TURING 图灵程序设计丛书

Packt

Feature Engineering Made Easy

特征工程入门与实践

[土] 锡南·厄兹代米尔 迪夫娅·苏萨拉 著
庄嘉盛 译

- 从零入手，全面了解特征工程
- 提升机器学习算法的效率、准确率

中国工信出版集团
人民邮电出版社
POSTS & TELECOM PRESS

TURING 图灵程序设计丛书

Feature Engineering Made Easy

特征工程入门与实践

[土] 锡南·厄兹代米尔 迪夫娅·苏萨拉 著
庄嘉盛 译

人民邮电出版社
北京

图书在版编目（CIP）数据

特征工程入门与实践 /（土）锡南·厄兹代米尔（Sinan Ozdemir），（土）迪夫娅·苏萨拉（Divya Susarla）著；庄嘉盛译. -- 北京：人民邮电出版社，2019.6
（图灵程序设计丛书）
ISBN 978-7-115-51164-5

Ⅰ. ①特… Ⅱ. ①锡… ②迪… ③庄… Ⅲ. ①机器学习 Ⅳ. ①TP181

中国版本图书馆CIP数据核字(2019)第080456号

内 容 提 要

机器学习模型的成功正是取决于如何利用不同类型的特征，例如连续特征、分类特征等。本书将带你了解特征工程的完整过程，使机器学习更加系统、高效。你会从理解数据开始学习，了解何时纳入一项特征、何时忽略一项特征，以及其中的原因。你还会学习如何将问题陈述转换为有用的新特征，如何提供由商业需求和数学见解驱动的特征，以及如何在自己的机器上进行机器学习，从而自动学习数据中的特征。

本书面向所有希望全面了解特征工程的读者，特别适合具有机器学习应用知识并希望改进机器学习模型结果的数据科学家阅读。

◆ 著　[土] 锡南·厄兹代米尔　迪夫娅·苏萨拉
译　庄嘉盛
责任编辑　杨　琳
责任印制　周昇亮

◆ 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
三河市祥达印刷包装有限公司印刷

◆ 开本：800×1000　1/16
印张：13.75
字数：324千字　2019年6月第1版
印数：1-3 000册　2019年6月河北第1次印刷
著作权合同登记号　图字：01-2018-4184号

定价：59.00元

读者服务热线：(010)51095183转600　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

站在巨人的肩上

Standing on Shoulders of Giants

iTuring.cn

前　言

本书的主题是特征工程。特征工程是数据科学和机器学习流水线上的重要一环，包括识别、清洗、构建和发掘数据的新特征，为进一步解释数据并进行预测性分析做准备。

本书囊括了特征工程的全流程，从数据检查到可视化，再到转换和进一步处理，等等。书中还会涉及各种或简单或复杂的数学工具，数据要经过这些工具处理、转换成适当的形式，才能进入计算机和机器学习流水线中进行处理。

作为数据科学家，我们将通过观察和变换来获取对数据的全新理解，这不仅会增强机器学习算法的效果，而且会增强我们对数据的洞悉力。

目标读者

本书面向希望理解并使用特征工程进行机器学习和数据挖掘的读者。

读者应能熟练使用 Python 进行机器学习和编程，才能顺着章节的展开循序渐进地了解新知识点。

本书内容

第 1 章，特征工程简介　这一章介绍特征工程的基本术语，简要阐释本书涉及的各类问题。

第 2 章，特征理解：我的数据集里有什么　这一章介绍我们在实际中会遇见的各类数据，并说明如何处理这些数据。

第 3 章，特征增强：清洗数据　这一章介绍填充缺失值的各种方法，以及为何某些处理方法会使机器学习性能变差。

第 4 章，特征构建：我能生成新特征吗　这一章介绍如何使用已有的特征构建新特征，以扩大数据集。

第 5 章，特征选择：对坏属性说不　这一章介绍定量的选择方法，用于判断哪些特征值得在

数据流水线中保留。

第 6 章，特征转换：数学显神通 这一章介绍如何使用线性代数和高等数学方法增强数据的刚性结构，从而提升流水线的性能。

第 7 章，特征学习：以 AI 促 AI 这一章介绍如何利用最先进的机器学习和人工智能算法，发现人类难以理解的特征。

第 8 章，案例分析 这一章介绍了一系列巩固特征工程思想的案例。

阅读须知

阅读本书有以下两点要求。

(1) 本书的所有编程示例均使用 Python。你需要有一台可以访问 Unix 式终端的计算机（Linux、Mac 或 Windows 均可），并安装 Python 3。

(2) 建议安装 Anaconda，因为这个环境几乎包含了示例中要用到的所有包。

下载示例代码

你可以从“图灵社区”本书页面（http://www.ituring.com.cn/book/2606）下载书中的示例代码。

文件下载结束之后，请确定使用以下软件的最新版本解压或提取文件：

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

https://github.com/PacktPublishing/提供了种类丰富的图书和视频资料相关代码包，好好看一下吧！

下载本书彩色图片

我们也提供含有彩色截图/图表的 PDF 文件。彩色图片能帮助你更深入地理解输出的变化。下载地址：https://www.packtpub.com/sites/default/files/downloads/FeatureEngineeringMadeEasy_ColorImages.pdf。

排版约定

本书采用不同的文本样式来区分不同类别的信息。

正文中的代码按以下样式显示："假设要进一步处理数据，我们的任务就是通过 3 个输入特征（datetime、protocol 和 urgent）准确地预测 malicious。简单地说，我们想要的系统可以把 datetime、protocol 和 urgent 的值映射到 malicious 的值。"

代码块的样式如下所示：

```
Network_features = pd.DataFrame({'datetime': ['6/2/2018', '6/2/2018',
'6/2/2018', '6/3/2018'], 'protocol': ['tcp', 'http', 'http', 'http'],
'urgent': [False, True, True, False]})
Network_response = pd.Series([True, True, False, True])
Network_features
>>
  datetime   protocol  urgent
0   6/2/2018      tcp  False
1   6/2/2018     http   True
2   6/2/2018     http   True
3   6/3/2018     http  False
Network_response
>>
0       True
1       True
2      False
3       True
dtype: bool
```

如果我们需要你重点关注某处，会加粗显示：

```
times_pregnant                0.221898
plasma_glucose_concentration  0.466581
diastolic_blood_pressure      0.065068
triceps_thickness             0.074752
serum_insulin                 0.130548
bmi                           0.292695
pedigree_function             0.173844
age                           0.238356
onset_diabetes                1.000000
Name: onset_diabetes, dtype: float64
```

新术语、重点词和屏幕上的文字将以黑体形式显示。

这个图标表示警告或需要特别注意的内容。

这个图标表示提示或技巧。

联系我们

一般反馈：发送邮件至 feedback@packtpub.com 并在主题处提及书名。如果对于本书任何方面有疑问，请发送邮件至 questions@packtpub.com。

勘误：尽管我们做了各种努力来保证内容的准确性，依然无法避免出现错误。如果你在书中发现文字或代码错误，请告知我们，我们将非常感谢。请访问 https://www.packtpub.com/submit-errata 提交勘误。[①]通过点击 Errata Submission Form 链接选择图书，然后输入勘误详情。

反盗版：如果你在网上发现有对我们图书的非法复制行为，请立即将地址或网站名通知我们，非常感谢。请联系 copyright@packtpub.com 并提供有盗版嫌疑的链接。

成为作者：如果你在某个领域有专业知识，并且有兴趣进行图书写作，请访问 authors.packtpub.com。

评论

请留下你的评论。阅读并使用本书之后，为什么不在购买网站上留下评论呢？其他读者可以根据你的客观意见来做出购买决定，Packt 可以了解你对产品有何看法，作者也能看到你对本书的反馈。谢谢！

想了解关于 Packt 的更多信息，请访问 packtpub.com。

电子书

扫描如下二维码，即可购买本书电子版。

① 针对本书中文版的勘误，请到 http://www.ituring.com.cn/book/2606 查看和提交。——编者注

目　录

第 1 章 特征工程简介

近年来，工程师和管理层都在尝试用**机器学习**（ML，machine learning）和**人工智能**（AI，artificial intelligence）来解决以往需要手动操作才能处理的问题。**自然语言处理**（NLP，natural language processing）的发展就是个很好的例子，特别是在自然语言生成和理解方面。更精确地说，我们希望构建一个 AI 系统，从用户（例如对最新款智能手机不满的用户）那里直接读取原始文本，以机器的速度和人类的措辞来流畅、准确地回复。本章会介绍以下有关特征工程的主题：

- 几个激动人心的例子，解释为什么特征工程至关重要；
- 对机器学习的基本理解，包括性能和评估方法；
- 本书各章的详细信息。

1.1 激动人心的例子：AI 驱动的聊天

我们的 AI 聊天系统 Arty 可以像人类员工一样回复客户支持请求。Arty 了解我们公司产品，可以随时提供服务。

下面是人与 AI 客服系统的一段对话。

人　类	AI
你好，我的手机坏了。	抱歉给您带来了不便，请问您的手机是怎么坏的？
手机死机了，我没法重启。	您的手机是什么型号的？
新款 iDroid 28	啊，知道了。按住电源键和音量减键 20 秒，手机就重启了。
可以了，谢谢！	不客气，祝您生活愉快。

这种系统令人兴奋，足以撼动市场，因为如此复杂的系统竟然可以十分简洁。让我们仔细分析一下。从表面上看，你有可能觉得，**这问题很简单啊！**提问很简单，回答也很简单，只是接收一个请求，给出一个回复。**“您好，我的手机死机了，应该怎么办？”**很简单，重启就好了。当然，表面上看起来一定是这样的。

```
from Arty import AI
AI.respond_to("my phone froze, what should I do?")
>> "reset it."
```

问题的难点在于要从AI的角度看问题。AI没有完整的人类经验，它既没读过古希腊史诗《伊利亚特》，也没看过儿童故事书《大红狗克里弗》，没办法消化信息。核心问题就是，AI没有什么阅读的经验。人们给这个AI几十万份（乃至几百万份）之前的人类聊天日志，让其从中发现规律。

上文中AI系统的训练数据节选如下。

请　求	回　复
你哈哦	您好，请问您需要什么帮助？
我手机坏了！！！！！！	天啊！！！！怎么了？
>等一下，我去遛狗。马上回来。	好。我等你。
嗨	您好，我是Mark，请问您需要什么帮助？

数据分为两列，**请求**表示最终用户输入客服聊天框的内容，**回复**则表示客服对所收到消息的回复。

在读过数千条包含错别字、脏话和中途掉线的聊天记录后，AI开始认为自己可以胜任客服工作了。于是，人类开始让AI处理新收到的消息。虽然人类没有意识到自己的错误，但是开始注意到AI还没有完全掌握这项本领。AI连最简单的消息都识别不了，返回的消息也没有意义。人类很容易觉得AI只是需要更多的时间和更多的数据，但是这些解决方案只是更大问题的小修小补，而且很多时候根本不管用。

这个例子中的潜在问题很有可能是AI的原始输入数据太差，导致AI认识不到语言中的细微差别。例如，问题可能出在这些地方。

- 错别字会无故扩大AI的单词量。“你哈哦”和“你好”是两个无关的词。
- AI不能理解同义词。用来打招呼的“你好”和“嗨”字面上看起来毫不相似，人为地增加了问题的难度。

1.2　特征工程的重要性

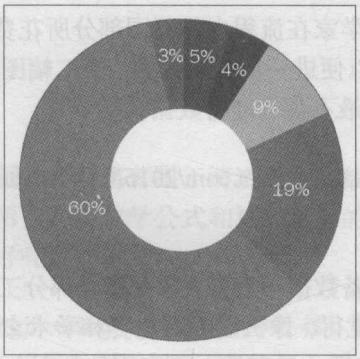
为了解决实际问题，数据科学家和机器学习工程师要收集大量数据。因为他们想要解决的问题经常具有很高的相关性，而且是在混乱的世界中自然形成的，所以代表这些问题的原始数据有可能未经过滤，非常杂乱，甚至不完整。

因此，过去几年来，类似**数据工程师**的职位应运而生。这些工程师的唯一职责就是设计数据流水线和架构，用于处理原始数据，并将数据转换为公司其他部门——特别是数据科学家和机器学习工程师——可以使用的形式。尽管这项工作和机器学习专家构建机器学习流水线一样重要，但是经常被忽视和低估。

在数据科学家中进行的一项调查显示，他们工作中超过 80%的时间都用在捕获、清洗和组织数据上。构造机器学习流水线所花费的时间不到 20%，却占据着主导地位。此外，数据科学家的大部分时间都在准备数据。超过 75%的人表示，准备数据是流程中最不愉快的部分。

上文提到的调查结果如下。

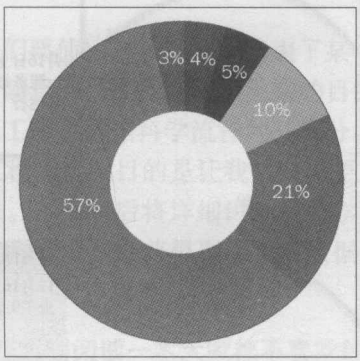
下图展示了数据科学家进行不同工作的时间比例。



从上图可见，数据科学家的工作占比如下。

- 设置训练集：3%
- 清洗和组织数据：60%
- 收集数据集：19%
- 挖掘数据模式：9%
- 调整算法：5%
- 其他：4%

下图展示了数据科学家最不喜欢的流程。



在一项类似的调查中，数据科学家认为他们最不喜欢的流程如下。

- 设置训练集：10%
- 清洗和组织数据：57%
- 收集数据集：21%
- 挖掘数据模式：3%
- 调整算法：4%
- 其他：5%

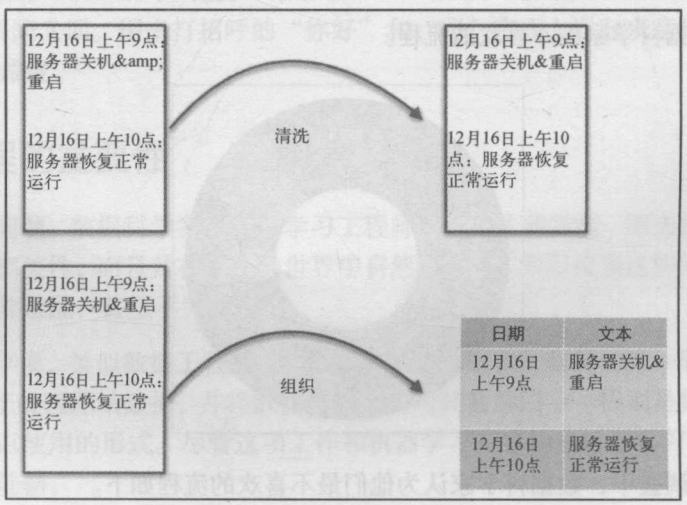
上面第一幅图表示了数据科学家在流程中的不同部分所花费的时间比例。数据科学家有超过80%的时间花在了准备数据上，以便进一步利用数据。第二幅图则表示了数据科学家最不喜欢的步骤。超过75%的人表示，他们最不喜欢准备数据。

数据源：https://whatsthebigdata.com/2016/05/01/data-scientists-spend-most-of-their-time-cleaning-data/。

好的数据科学家不仅知道准备数据很重要，会占用大部分工作时间，而且知道这个步骤很艰难，没人喜欢。很多时候我们会觉得，像机器学习竞赛和学术文献中那样干净的数据是理所当然的。然而实际上，超过90%的数据（最有趣、最有用的数据）都以原始形式存在，就像在之前AI聊天系统的例子中一样。

准备数据的概念很模糊，包括捕获数据、存储数据、清洗数据，等等。之前的图中显示，清洗和组织数据占用的工作时间十分可观。数据工程师在这个步骤中能发挥最大作用。清洗数据的意思是将数据转换为云系统和数据库可以轻松识别的形式。组织数据一般更为彻底，经常包括将数据集的格式整体转换为更干净的格式，例如将原始聊天数据转换为有行列结构的表格。

清洗数据和**组织数据**的区别如下图所示。



图片上半部分的转换代表清洗服务器日志，包含数据和服务器状态的描述文本。注意在清洗时，Unicode 字符&被转换为了更可读的&。清洗前后，文档的格式基本保持不变。下半部分的组织转换则彻底得多，把原始数据转换为了行列结构，其中每行代表服务器的一次操作，每列代表服务器操作的属性（attribute）。在这个例子中，两个属性是**日期**和**文本**。

清洗和组织数据都属于更大的数据科学范畴，也是本书要讨论的主题——特征工程。

1.3 特征工程是什么

终于说到本书的主题了。

是的，本书的主题是特征工程。我们将着眼于清洗和组织数据的过程，为机器学习流水线服务。除了这些概念，我们还会介绍如何用数学公式和神经理解的方式看待数据转换，但是现在暂时不涉及。让我们从概念开始入手吧。

特征工程（feature engineering）*是这样一个过程：将数据转换为能更好地表示潜在问题的特征，从而提高机器学习性能。*

为了进一步理解这个定义，我们看看特征工程具体包含什么。

- **转换数据的过程**：注意这里并不特指原始数据或未过滤的数据，等等。特征工程适用于任何阶段的数据。通常，我们要将特征工程技术应用于在数据分发者眼中已经**处理过**的数据。还有很重要的一点是，我们要处理的数据经常是表格形式的。数据会被组织成行（观察值）和列（属性）。有时我们从最原始的数据形式开始入手，例如之前服务器日志的例子，但是大部分时间，要处理的数据都已经在一定程度上被清洗和组织过了。
- **特征**：显而易见，这个词在本书中会很常用。从最基本的层面来说，特征是对机器学习过程有意义的数据属性。我们经常需要查看表格，确定哪些列是特征，哪些只是普通的属性。
- **更好地表示潜在问题**：我们要使用的数据一定代表了某个领域的某个问题。我们要保证，在处理数据时，不能一叶障目不见泰山。转换数据的目的是要更好地表达更大的问题。
- **提高机器学习性能**：特征工程是数据科学流程的一部分。如我们所见，这个步骤很重要，而且经常被低估。特征工程的最终目的是让我们获取更好的数据，以便学习算法从中挖掘模式，取得更好的效果。本书稍后将详细讨论机器学习的指标和效果，但是现在我们要知道的是，执行特征工程不仅是要获得更干净的数据，而且最终要在机器学习流水线中使用这些数据。

你一定在想：**为什么我应该花时间阅读一本大家都不喜欢的事情的书？**我们觉得，很多人之所以不喜欢特征工程，是因为他们常常看不到这些工作的益处。

大部分公司会同时招聘数据工程师和机器学习工程师。数据工程师主要关注准备和转换数据，而机器学习工程师一般拥有算法知识，知道如何从清洗好的数据中挖掘出模式来。

这两种工作一般是分开的，但是会交织在一起循环进行。数据工程师把数据集交给机器学习工程师，机器学习工程师则会说结果不好，让数据工程师进一步转换数据，反反复复。这种过程不仅单调重复，而且影响大局。

如果工程师不具备特征工程和机器学习两方面的知识，则整个流程很有可能不会那么有效。因此本书应运而生。我们会讨论特征工程，以及特征工程和机器学习如何直接相关。这个方法是以结果为导向的，我们认为，只有能提高机器学习效果的技术才是有用的技术。现在我们来深入了解数据、数据结构和机器学习的基础知识，以确保术语的统一性。

数据和机器学习的基础知识

一般来说，我们处理的数据都是表格形式的，按行列组织。可以将其想象成能在电子表格程序（例如 Microsoft Excel）中打开。数据的每行又称为**观察值**（observation），代表问题的一个实例或例子。例如，如果数据是关于股票日内交易的，那么每个观察值有可能是一小时内整体股市和股价的涨跌。

又例如，如果数据是关于网络安全的，那么观察值也许是可能的黑客攻击，或者是无线网络发送的一个数据包。

下表是网络安全领域的示例数据，确切地说是网络入侵领域。

DateTime	Protocol	Urgent	Malicious
June 2nd, 2018	TCP	FALSE	TRUE
June 2nd, 2018	HTTP	TRUE	TRUE
June 2nd, 2018	HTTP	TRUE	FALSE
June 3rd, 2018	HTTP	FALSE	TRUE

可以看到，每行（每个观察值）都是一次网络连接，有 4 个属性：`DateTime`（日期）、`Protocol`（协议）、`Urgent`（紧急）和 `Malicious`（恶意）。我们暂时不深入研究每个属性，先观察以表格形式给出的数据结构。

因为大部分数据都是表格形式的，也可以看看一种特殊的实例：数据只有一列（一个属性）。例如，我们要开发一个软件，输入房间的一张图像，它会输出房间中是否有人。输入的数据矩阵有可能只有一列——房间照片的链接（URL），别的什么都没有。

例如，下面的表格中只有一列，列标题是“照片 URL”。表格中数据的值（这些 URL 仅为示例，并不指向真的图片）对数据科学家而言具有相关性。

照片 URL
http://photo-storage.io/room/1
http://photo-storage.io/room/2
http://photo-storage.io/room/3
http://photo-storage.io/room/4

输入的数据有可能只有一列，像这个例子一样。在创建图像分析系统时，输入有可能仅仅是图像的 URL。作为数据科学家，我们要从这些 URL 中构建特征。

数据科学家要准备好接受并处理多或少、宽或窄（从特征上讲）、完整或稀疏（可能有缺失值）的数据，并准备好在机器学习中应用这些数据。现在是时候讨论机器学习了。机器学习算法是按其从数据中提取并利用模式、以基于历史训练数据完成任务的能力来定义的。是不是摸不到头脑？机器学习可以处理很多类型的任务，因此我们不给出定义，而是继续深入探讨。

大体上，我们把机器学习分为两类：监督学习和无监督学习。两种算法都可以从特征工程中获益，所以了解每种类型非常重要。

1. 监督学习

一般来说，我们都是在监督学习（也叫预测分析）的特定上下文中提到特征工程。监督学习算法专门处理预测一个值的任务，通常是用数据中的其他属性来预测余下的一个属性。以如下表示网络入侵的数据集为例。

DateTime	Protocol	Urgent	Malicious
June 2nd, 2018	TCP	FALSE	TRUE
June 2nd, 2018	HTTP	TRUE	TRUE
June 2nd, 2018	HTTP	TRUE	FALSE
June 3rd, 2018	HTTP	FALSE	TRUE

还是前文用到的数据集，这次我们在预测分析的上下文中深入探讨。

注意，数据集有 4 个属性：`DateTime`、`Protocol`、`Urgent` 和 `Malicious`。假设 `Malicious` 属性包含代表该观测值是否为恶意入侵的值。所以在这个小数据集中，第 1 次、第 2 次和第 4 次连接都是恶意入侵。

进一步假设，在这个数据集中，我们要尝试用 3 个属性（`DateTime`、`Protocol` 和 `Urgent`）准确预测 `Malicious` 属性。简单地说，我们想建立一个系统，将 `DateTime`、`Protocol` 和 `Urgent` 属性的值映射到 `Malicious` 的值。监督学习问题就是这样建立起来的：

```
Network_features = pd.DataFrame({'datetime': ['6/2/2018', '6/2/2018',
'6/2/2018', '6/3/2018'], 'protocol': ['tcp', 'http', 'http', 'http'],
'urgent': [False, True, True, False]})
Network_response = pd.Series([True, True, False, True])
Network_features
>>
```