者的反馈。让我们知道你对本书的看法，哪些部分喜欢还是不喜欢。读者反馈对我们来说很重要，因为它有助于我们了解读者真正能从中获益最多的地方。将你的反馈发送到邮箱 feedback@packtpub.com 即可，并且在你的邮箱标题中提到本书的书名。如果你有一个擅长或是比较感兴趣的话题，可以撰写或投稿书刊，请在 www.packtpub.com/authors 网站上查看作者指南。

客户支持

既然你是一本 Packt 书的拥有者，在购买时能够获得很多额外的资源。

示例代码下载

你可以从 http://www.packtpub.com 通过个人账号下载本书的示例代码文件。如果你在别处购买了本书，可以访问 http:// www. packtpub.com/support 并进行注册，就可以直接通过邮件方式获得相关文件。你可以按照以下步骤下载代码文件：

1. 使用你的电子邮件地址和密码登录或注册到我们的网站；
2. 将鼠标指针置于顶部的 SUPPORT 选项卡上；
3. 点击 Code Downloads & Errata；
4. 在 Search 框中输入图书的名称；
5. 选择你要下载的代码文件的相应图书；
6. 从你购买本书的下拉菜单中选择；
7. 点击 Code Download。

你也可以在 Packt 出版社网站关于本书的网页上通过点击 Code Files 按钮来下载代码文件。通过在 Search 框中输入图书的名称来访问该页面。请注意，你需要登录到你的 Packt 账户。一旦文件被下载，请确保你使用的是最新版本的工具解压文件夹：

- WinRAR/7-Zip for Windows
- Zipeg/iZip/UnRarX for Mac
- 7-Zip/PeaZip for Linux

本书的代码可从 GitHub 下载，托管在 https://github.com/PacktPublishing/Natural-Language-Processing-with-Python-Cookbook 上。其余代码包可以从 https://github.com/PacktPublishing/ 上丰富的图书和视频目录中获取。请检验测试！

ABOUT THE AUTHORS

作者简介

克里希纳·巴夫萨（Krishna Bhavsar）花了大约 10 年时间在各行业领域如酒店业、银行业、医疗行业等进行自然语言处理、社交媒体分析和文本挖掘方面的研究。他致力于用不同的 NLP 语料库如 Stanford CoreNLP、IBM 的 SystemText 和 BigInsights、GATE 和 NLTK 来解决与文本分析有关的行业问题。克里希纳还致力于分析社交媒体给热门电视节目和流行零售品牌以及产品带来的效应。2010 年，他在 NAACL 上发表了一篇关于情感分析增强技术的论文。近期，他创建了一个 NLP 管道 / 工具集并开源以便公众使用。除了学术和科技，克里希纳还热衷于摩托车和足球，空闲时间喜欢旅行和探索。他骑摩托车参加过环印度公路旅行并在东南亚和欧洲大部分国家徒步旅行过。

首先，我要感谢我的母亲，她是我生命中最大的动力和坚强的支柱。我要感谢 Synerzip 的管理团队和我所有的朋友在我写作过程中对我的支持。最后，特别感谢 Ram 和 Dorothy 在这困难的一年让我保持前进。

纳雷什·库马尔（Naresh Kumar）曾为财富 500 强企业设计、实施和运行超大型因特网应用程序，在这方面他拥有超过十年的专业经验。他是一位全栈架构师，在电子商务、网络托管、医疗、大数据及分析、数据流、广告和数据库等领域拥有丰富的实践经验。他依赖开源并积极为其做贡献。纳雷什一直走在新兴技术的前沿，从 Linux 系统内部技术到前端技术。他曾在拉贾斯坦邦的 BITS-Pilani 学习，获得了计算机科学和经济学的双学位。

普拉塔普·丹蒂（Pratap Dangeti）在班加罗尔的研究和创新实验室开发机器学习和深度学习方法，以用于结构化、图像和 TCS 文本数据。他在分析和数据科学领域拥有丰富的经验，并在 IIT Bombay 获得了工业工程和运筹学项目的硕士学位。普拉塔普是一名人工智能爱好者。闲暇时，他喜欢阅读下一代技术和创新方法。他还是 Packt 出版的《 Statistics for Machine Learning 》一书的作者。

我要感谢我的母亲 Lakshmi，感谢她在我的职业生涯以及撰写本书的过程中给予我的支持。我要把这本书献给她。我还要感谢我的家人和朋友，没有他们的鼓励，我不可能完成这本书。

ABOUT THE REVIEWER

审校者简介

Juan Tomas Oliva Ramos 是墨西哥瓜纳华托大学的环境工程师，他拥有行政工程和质量硕士学位。他在专利管理和开发、技术创新项目以及通过流程统计控制开发技术解决方案方面拥有超过 5 年的经验。

自 2011 年以来，Juan Tomas Oliva Ramos 一直是统计、创业和项目技术开发方面的教师。他还是一名企业家导师，并在墨西哥瓜纳华托德尔林孔理工高等研究院开设了一个新的技术管理和创业部门。

Juan 是 Alfaomega 的审稿人，致力于完成《 Wearable Designs for Smart Watches, Smart TVs and Android Mobile Devices 》一书。

Juan 还通过编程和自动化技术开发了原型系统，用于改进操作，并已经注册了专利。

感谢 Packt 出版社让我有机会审校这本精彩的书，并与一群志同道合的伙伴合作。

感谢我美丽的妻子 Brenda，我们的两位魔法公主（Maria Regina 和 Maria Renata）和我们的下一位成员（Angel Tadeo），你们每天都给了我力量、幸福和欢乐。谢谢你们。

CONTENTS

目　录

CHAPTER 1

第 1 章

语料库和 WordNet

1.1 引言

解决任何实际的**自然语言处理（NLP）**问题，都需要处理大量的数据。这些数据通常以公开语料库的形式存在，并可以由 NLTK 数据包的附加组件提供。例如，如果要创建一个拼写检查器，需要用一个大型单词语料库进行匹配。

本章将涵盖以下内容：

- 介绍 NLTK 提供的各种有用的文本语料库
- 如何用 Python 访问内置语料库
- 计算频率分布
- WordNet 及其词法特征介绍

我们将通过实践的方式来理解这些内容。下面我们会进行一些练习，通过实例来完成这些学习目标。

1.2 访问内置语料库

如前所述，NLTK 有许多可供使用的语料库。这里假设你已经在计算机上完成了 NLTK 数据库的下载和安装。如果没有，你可以通过网址 http://www.nltk.org/data.html 下载。此外，NLTK 数据库内的完整语料库列表可以通过网址 http://www.nltk.org/nltk_data/ 获取。

现在，我们的第一个任务 / 实例是学习如何访问这些语料库。我们在路透社语料库（Reuters corpus）上做一些实验。将语料库导入我们的程序中，并尝试用不同的方式进行访问。

如何实现

1. 创建一个新文件，将其命名为 reuters.py，并在该文件中添加以下代码。这是在整个

NLTK 数据集中仅访问路透社语料库的特定方式：

```
from nltk.corpus import reuters
```

2. 当我们想知道这个语料库中有什么内容时，最简单的方法是调用语料库对象中的 fileids() 函数。在程序中添加以下代码：

```
files = reuters.fileids()
print(files)
```

3. 运行该程序，将得到如下输出：

```
['test/14826', 'test/14828', 'test/14829', 'test/14832',
'test/14833', 'test/14839',
```

这些是路透社语料库中的文件列表和它们的相对路径。

4. 访问这些文件的具体内容。使用语料库对象的 words() 函数来访问 test/16097 文件：

```
words16097 = reuters.words(['test/16097'])
print(words16097)
```

5. 再次运行该程序，会出现一行新的输出内容：

```
['UGANDA', 'PULLS', 'OUT', 'OF', 'COFFEE', 'MARKET', ...]
```

输出了 test/16097 文件中的单词列表。虽然整个单词列表被加载到内存对象中，此处仅输出部分结果。

6. 从 test/16097 文件中获取特定数量的单词（例如 20 个）。当然，我们可以指定想要获取的单词数，并将其存储在列表中以供使用。添加如下两行代码：

```
words20 = reuters.words(['test/16097'])[:20]
print(words20)
```

运行代码，输出结果如下：

```
['UGANDA', 'PULLS', 'OUT', 'OF', 'COFFEE', 'MARKET', '-', 'TRADE',
'SOURCES', 'Uganda', "'", 's', 'Coffee', 'Marketing', 'Board', '(',
'CMB', ')', 'has', 'stopped']
```

7. 进一步，路透社语料库不仅仅是一个文件列表，而且还被按层次分成 90 个主题。每个主题都有许多与之关联的文件。也就是说，当你访问任何一个主题时，实际上访问的是与该主题相关的所有文件的集合。添加如下代码以输出主题列表：

```
reutersGenres = reuters.categories()
print(reutersGenres)
```

运行代码，输出控制台将有如下输出：

```
['acq', 'alum', 'barley', 'bop', 'carcass', 'castor-oil', 'cocoa',
'coconut', 'coconut-oil', ...
```

显示了所有的 90 个类别。

8. 最后，编写四行简单的代码，这不仅可以访问两个主题，还可以将单词以一行一个句子这样松散的方式打印出来。将以下代码添加到 Python 文件中：

```
for w in reuters.words(categories=['bop','cocoa']):
  print(w+' ',end='')
  if(w is '.'):
    print()
```

9. 简单解释一下，我们首先选择了类别 bop 和 cocoa，并打印这两个类别文件中的每个单词。每遇到一个点号（.），就插入一个新的行。运行代码，控制台将输出以下内容：

```
['test/14826', 'test/14828', 'test/14829', 'test/14832',
'test/14833', 'test/14839', ...
['UGANDA', 'PULLS', 'OUT', 'OF', 'COFFEE', 'MARKET', ...]
['UGANDA', 'PULLS', 'OUT', 'OF', 'COFFEE', 'MARKET', '-', 'TRADE',
'SOURCES', 'Uganda', "'", 's', 'Coffee', 'Marketing', 'Board', '(',
'CMB', ')', 'has', 'stopped']
['acq', 'alum', 'barley', 'bop', 'carcass', 'castor-oil', 'cocoa',
'coconut', 'coconut-oil', ...
SOUTH KOREA MOVES TO SLOW GROWTH OF TRADE SURPLUS South Korea ' s
trade surplus is growing too fast and the government has started
taking steps to slow it down , Deputy Prime Minister Kim Mahn-je
said .
He said at a press conference that the government planned to
increase investment , speed up the opening of the local market to
foreign imports, and gradually adjust its currency to hold the
surplus " at a proper level ." But he said the government would not
allow the won to appreciate too much in a short period of time .
South Korea has been under pressure from Washington to revalue the
won .
The U .
S .
Wants South Korea to cut its trade surplus with the U .
S ., Which rose to 7 .
4 billion dlrs in 1986 from 4 .
3 billion dlrs in 1985 .
.
.
.
```

1.3　下载外部语料库，加载并访问

现在我们已经学会了如何加载和访问内置语料库，下面将学习如何下载并加载，以及访问外部语料库。许多内置语料库都非常适用于训练，但是为了解决实际问题，通常需要一个外部数据集。在本节实例中，我们将使用 Cornell CS 电影评论语料库，该语料库对评论做了正面和负面的标记，已被广泛应用于训练情感分析模块。

1.3.1　准备工作

首先，你需要从互联网上下载数据集并将其解压缩。链接如下：http://www.cs.cornell.edu/people/pabo/movie-review-data/mix20_rand700_tokens_cleaned.zip。然后，将生成的评

论（Reviews）目录存储在计算机的安全位置。

1.3.2　如何实现

1. 创建一个名为 external_corpus.py 的新文件，并向该文件添加如下内容：

```
from nltk.corpus import CategorizedPlaintextCorpusReader
```

由于下载的语料库已经分类，我们将使用 CategorizedPlaintextCorpusReader 来读取和加载所给的语料库。用这种方式来获取正面评论和负面评论。

2. 读取语料库。我们需要知道从 Cornell 上下载的文件解压缩后的 Reviews 文件夹的绝对路径，添加以下四行代码：

```
reader = CategorizedPlaintextCorpusReader(r'/Volumes/Data/NLP-
CookBook/Reviews/txt_sentoken', r'.*\.txt', cat_pattern=r'(\w+)/*')
print(reader.categories())
print(reader.fileids())
```

第一行是通过调用 CategorizedPlaintextCorpusReader 构造函数来读取语料库。从左到右的三个参数分别是计算机上 txt_sentoken 文件夹的绝对路径，txt_sentoken 文件夹中的所有示例文档名以及给定语料库中的类别（在本例中为 pos 和 neg）。仔细观察，你会发现这三个参数都是正则表达式。接下来的两行将验证是否正确加载语料库，并打印出语料库的相关类别和文件名。运行该程序，你会看到以下内容：

```
['austen-emma.txt', 'austen-persuasion.txt', 'austen-sense.txt',
'bible-kjv.txt',....]
[['The', 'Fulton', 'County', 'Grand', 'Jury', 'said', 'Friday',
'an', 'investigation', 'of',...]]
```

3. 现在已经确保了该语料库被正确加载，下面继续访问这两个类别中的任何一个示例文档。为此，我们首先创建一个列表，每个列表分别包含 pos 和 neg 两个类别的样本。添加以下两行代码：

```
posFiles = reader.fileids(categories='pos')
negFiles = reader.fileids(categories='neg')
```

reader.fileids() 方法的参数为类别名称。你可以发现，以上两行代码的目的是直接明了的。

4. 现在我们从 posFiles 和 negFiles 的列表中随机选择一个文件。为此，我们需要利用 Python random 库中的 randint() 函数。我们添加如下几行代码，接下来会详细说明它们的具体功能：

```
from random import randint
fileP = posFiles[randint(0,len(posFiles)-1)]
fileN = negFiles[randint(0, len(posFiles) - 1)]
print(fileP)
print(fileN)
```

第一行从 random 库中导入 randint() 函数。接下来分别从正面和负面类别评论集中随机选择一个文件。最后两行只是打印文件名。