

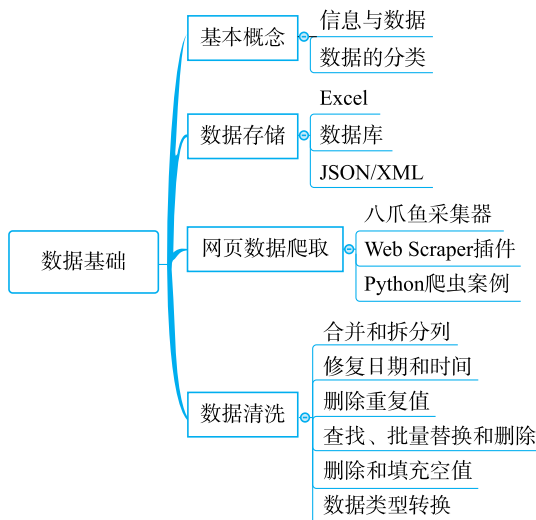
第 1 部 分

可视化概述



- 第 1 章 数据基础
- 第 2 章 数据分析
- 第 3 章 可视化基础

数据基础



1.1 基本概念

数据是人们获取信息、进行决策和实施管理的重要依据。在信息爆炸的时代,数据无处不在,无论是日常生活中的社交平台、传感设备所采集的数据,还是企业内部的交易记录,都属于数据范畴。本节主要介绍信息与数据的内涵、二者之间的关系,以及数据的常见分类方法。

1.1.1 信息与数据

数据(Data)通常指经过收集和存储的原始事实、数字、符号、文字或图像等。这些数据本身并不具备明确的含义,需要经过处理和分析才能转化成有用的信息。例如,一串未加解释的数字序列可能仅代表一些测量值或交易记录,只有经过统计、归纳、提炼和分析,才能成为企业决策的重要依据。

信息(Information)是经过加工、整理、分析之后形成的具有实际意义的信息。信息强调的是数据的语境化和背景知识的注入,使得原始数据转变为能够指导行为、支持判断的知识。简单来说,数据是“原材料”,而信息则是“成品”。这种转化过程往往依赖特定的领域知识、算法处理以及数据关联关系的发现。

数据与信息之间的关系

- (1) 数据是构成信息的基础,没有数据就无法获得信息。
- (2) 信息是数据加工后的成果,是数据蕴含价值的体现。
- (3) 在信息传递过程中,还可能出现噪声和误差,因此数据质量直接影响信息的准确性。

通过对信息与数据的区别和内在联系的理解,我们能够更清楚地认识到收集和整理数据的重要性,也能够明白为何需要从海量的原始数据中提取有价值的信息。

1.1.2 数据的分类

在实际应用中,数据往往来源广泛、形式各异,因此对数据进行分类具有重要意义。在数据可视化的过程中,不同类型的数据在处理方法、存储模式及展示手段上都有所不同。常见的数据分类方法主要包括按照数据结构分类、按照数据来源分类,以及按照数据用途分类等。这里我们重点讨论按照数据结构的分类方式,即将数据分为结构化数据、半结构化数据和非结构化数据。

1. 结构化数据

结构化数据(Structured Data)是指存储在固定格式中的数据,其数据项之间具有明确的关系。结构化数据通常存在于数据库表中,每一行代表一条记录,每一列代表某一特定属性或数据项。以客户信息表为例,通常包含客户 ID、姓名、年龄、联系方式及购买记录等字段。各字段的数据类型可能包括文本型、数值型、日期型等,每种数据类型都遵循预先定义的数据格式。这种固定的模式使得数据在输入时必须满足一定的约束条件,从而确保数据质量和一致性。常见的结构化数据包括关系数据库中的表格数据、电子表格(如 Excel 文档)、CSV 文件等,其特点如下。

- **固定模式**: 数据按照预先设计的表结构存储,各字段的含义和数据类型均已定义。
- **易于查询**: 使用 SQL 等标准语言可对数据进行复杂查询和统计。
- **高度组织化**: 数据之间存在明显的关系,可以通过主键、外键等进行关联。

在数据可视化中,结构化数据常被直接用于绘制柱状图、折线图、饼图等传统图表,因为其便于按照行列进行汇总、统计和分析。结构化数据举例如表 1.1 所示。

表 1.1 结构化数据举例

姓 名	年 龄	部 门	职 位
张三	30	研发	工程师
李四	28	市场	产品经理

2. 半结构化数据

半结构化数据(Semi-structured Data)介于结构化数据与非结构化数据之间,其数据格式不具有严格的表格式结构,但内部包含一定的标记或标签,可用于描述数据本身的结构信息。典型的半结构化数据格式有 JSON、XML、YAML 等,其特点如下。

- **灵活性高**: 数据中可以包含嵌套关系、层级结构,允许数据格式在一定范围内变化。

- **自描述性**: 标签和标记能够帮助理解数据的含义,即使缺乏完整的模式定义,也能从中提取出有意义的信息。
- **易于扩展**: 当数据需要增加新的属性时,无须修改整体数据格式。

对于数据可视化而言,处理半结构化数据往往需要先对数据进行解析和转换,将嵌套或层次信息整理成适合图表展示的格式。JSON 和 XML 格式的半结构化数据举例如图 1.1 所示。

JSON	XML
<pre>{ "姓名": "张三", "年龄": 30, "部门": "研发", "职位": "工程师" }, { "姓名": "李四", "年龄": 28, "部门": "市场", "职位": "产品经理" }</pre>	<pre><Employees> <Employee> <Name>张三</Name> <Age>30</Age> <Department>研发</Department> <Position>工程师</Position> </Employee> <Employee> <Name>李四</Name> <Age>28</Age> <Department>市场</Department> <Position>产品经理</Position> </Employee> </Employees></pre>

图 1.1 JSON 和 XML 格式的半结构化数据举例

3. 非结构化数据

非结构化数据(Unstructured Data)是指没有预定义数据模型或结构的数据,包括文本、图片、音频、视频等,其特点如下。

- **无固定模式**: 数据以原始形式存储,不存在预定的字段或记录结构。
- **多样性高**: 数据形式多种多样,需要借助自然语言处理、图像识别等技术进行信息提取。
- **数据量大且复杂**: 在大数据环境下,非结构化数据往往占比较大,如何进行快速分析和关联处理成为一大挑战。

在数据可视化场景中,非结构化数据通常需要经过预处理(如文本挖掘、图像处理、语音识别等步骤),再将提取出的关键信息进行整理,最终用于构建词云、关系网络图等可视化展示。文本型非结构化数据举例如图 1.2 所示。

张三, 今年30岁, 目前在研发部担任工程师, 主要负责技术开发和维护工作。
李四, 28岁, 就职于市场部, 担任产品经理, 负责产品策略制定和市场推广。

图 1.2 文本型非结构化数据举例

通过对这三种数据类型的区分,同学们可以认识到不同数据格式对处理流程和方法选

择的影响。在实际应用中,常常会遇到多种数据并存的情况,因此掌握数据分类的知识有助于更灵活地选择合适的工具和算法来解析数据,并最终将数据转换为直观易懂的图表。

1.2 数据存储

数据存储的方式有很多,如 Excel、数据库,或者 JSON、XML 文件等,每种存储方式都有其优缺点和适用场景。

1.2.1 Excel

Excel 是 Microsoft 为使用 Windows 和 macOS 操作系统的计算机编写的一款电子表格软件。直观的界面、出色的计算功能和图表工具,再加上成功的市场营销,使 Excel 成为最流行的个人计算机数据处理软件。

对于个人用户来说,使用 Excel 进行数据存储和数据分析非常方便:成本低,功能足够强大,几乎可以处理各种数据;工具栏可以快速使用各种功能;日常工作中常用的表格处理操作,如增加行、删除列、合并单元格、表格转置等,在 Excel 中通过菜单或工具按钮即可完成;提供了包括财务、日期与时间、数学与三角函数、统计、查找与引用、数据库、文本、逻辑和信息等九大类数百个内置函数,可以满足许多领域的数据处理与分析的要求;具有丰富强大的绘制图表功能,可以直观地显示出数据的众多特征,例如数据的最大值、最小值、发展变化趋势、集中程度和离散程度等。

Excel 也有一些局限性,具体如下。

(1) 数据量。数据量不能过多,这是 Excel 最大的局限性。当 Excel 的数据量过大时,它的查询和计算的速度会明显下降。

(2) 数据安全性。Excel 提供的安全性有限,它只能限制用户访问和修改的权限,但是无法对用户进行角色的管理,也不能对数据进行行级的访问限制。Excel 在一些简单的破解程序面前毫无招架之力,其数据的安全性无法得到保证。

(3) 多用户管理。因为 Excel 是一个单机程序,所以一个 Excel 文件通常无法被多个用户同时管理。

(4) 突发性。面对突发事故(如数据丢失、修改异常等),对 Excel 进行回滚操作不方便。

1.2.2 数据库

在操作系统出现之后,随着计算机应用范围的扩大,需要处理的数据量迅速膨胀。最初,数据与程序一样,以简单的文件作为主要存储方式。以这种方式存储的数据在逻辑上更简单,但可扩展性差,访问这种数据的应用程序需要先了解数据的具体存储格式。当系统数据量大或用户访问量小时,应用程序还需要解决数据的完整性、一致性及安全性等一系列的问题。因此,需要有一种系统软件能够像操作系统屏蔽了硬件访问复杂性那样,屏蔽数据访问的复杂性。由此产生了数据管理系统,即数据库。

所谓“数据库”是以一定方式存储在一起、能与多个用户共享、具有尽可能小的冗余度、与应用程序彼此独立的数据集合。数据库是存储大量乃至海量的数据的首要选择,它在存

储数据方面有许多优势,具体如下。

(1) 存储容量大,占用空间少。数据库可以存储海量的数据类型,如文字、图片、音频和视频。例如,视频网站会使用数据库保存用户上传的视频信息。

(2) 字段定义明确,格式统一。数据库的每个字段都有预设的数据类型(如整型、日期、字符串等),不易出错。例如,“出生日期”列只能填入合法的日期格式,避免了格式混乱。

(3) 安全性高。数据库能通过用户验证、权限控制、加密等方式防止数据被非法访问或篡改。例如,银行系统会对用户账户数据加密,并记录访问日志,确保数据安全。

(4) 备份灵活,恢复方便。数据库支持多种备份方式,如全备份、增量备份等。当出现误删、系统崩溃时,可快速恢复数据。例如,企业每天对订单数据进行定时备份,一旦出错可迅速还原。

数据库的缺点是可视化程度不如 Excel 直观,且部分非开源数据库需要收取高昂的费用。此外,Excel 常用可视化交互操作,而数据库的核心操作通常需要借助代码来实现,上手难度更大。一般的做法是用数据库来存储数据,然后进行清洗、提取处理,输出需要二次加工的少量数据到 Excel 中进行处理,或者直接用 Excel 呈现数据结果。数据库的字段设置举例如图 1.3 所示。

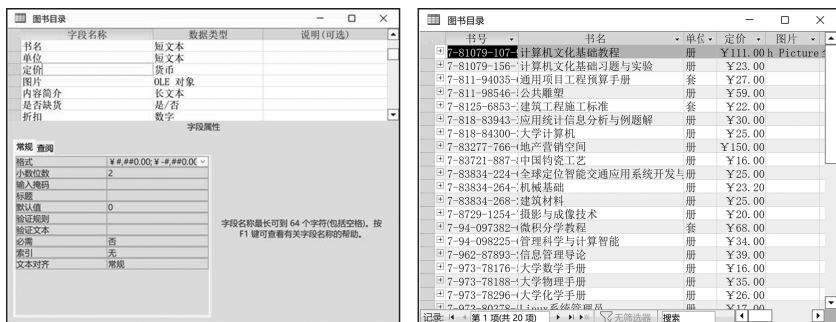


图 1.3 数据库的字段设置举例

1.2.3 JSON/XML

JavaScript 对象表示法(JavaScript Object Notation,JSON)是一种开放标准的文件格式和数据交换格式,易于用户阅读和编写,同时也易于机器解析和生成。JSON 独立于语言设计,很多编程语言都支持 JSON 格式的数据交换。JSON 最初被构想和设计为一种轻量级数据交换格式,其内容由属性和值组成。相对于传统的关系数据库,一些基于文档存储的非关系数据库选择 JSON 作为其数据存储格式,如 MongoDB、CouchDB、RavenDB 等。

如图 1.4 所示,JSON 格式的数据以键-值对的形式存在,可以通过键找到对应的值,比如通过“name”这个键,可以找到“霸王别姬”这个值。JSON 格式的基本语法非常简单、整洁,只由花括号{}、方括号[]、逗号和冒号组成,剩下的大多是有用的数据。由此可见,JSON 格式的数据利用率非常高。

在 JSON 格式的语法中,花括号{}中存储的是一个对象,一对花括号就是一个数据对象,每个数据对象下面还可以有对象,存在继承关系,表明对象之间的层次关系;方括号[]中存储的是数组,数组可以存储更多的数据,且不散,逗号用于分隔数据,让数据成为一个整

体,不能与其他数据混合;冒号用于对象,使用键-值对的形式表示,键是属性名,后面使用冒号与数据分开,且具有对应关系。

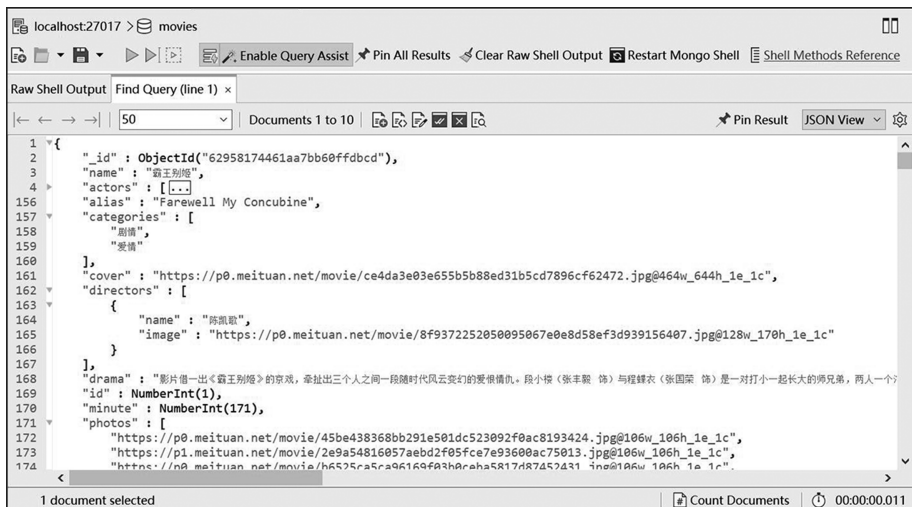


图 1.4 以 JSON 格式存储在 MongoDB 数据库中的数据

可扩展标记语言(Extensible Markup Language,XML)是标准通用标记语言的子集,可以用来标记数据,定义数据类型,是一种允许用户对自己的标记语言进行定义的源语言。XML 能够通过不同的标签来标记不同的内容,且使用的标签是自定义的。XML 通过在标签中添加属性这一简单的方法来存储元数据。

XML 被设计用来结构化、存储以及传输信息。如图 1.5 所示,XML 不会做任何事情,它仅仅是包装在 XML 标签中的纯粹的信息。我们需要编写软件或程序,才能传送、接收和显示这个文档。

```
<note>
<to>Tove</to>
<from>Jani</from>
<heading>Reminder</heading>
<body>Don't forget me this weekend!</body>
</note>
```

图 1.5 以 XML 格式存储在 MongoDB 数据库中的数据

在一些应用框架中,用户对 XML 设置命名空间后,就可以使用相应框架命名空间中的标签来标记不同的内容。

JSON 与 XML 最大的不同在于,XML 是一个完整的标记语言,而 JSON 不是。这使得 XML 在程序判读上需要花费比 JSON 更多的工夫。XML 的设计理念与 JSON 不同,XML 利用标记语言的特性提供了绝佳的延展性(如 XPath),在数据存储、扩展及高级检索方面比 JSON 更具优势;而 JSON 则比 XML 更加小巧,且支持浏览器的内建快速解析,所以更适用于网络数据传输领域。

JSON 格式取代了 XML 给网络传输带来了很大的便利,但是却没有 XML 一目了然的特性,尤其是当 JSON 格式的数据很长的时候,会让人陷入烦琐复杂的数据节点查找中。

1.3 网页数据爬取

网页数据爬取是指从网页上提取特定内容,比如网页上的文字、图像、声音、视频和动画等。对于开发人员来说,拥有编程能力使得他们构建一个网页数据爬取程序变得非常容易并且有趣。但是对于大多数没有编程知识的人来说,最好使用一些网络爬虫软件从指定网页获取特定内容。本节将介绍八爪鱼采集器和 Web Scraper 数据爬取插件,并演示 Python 数据爬虫。

1.3.1 八爪鱼采集器

八爪鱼采集器不仅可以循环采集网页中的文本、图像和链接,还可以爬取网页中的动态信息。八爪鱼官网提供了丰富的数据采集教程,涵盖电商、新闻媒体、生活服务、公开企业信息等主流网站场景。

八爪鱼采集器的核心原理是:通过记录和模拟人的一系列上网行为,代替人眼浏览网页,代替人手工复制网页数据,从而实现从网页自动化采集数据。然后通过不断重复一系列设定的动作流程,实现全自动采集大量数据。

八爪鱼采集器在配置规则、采集数据的时候,主要会经过以下几个步骤:打开网页、单击元素、输入文本、提取数据、循环、下翻下拉列表、条件分支、鼠标悬停。针对这些步骤,八爪鱼采集器内置了很多高级选项。在对具体网页进行数据采集的过程中,网页结构、网页情况是不一样的,可以针对不同的网页进行不同的设置。

【例 1.1】 从豆瓣音乐排行榜爬取前 20 条数据。下面以豆瓣音乐排行榜为例来解析采集过程,需要采集歌曲的排名、歌名、播放量和作者的关注人数等四个部分的内容,具体采集流程如下。

(1) 下载安装八爪鱼采集器软件。可以直接在浏览器中搜索八爪鱼采集器,跳转到官网进行下载,该软件需要先进行注册才可以使用。

(2) 输入网址,设置采集元素,单击“开始采集”按钮,采集过程如图 1.6 所示。



图 1.6 “豆瓣音乐”采集图

(3) 采集并导出数据,如图 1.7 所示。



图 1.7 八爪鱼采集器成功导出数据界面

例 1.1 从豆瓣音乐排行榜爬取前 20 条数据的操作过程见二维码。

在使用八爪鱼采集器进行数据采集时,最核心的部分是采集器的设计,有时候可能需要用到网页的相关知识,这里不再介绍,如有需要进行更深一步的学习,可以参考八爪鱼采集器官网的技术手册。

1.3.2 Web Scraper 插件

Web Scraper 是一款操作简便的网页爬虫插件,支持 chrome 及所有 chromium 内核浏览器,适合普通用户迅速上手。可以方便地通过鼠标和简单配置获取数据,例如知乎回答列表、微博热门内容、微博评论、电商网站商品信息、博客文章列表等。Web Scraper 能够快速、有效、准确地提取网页数据,可任意选择爬取范围,同时还可以将数据转换为 CSV 文件输出。Web Scraper 插件需要安装 chrome 或火狐浏览器才能使用。

【例 1.2】 从知乎网站爬取回答者的昵称、回答内容、赞同数和评论数等信息,操作过程见二维码。使用 Web Scraper 插件爬取数据,需要先设置爬取的数据以及数据之间的关系,下面演示从知乎爬取数据的过程。

(1) 首先在 Chrome 浏览器中打开需要爬取的链接,按 F12 键调出开发者工具,单击 Web Scraper 标签栏。

(2) 转到 Create new sitemap 选项卡,填写 Sitemap name 和 Start URL,如图 1.8 所示。

(3) 添加选择器,单击 Add new selector 按钮。

(4) 分析知乎问题的结构。如图 1.9 所示,一个问题由多个区域组成,一个区域就是一个回答,这个回答区域包括了昵称、赞同数、回答内容和评论数等。方框框住的部分就是要爬取的内容。爬取数据的逻辑是:由入口页进入,获取当前页面已加载的回答,找到一个回答区域,提取里面的昵称、赞同数、回答内容、评论数等,之后依次向下执行,当已加载的区域获取完成后,模拟向下滚动鼠标,加载后续的部分,一直循环往复,直到全部加载完毕。

(5) 内容结构的拓扑图如图 1.10 所示, _root 根节点下包含若干个回答区域,每个区域下包含昵称 name、赞同数 zan、回答内容 cont。



V1.1 八爪鱼
采集豆瓣
音乐排行



V1.2 Web
Scraper 安装



V1.3 Web
Scraper 爬取
知乎数据

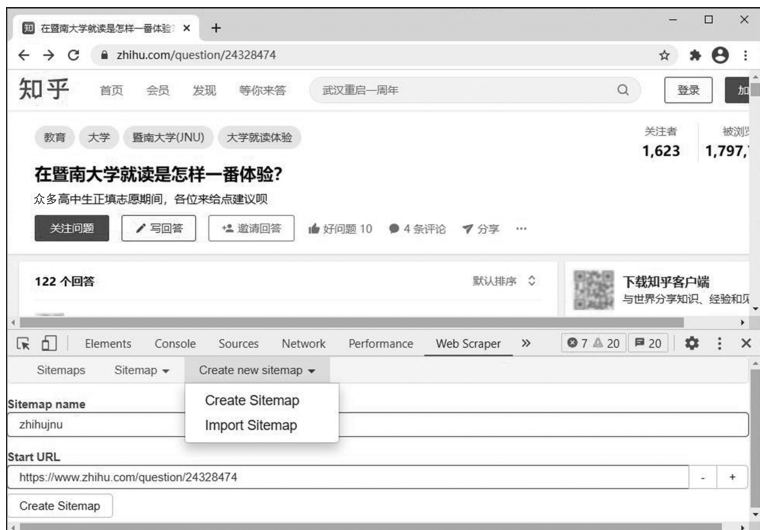


图 1.8 在知乎网页打开 Web Scraper 插件

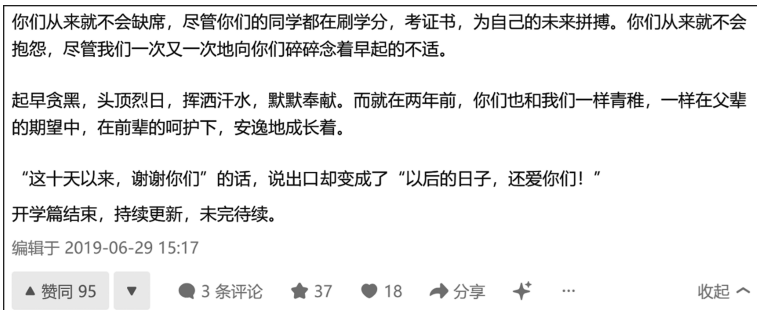


图 1.9 页面待爬取元素示例

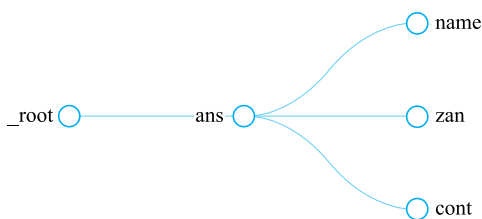


图 1.10 内容结构的拓扑图

(6) 创建选择器。首先填写 Selector ID 为 ans(名称可自定义), Type 类型选择 Element scroll down。Element 类型适用于选取包含子元素的大范围区域,本次需采集的回答区域匹配该类型,可从该区域中提取所需数据;而 Element scroll down 类型,可通过页面向下滚动的方式加载更多内容,专门适配下拉加载更多的场景。

(7) 完成多选。如图 1.11 所示,单击选择器 Selector 部分的 Select,在页面中通过绿色框依次选中两个回答区域并单击,其余回答区域将自动标红,如图 1.12 所示。完成后单击“Done selecting”按钮,选择 Multiple 并保存。

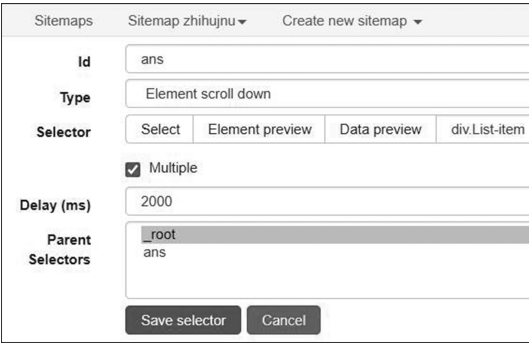


图 1.11 选择器属性设置

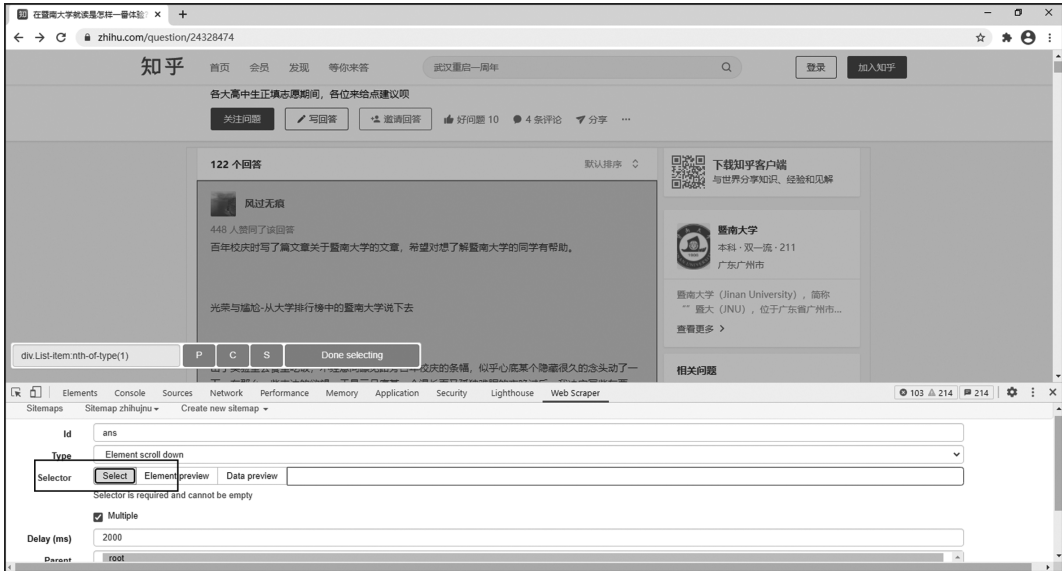


图 1.12 选中区域示例

(8) 接下来单击红色区域,进入刚刚创建的 ans 选择器中,精准定位需要爬取的元素作为子选择器,如图 1.13 所示。

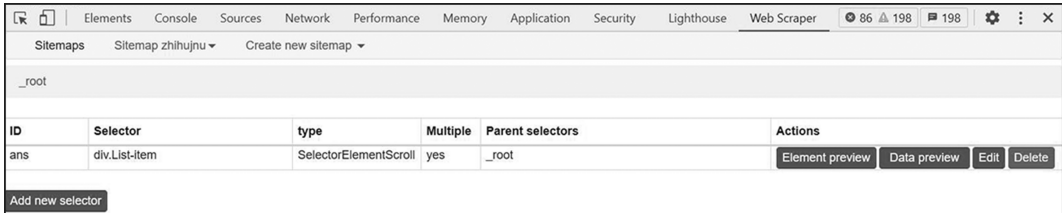


图 1.13 ans 选择器界面

(9) 创建昵称子选择器,设置 Id 为 name,Type 选择 Text,Selector 选择昵称部分,选中部分显示为红色,可以按 Shift 键选中多个,如图 1.14 所示。

(10) 创建赞同数选择器,参数设置如图 1.15 所示。



图 1.14 创建子选择器 name

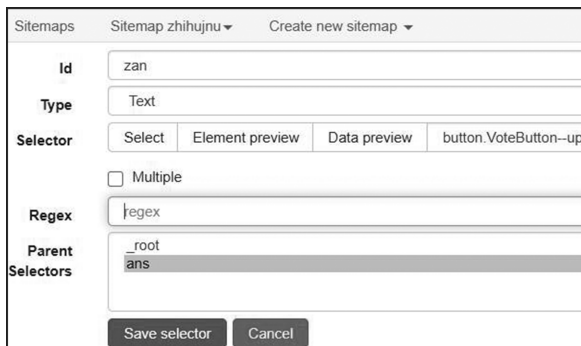


图 1.15 创建子选择器 zan

(11) 创建内容选择器,参数设置如图 1.16 所示,由于内容带有格式并且较长,单击文字边框处会比较容易选中所有内容。

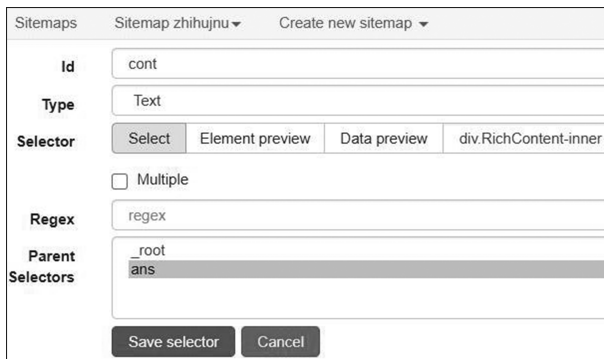


图 1.16 创建子选择器 cont

(12) 执行爬取操作,如图 1.17 所示,由于内容较多,可能需要几分钟的时间。如果仅是为了做测试,可以找一个回答数较少的问题进行。

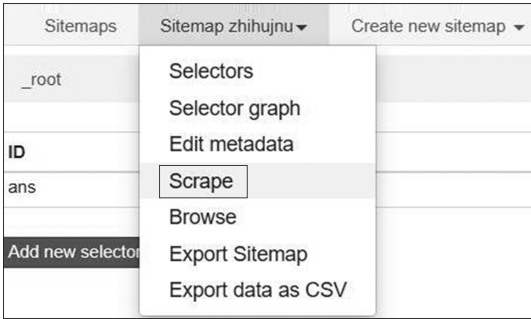


图 1.17 执行爬取操作

Web Scraper 插件的优点是可以爬取动态加载的数据,只要爬取频率慢一点,被网站屏蔽的概率就较小,爬取过程就像是真实的用户访问一样。Web Scraper 的缺点是爬取效率较低,无法实现并发爬取、快速切换 IP 等,所以 Web Scraper 适合轻量级的数据爬取。

【例 1.3】 从豆瓣音乐排行榜爬取前 20 条数据,包括排名次序、歌名和播放次数等数据,操作过程见二维码。



V1.4 Web Scraper爬取豆瓣音乐排行数据

1.3.3 Python 爬虫案例

Python 爬虫需要有一定的程序设计基础。本节仅演示 Python 爬虫功能的便捷、强大,并通过一个具体的例子来展示 Python 爬虫的高效率,爬虫代码详见程序 1.1。关于 Python 爬虫的详细介绍请参看第 8 章。

【例 1.4】 用 Python 爬取全国 10 个大城市近 10 年来每个月的房地产指数情况,总共爬取 1200 条数据。



V1.5 Python 爬虫演示

```
import requests
import csv
from lxml import etree
import json
first_row=['月份','城市','指数','环比(%)','指数','环比(%)','指数','环比(%)','指数','环比(%)']
with open('./XinFang_10yrs.csv', mode='w', encoding='mbcs', newline='') as file_obj:
    file_obj=csv.writer(file_obj)
    # 首行标签
    file_obj.writerow(first_row)
    for i in range(2020,2010,-1):
        years=str(i)
        for k in range(12, 0, -1):
            month=str(k).zfill(2)
            # 接口
            url='https://fdc.fang.com/index/XinFangIndex.aspx? action=month&month='
            '+years + '%25u5E74' +month + '%25u6708'
```

```
title=str(years)+'年'+str(month)+'月'  
# 请求页面的值  
response=requests.get(url)  
html=json.loads(response.text)['data']  
tree=etree.HTML(html)  
# 找到每个 tr  
tr_list=tree.xpath('//tr')  
for i in range(len(tr_list)):  
    td=tr_list[i].xpath('./td')  
    # 获取每个 td 的值  
    row=[title]  
    for j in range(len(td)):  
  
row.append(td[j].xpath('string(.)').strip())  
file_obj.writerow(row)
```

程序 1.1 近 10 年 10 个大城市房地产指数爬虫代码



1.4 数据清洗

数据清洗也叫数据预处理,目的是将重复、多余的数据筛选清除,将缺失的数据补充完整,将错误的数据纠正或删除,最后整理成为可以进一步加工、使用的数据。数据清洗的一般步骤是:分析数据、缺失值处理、异常值处理、去重处理、噪声数据处理。

本书涉及的数据处理技术主要针对较小量的数据,这里的数据清洗指的是手动的简单清洗处理,本节讲解如何使用 Excel 对数据进行整理。



V1.6 合并列

1.4.1 合并和拆分列

从外部数据源导入数据后的常见任务是将两列或多列合并为一列,或将一列拆分为两列或多列。例如,从网页上爬取数据时,可能多个字段作为一个数据项显示,此时就需要将其拆开;有时也可能需要将多列合并为一列,比如将名字和姓氏列合并为一个全名列。

在将两列数据合并时,可以使用 & 符号来实现字符串的连接操作,也可以使用 Ctrl+E 键进行快速填充合并。

【提醒】 选中要合并的单元格区域,单击“开始”选项卡下“对齐”组中的对话框启动器,弹出“设置单元格格式”对话框,在“对齐”选项卡下勾选“合并单元格”复选框,可以将两个单元格区域合并。如果使用该方法将多个单元格区域合并,只会保留最左侧的单元格数据。单元格区域合并通常用于数据格式化显示设置。



V1.7 拆分列

拆分数据列时可以使用函数或快捷键,也可以使用分列功能,分列功能比较简便,如果数据是用分隔符分隔开的,则可使用分隔符分成多列,也可以使用固定长度分列的方式实现分列。具体做法是:选中待分隔数据列,单击“数据”→“分列”按钮,弹出“文本分列向导”对话框,按提示步骤完成即可,如图 1.18 所示。

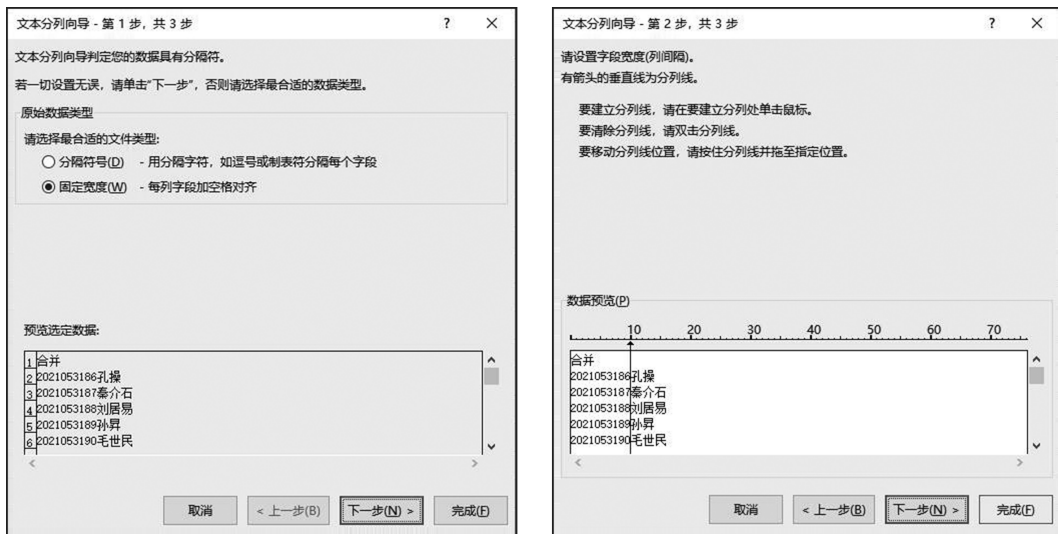


图 1.18 数据分列

1.4.2 修复日期和时间

由于存在许多不同的日期格式,并且这些日期格式可能混有编号部件代码或其他包含斜线标记或连字符的字符串,因此日期和时间通常需要进行转换并重新设置格式。

如果数据中的日期是字符串,比如“2021. 12. 31”,可以选中日期数据,按 Ctrl+H 键打开“替换”对话框,将“.”全部替换为“/”,即可将数据转换为日期格式。

以日期格式显示时,选中日期数据,右击,在弹出的快捷菜单中选择“设置单元格格式”命令,可在自定义选项中设置成“yyyy/mm/dd”格式,数据“2021/03/23”就会显示为“2021/03/23”;如果设置成“yyyy/m/d”格式,就会显示为“2021/3/23”。

1.4.3 删除重复值

导入数据时,重复行是较为常见的问题。建议先筛选唯一值,核对确认结果符合需求后,再执行删除重复值的操作,如图 1.19 所示。具体步骤:选中需处理重复项的列,单击“数据”→“重复项”→“删除重复项”命令;在弹出的选项中选择“扩展选定区域”,再次单击“删除重复项”命令完成确认;操作完成后,表格内的重复数据即可被清除。

1.4.4 查找、批量替换和删除

使用 Ctrl+F 键可打开查找窗口,用于快速定位目标文本;使用 Ctrl+H 键可打开替换窗口,实现批量文本替换或者删除。若需删除格式统一的前导字符串(如后跟冒号和空格的标签)或后缀说明等冗余内容,可在“查找内容”框中输入目标文本,“替换为”框保持为空,即可批量删除对应文本,完成数据规整。



图 1.19 删除重复值



V1.8 日期
格式设置



V1.9 删除
重复值

1.4.5 删除和填充空值

1. 批量删除空值

首先选中要编辑的数据区域,使用 Ctrl+G 键,在弹出的“定位”对话框中选择“空值”,单击“确定”按钮后,发现空值的部分都已经显示出来了。此时可以选择删除空行、空列或空单元格(为了保持数据的一致性,一般不建议只删除空单元格,容易导致数据错位移动,最好删除相应的行或列)。

2. 批量填充空值

选中空的单元格,使用 Ctrl+G 键,也可以通过直接选中的方式选择,然后输入要填充的数据,按 Ctrl+Enter 键批量填充,可以一次性填充所有单元格为相同的内容。

1.4.6 数据类型转换

正确判断 Excel 单元格的数据类型是非常重要的。很多 Excel 函数会对单元格的数据类型有适配要求。其中,数字与文本之间的转换最为常用。

1. 将数字存储为文本

默认情况下,我们在单元格中输入能够转为数字的字符串,Excel 兼容软件就会把它存为数字。

在数字前加半角的’(单引号),即可将数字转换成文本。在程序读取 Excel 中的数据时,不会把这个单引号读取出来。也可以使用数据“分列”的方法,把数字批量转换为文本。

2. 将文本存储为数字

如果是少量的单元格需要将文本转换为数字,可以直接删除数字前面的单引号。如果单元格较多,可以选中需要转换的单元格,然后单击  选择“转换为数字”选项。也可以使用“分列”的方法,把文本批量转换为数字。

【提醒】 如果单元格中既含有数字,也含有字符,则无法直接将其转换为数字,这时的单元格中也不会有字符标识的单引号。

【例 1.5】 使用 Excel 数据清洗方法清洗“豆瓣音乐排行”,该数据是使用八爪鱼采集器爬取的数据。

使用 Excel 进行数据清洗的步骤如下。

(1) 原始数据分析。

分析待处理的数据,保证这些数据的逻辑是一致的,并且数据类型符合处理的要求,比如日期类型的格式不能以字符的形式存在,需要进行数学计算的数据必须是数值类型,检查是否有多余的数据,检查空单元格、空行和空列的情况。

(2) 调整格式统一。

调整格式统一有助于发现数据中出现的一些问题,比如数据的缺失、错误等,必要时可借助 Word 或 TXT 文件进行文本处理。

【提醒】 从网络中爬取的数据通常以半结构化形式存在,可存储在 Excel 中,但是 Excel 无法显示特殊的字符,比如回车符、换行符等,可以充分使用 Word 中的字符串查找、替换操作进行特殊字符的处理,比如,替换段落标记为句号,如图 1.20 所示。



V1.10 批量
删除空值



V1.11 批量
填充空值



V1.12
数字转换
为文本



V1.13 文本
转换为数字



V1.14 豆瓣
音乐数据
清洗



图 1.20 替换段落标记为句号

这里将从八爪鱼采集器中提取的后 10 项的数据放入 Word 中进行文本处理,需要执行这些操作:删除段落标记、删除多余空格,以及将空格替换成斜线作为分隔符,以便于在 Excel 中进行提取填充。

(3) 填充提取。

使用 Ctrl+E 键进行数据提取,提取出歌名、作者、播放次数、关注人数等信息。

(4) 查找缺失值并继续爬取。

后 10 项数据缺少作者的关注人数,需要再次爬取。

1.5 练习题

1. 列举一些数据,将它们按照结构化、半结构化和非结构化进行分类,并简要说明分类依据。

2. 使用 Excel 创建一个学生成绩表,包含以下字段:姓名、学号、课程名称、成绩。完成以下操作:

(1) 为“成绩”列设置条件格式,低于 60 分标红。

(2) 按“课程名称”对数据进行排序。

(3) 统计每门课程的平均成绩。

3. 使用 SQLite 或 MySQL 创建一个名为 student_db 的数据库,并设计一个 students 表,包含以下字段: id(主键), name, age, major, score。插入至少 5 条数据,并用 SQL 语句查询所有 score 大于 80 的记录。

4. 将以下 JSON 数据手动转换为 XML 格式,并说明两者的主要结构差异:

```
{
  "student": {
    "name": "Alice",
    "age": 20,
    "major": "Computer Science"
  }
}
```

5. 使用八爪鱼采集器,采集某电商网站上“小米手环”相关商品名称和价格,将数据导出为 Excel 文件,并简要描述采集过程。
6. 使用 Web Scraper 插件,采集豆瓣电影 Top250 的电影名称和评分,导出数据并简要说明采集步骤。
7. 编写一个 Python 爬虫程序,爬取知乎某话题下的前 10 条热门回答,并将标题和点赞数保存到 CSV 文件中。
8. 将一系列字符串格式的日期(如“2025 年 12 月 1 日 12:30”)转换为标准时间格式,并计算每条记录到现在的时间差(天数)。
9. 使用合适的采集工具从京东商城采集 2025 年 8 月“AI 相关”书籍的销售排名,给出具体的名次和书本详细信息。
10. 对比整理 5 种以上数据采集工具,总结其优缺点,并给出具体的采集案例。