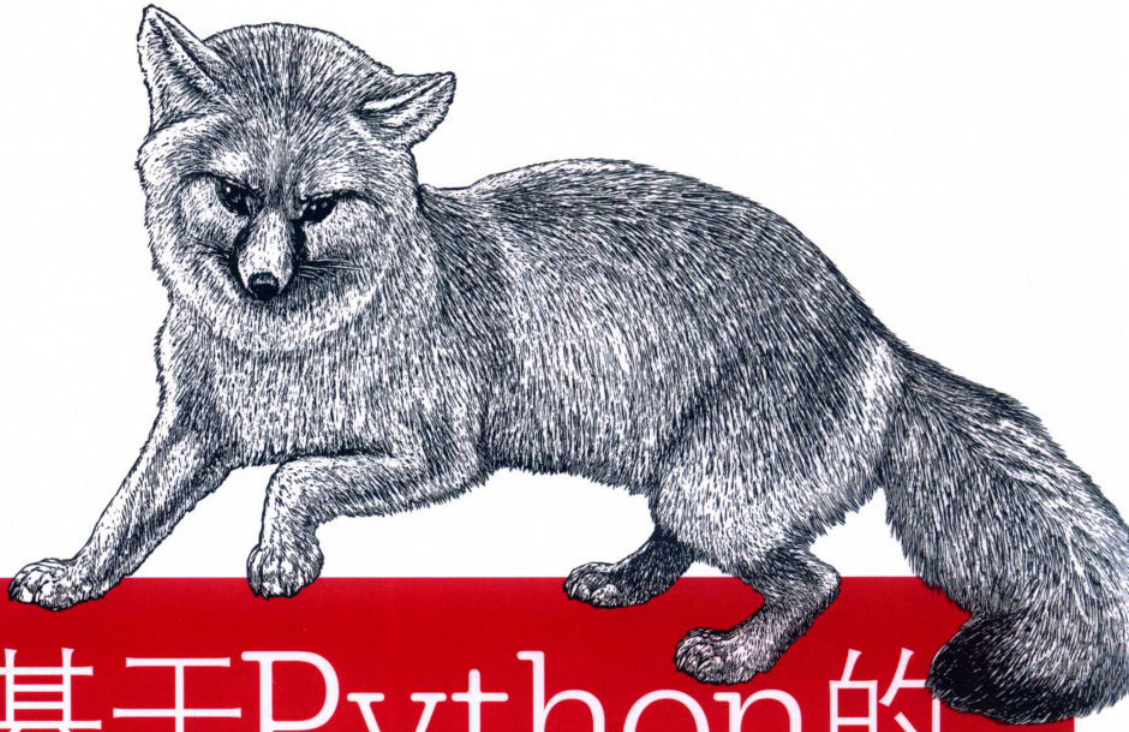
O'REILLY®



基于Python的智能文本分析

Applied Text Analysis with Python

Benjamin Bengfort
Rebecca Bilbro Tony Ojeda 著
陈光 译

中国电力出版社

基于Python的智能文本分析

Benjamin Bengfort, Rebecca Bilbro,
Tony Ojeda 著

陈光 译

opol • Tokyo O'REILLY®

edia, Inc. 授权中国电力出版社出版

中国电力出版社

图书在版编目（CIP）数据

基于Python的智能文本分析 / (美) /本杰明·班福特（Benjamin Bengfort），(美)瑞贝卡·比尔布罗（Rebecca Bilbro），(美) 托尼·奥杰达（Tony Ojeda）著；陈光译. — 北京：中国电力出版社，2020.1

书名原文：Applied Text Analysis with Python

ISBN 978-7-5198-3829-4

I. ①基… II. ①本… ②瑞… ③托… ④陈… III. ①软件工具－程序设计 IV. ①TP311.561

中国版本图书馆CIP数据核字(2019)第239871号

北京市版权局著作权合同登记 图字：01-2019-4196号

出版发行：中国电力出版社

地　址：北京市东城区北京站西街19号（邮政编码100005）

网　址：http://www.cepp.sgcc.com.cn

责任编辑：刘 炽（liuchi1030@163.com）

责任校对：黄蓓　常燕昆

装帧设计：Karen Montgomery，张健

责任印制：杨晓东

印　刷：北京天宇星印刷厂

版　次：2020年1月第一版

印　次：2020年1月北京第一次印刷

开　本：750毫米×980毫米　16开本

印　张：20.5

字　数：392千字

印　数：0001—3000册

定　价：88.00元

O'Reilly Media, Inc.介绍

O'Reilly以“分享创新知识、改变世界”为己任。40多年来我们一直向企业、个人提供成功所必需之技能及思想，激励他们创新并做得更好。

O'Reilly业务的核心是独特的专家及创新者网络，众多专家及创新者通过我们分享知识。我们的在线学习（Online Learning）平台提供独家的直播培训、图书及视频，使客户更容易获取业务成功所需的专业知识。几十年来O'Reilly图书一直被视为学习开创未来之技术的权威资料。我们每年举办的诸多会议是活跃的技术聚会场所，来自各领域的专业人士在此建立联系，讨论最佳实践并发现可能影响技术行业未来的新趋势。

我们的客户渴望做出推动世界前进的创新之举，我们希望能助他们一臂之力。

业界评论

“O'Reilly Radar博客有口皆碑。”

——Wired

“O'Reilly凭借一系列非凡想法（真希望当初我也想到了）建立了数百万美元的业务。”

——Business 2.0

“O'Reilly Conference是聚集关键思想领袖的绝对典范。”

——CRN

“一本O'Reilly的书就代表一个有用、有前途、需要学习的主题。”

——Irish Times

“Tim是位特立独行的商人，他不光放眼于最长远、最广阔的领域，并且切实地按照Yogi Berra的建议去做了：‘如果你在路上遇到岔路口，那就走小路。’回顾过去，Tim似乎每一次都选择了小路，而且有几次都是一闪即逝的机会，尽管大路也不错。”

——Linux Journal

目录

前言

我们生活在一个充满各种各样数字助理的世界，这使我们能与其他人以及大量的信息资源建立联系。这些智能设备的部分吸引力，在于它们不仅能传达信息；在一定程度上，它们也能理解信息，将大量数据聚合、过滤和汇总，处理成易于理解的形式，在高层次上促进人与人的交互。机器翻译、问答系统、语音识别、文本摘要，以及聊天机器人等应用，正成为我们计算生活中不可或缺的一部分。

如果已经浏览过本书，那么你可能会和我们一样，对将自然语言理解组件包含在更广泛的应用和软件中的可能性感到兴奋。语言理解组件建立在现代文本分析框架之上：结合了字符串操作、词汇资源、计算语言学和机器学习算法等技术、方法的工具包，用来在语言数据和机器可理解形式之间进行相互转换。不过，在我们开始讨论这些方法和技术之前，明确这套框架的挑战和机遇，以及为什么现在讨论这些正当其时这两个问题非常重要。

典型的美国高中毕业生已经记住了大约 60000 个单词和数千个语法概念，足以在专业环境中进行交流。这看起来可能很多，但是想想看，写个简短的 Python 脚本，可以快速访问在线词典里任何术语的定义、词源和用法，又是多么微不足道。实际上，美国人日常实践中使用的各种语言概念，仅仅是牛津词典的十分之一，也只有目前谷歌识别范围的 5%。

然而，能随时访问规则和定义，显然还不足以进行文本分析。如果是这样，Siri 和 Alexa 就已经能完全理解我们，谷歌也可以只返回少量搜索结果，我们就可以马上和世界上讲任何语言的人聊天。为什么人类可以从很小的时候（在他们远未积累到成年人的词汇量之前）就能流畅地完成上述任务？而已有的计算版本却远不能与之匹敌？自然语言不仅仅需要死记硬背，显然，仅是确定性计算技术是不够的。

自然语言的计算挑战

自然语言不是用规则定义的，而是通过使用来定义的，只有通过逆向工程才能计算。在很大程度上，我们能决定使用的词具体什么含义，尽管这种“造义”只能通过协同才能完成。将“crab”这个词从海洋动物延伸到脾气乖戾的人，或采用特定侧方运动形式的人，需要发言者 / 作者和听众 / 读者在交流发生时对词的含义达成一致。因此，语言通常会受到人群和地域限制，那些和我们有过类似生活经历的人，往往更容易和我们达成对含义的一致理解。

与正式语言必须面向特定领域不同，自然语言强调一般性和通用性。我们用同一个词来订购海鲜午餐，写一首诗表达不满情绪，讨论天文并聊聊星云。为了捕捉各种话语中表达的程度，语言必须是冗余的。冗余带来了挑战，我们不能（也没有）为每种关联指定特别的文字符号，在默认情况下每个符号都不明确。词汇歧义和结构歧义是人类语言的重要成果；模糊不仅能够使我们创造新的想法，还可以让具有不同经历的人跨越国界和文化进行交流，尽管偶然的误会几乎无法避免。

语言数据：词条和词

为了充分挖掘以语言形式编码的数据，我们有必要重新思考，认为语言并非直观和自然的，而是随意和模糊的。文本分析的单位是词条（token），一个表示文本的编码字节串。相对应的，词（words）是表示意义的符号，能将文字结构或语言结构映射到声音元素和视觉元素。词条不是词（虽然做到“只见词条不见词”很难）。考虑词条“`crab`”，如图 P-1 所示。该词条表示词

义 *crab-n1*，即第一个名词定义，可食用的甲壳类动物，生活在海边，长有可以夹合的钳子。

图 P-1：词将符号映射到概念

所有这些其他概念会以某种形式附加在这个符号上，但符号完全是随意的；对希腊的读者来说，类似的映射表达了略有差别的内涵，但意义基本保持不变。这是因为词没有固定的、普遍的、不受文化和语言等语境影响的意义。英语的读者习惯于适应性单词形式，通过增加前缀、后缀来改变时态、性别等。另外，中国的读者能识别很多由顺序决定意义的象形文字。

冗余、含糊、带有特定角度，意味着自然语言是动态的，通过快速演进来适应当下的人类体验。现在，我们听说关于表情符号的语言学研究已经可以翻译“白鲸”可能也不觉得惊讶。[注1] 就算我们能系统定义表情符号使用的语法，等我们完成，语言本身也已经变了——哪怕是表情符号语言！例如，就在我们开始写这本书的时候，手枪的表情符号就已经从武器变成了玩具（至少在智能手机上显示时是这样），这反映了我们看待和使用该符号时文化层面的转变。

注 1： Fred Benenson, Emoji Dick, (2013) *http://bit.ly/2GKft1n*。

除了适应语言的新符号和结构，还要适应新定义、新的上下文和用法。“battery”一词在电子时代意义发生了改变，表示将化学能转化为电能的储存装置。但是从 Google Books Ngram Viewer[注2] 上可以看到，19 世纪末 20 世纪初，“battery”更为广泛地用作截然不同的含义，表示一系列相互连接的机器或强大的枪支装置。语言只有放在特定的上下文里才能被准确理解，这个上下文除了前后文字以外还包括所处的时间范围。准确识别单词含义需要的计算远比查字典复杂得多。

机器学习

自然语言的特殊性在胜任人类交流工具的同时，也让它很难用确定规则来进行解析。人类在语言理解方面的灵活性，决定了虽然只用到 60000 个符号，我们却可以在语言即时理解方面远远超过计算机。因此，在软件环境中，我们需要具有同样模糊性和灵活性的计算技术，统计机器学习是目前文本分析领域的最新技术。虽然自然语言处理的应用几十年前就有了，但因为有了机器学习才得以实现之前无法想象的一定程度的灵活性和响应能力。

机器学习的目标是将数据拟合到某个模型，创建现实世界的表示，该表示能基于发现的模式做出决策，生成对新数据的预测。实际上，这是通过选择符合输入数据与目标数据间关系的模型族来实现的，该模型族确定某种包含参数和特征的形式，然后用优化程序来最小化模型对训练数据的误差。在训练好的模型上引入新数据，就可以进行预测，返回标签、概率、归属或具体值。面临的最大挑战，是在准确学习已知数据模式和形成良好泛化能力之间取得平衡，这样模型才能在没见过的数据上表现良好。

很多语言类应用不是单一的机器训练模型，而是多个相互联系、相互影响的模型。模型可以用新数据重新训练、面向新的决策空间，甚至对每个用户进行定制，以便在输入新信息、或应用各方面随时间变化时，可以继续正常使用。在应用程序背后，各个相互竞争的模型可以竞争、衰退，并最终消亡。这意

注 2： Google, Google Books Ngram Viewer, (2013) *http://bit.ly/2GNlKtk*。

味着机器学习应用可以实现生命周期管理，通过日常维护和对工作流的监控，来实现和语言相关的动态性和局部特性。

文本分析工具

由于文本分析主要是应用机器学习技术，因此需要具有丰富的科学、数字计算库的语言。针对文本机器学习，Python 有一组强大的工具集，包括 Scikit-Learn、NLTK、Gensim、spaCy、NetworkX 和 Yellowbrick。

- *Scikit-Learn* 是 SciPy（Scientific Python）的扩展，提供通用机器学习 API。Scikit-Learn 构建于 Cython 之上，基于高性能 C 语言库，如 LAPACK、LibSVM、Boost 等，兼具高性能和易用性，最适合分析中小型数据集。Scikit-Learn 可用于开源和商业项目，其为回归、分类、聚类和降维模型提供了统一界面，同时提供了交叉验证和超参数调整的实用工具。

- *NLTK*，自然语言工具集，由学术界专家用 Python 编写，称得上是 NLP“内置电池”级资源。最早是面向教学的 NLP 工具集，包含语料库、词汇资源、语法、语言处理算法和预训练模型，可以帮助 Python 程序员快速开始处理各种语言的文本数据。

- *Gensim* 是个强大、高效、无障碍的工具库，专注于文本无监督语义建模。最初设计用来探索文档间的相似性（生成相似性），目前开放了基于潜在语义技术的主题建模方法，也包括其他无监督库，如 word2vec。

- *spaCy* 将学术领域最先进技术封装成简单易用的 API，提供产品级语言处理能力。特别是，spaCy 专注于面向深度学习的文本预处理，以及用大量文本构建信息提取系统或自然语言理解系统。

- *NetworkX* 是全面的图分析包，用于生成、序列化、分析和操纵复杂网络。虽然不是专门的机器学习或文本分析库，但图数据结构能对复杂关系进行

编码，图算法可以遍历图或在其中发现意义，是文本分析工具集不可或缺的一部分。

- *Yellowbrick* 是一套视觉诊断工具，用于分析和解释机器学习工作流。通过扩展 Scikit-Learn API，Yellowbrick 提供了直观、易于理解的特征选择、建模和超参数优化过程的可视化，引导模型选择过程找到最有效的文本数据模型。

本书的主要内容

本书中我们专注于用 Python 库进行应用机器学习的文本分析。本书侧重面向应用，意味着我们不会关注语言学或统计模型的学术内涵，而是关注如何在软件应用中用文本训练模型的有效部署。

我们所说的文本分析模型与机器学习工作流直接相关，找到由特征、算法和超参数组成的模型的搜索过程，使之能很好地匹配训练数据，并对未知数据生成最佳估计。该工作流从训练数据集的构建和管理开始，该数据集也叫做文本分析语料库。然后，我们将探索特征提取和预处理方法，进而将文本重组成机器学习可以理解的数字数据。基于掌握的一些基本特征，我们会探索文本分类和文本聚类技术，并对本书前几章进行总结。

之后的章节侧重于扩展能实现更丰富功能的模型，并创建文本感知应用。我们首先探索如何将上下文嵌入得到特征，然后对文本进行可视化解释，以指导后续的模型选择过程。接下来，我们将研究如何使用图分析技术，分析从文本中提取的复杂关系。然后，我们转而探索会话代理，并加深我们对文本句法和语义分析的理解。我们用基于多进程和 Spark 的可扩展文本分析的实践讨论来结束全书，最后，我们将探讨文本分析的下一阶段：深度学习。

本书的读者对象

本书面向有兴趣将自然语言处理和机器学习应用于其软件开发工具集的 Python 程序员。我们不要求读者具有特殊的学术背景或数学知识，而是专注

于工具和技术，避免冗长的解释。本书中我们主要分析英语语言，因此了解基本的语法知识，如名词、动词、副词和形容词如何相互关联，对理解本书是有帮助的。对从未接触过机器学习和语言学，但对 Python 编程有较深理解的读者，也不会对我们提到的概念感到不知所措。

使用示例代码和 GitHub 代码库

本书中的示例代码旨在描述如何编写 Python 代码，以执行特定任务。由于面向读者，所以代码比较简洁，通常省略执行时所需的关键语句，例如，从标准库导入的声明。此外，代码通常基于本书其他部分或章节中的代码构建，有时需要稍加修改，才能在新的场景下工作。例如，我们可以按如下方式定义一个类：

```
class Thing(object):

    def __init__(self, arg):
        self.property = arg
```

这个类定义用于描述类的基本属性，并设置关于实现细节的后续讨论的大体框架。稍后，我们可以向类中添加方法，如下所示：

```
...

    def method(self, *args, **kwargs):
        return self.property
```

请注意代码段开头的省略号，表示这是前一个代码段中类定义的延续。这意味着简单地复制和粘贴示例代码段可能无法运行。重要的是，代码可能会对存储在磁盘上的数据进行操作，而数据位于 Python 程序执行时可访问的位置。我们尽量保证通用，但无法考虑所有操作系统和数据源的情况。为了给可能想要运行本书中示例的读者提供支持，我们在 GitHub 代码库中保存了完整的可执行示例。这些示例可能与书中略有不同，但在任何系统上都应该可以用 Python 3 轻松执行。另外，请注意，代码库会保持更新，请查看 *README* 了解发生的任何改动。你当然可以分支存储并修改代码，以便在自己的环境中运行，我们强烈建议这样做！

排版约定

本书使用如下的排版约定：

斜体

表示新术语、URL、email 地址、文件名和文件扩展名。

等宽字体（`constant width`）

表示程序片段，以及书中出现的变量、函数名、数据库、数据类型、环境变量、语句和关键字等。

加粗等宽字体（**`constant width bold`**）

表示应该由用户输入的命令或其他文本。

等宽斜体（*`constant width italic`*）

表示应该由用户输入的值或根据上下文确定的值替换的文本。

表示提示或建议。

表示一般性说明。

表示警告或提醒。

使用代码示例

补充材料（代码示例、练习等）可在这里下载：*https://github.com/foxbook/atap*。

本书的目的是帮助你完成工作。一般来说，你可以在自己的程序或者文档中使用本书附带的示例代码。你无需联系我们获得使用许可，除非你要复制大量的代码。例如，使用本书中的多个代码片段编写程序就无需获得许可。但以 CD-ROM 的形式销售或者分发 O'Reilly 书中的示例代码则需要获得许可。回答问题时援引本书内容以及书中示例代码，无需获得许可。在你自己的项目文档中使用本书大量的示例代码时，则需要获得许可。

我们不强制要求署名，但如果你这么做，我们深表感激。署名一般包括书名、作者、出版社和书号。例如："Applied Text Analysis with Python by Benjamin Bengfort, Rebecca Bilbro, and Tony Ojeda (O'Reilly). 978-1-491-96304-3"。

本书的 BibTex 引用信息如下：

```
@book{
    title = {Applied {{Text Analysis}} with {{Python}}},
    subtitle = {Enabling Language Aware {{Data Products}}},
    shorttitle = {Applied {{Text Analysis}} with {{Python}}},
    publisher = {{O'Reilly Media, Inc.}},
    author = {Bengfort, Benjamin and Bilbro, Rebecca and Ojeda, Tony},
    month = jun,
    year = {2018}
}
```

如果你觉得自身情况不在合理使用或超出上述许可范围内，请通过邮件和我们联系，地址是：*permissions@oreilly.com*。

O'Reilly Safari

Safari（以前叫做 Safari Books Online）是一个会员制的培训和参考平台，面向企业、政府、教育工作者和个人。会员可访问来自 250 多家出版商的数千本书籍、培训视频、学习路径、互动教程和精选播放列表，包括 O'Reilly Media，Harvard Business Review，Prentice Hall Professional，Addison-Wesley Professional，Microsoft Press，Sams，Que ，Peachpit Press，Adobe，Focal Press，Cisco Press，John Wiley & Sons，Syngress，Morgan Kaufmann，IBM Redbooks，Packt，Adobe Press，FT Press，Apress，Manning，New Riders，McGraw-Hill，Jones & Bartlett 和 Course Technology 等。

更多信息，请访问 *http://oreilly.com/safari*。

联系我们

请把对本书的评价和问题发给出版社：

美国：

O'Reilly Media, Inc.
1005 Gravenstein Highway North
Sebastopol, CA 95472

中国：

北京市西城区西直门南大街2号成铭大厦C座807室（100035）
奥莱利技术咨询（北京）有限公司

本书有一个专属网页（*http://bit.ly/applied-text-analysis-with-python*），你在上面可以找到图书的相关信息，包括勘误表、示例代码，以及其他信息。

对于本书的评论和技术性问题，请发送电子邮件到：*bookquestions@oreilly.com*。

要了解更多 O'Reilly 图书、培训课程、会议和新闻的信息，请访问以下网站：*http://www.oreilly.com*。

我们的 Facebook：*http://facebook.com/oreilly*。

我们的 Twitter：*http://twitter.com/oreillymedia*。

我们的 YouTube：*http://www.youtube.com/oreillymedia*。

致谢

感谢我们的技术评审员，他们在第一稿花费了大量时间和精力，为本书提