第3章

大数据系统输入

3.1 大数据采集过程及数据来源

数据采集又被称为数据获取，是从系统外部采集数据并将之输入系统内部接口的过程。随着互联网的快速发展，数据采集已经被广泛应用于互联网及分布式计算领域，例如，摄像头和麦克风等传感器都是数据采集工具。

3.1.1 大数据采集来源

根据应用系统可以将大数据采集的来源分为 4 种：管理信息系统、Web 信息系统、物理信息系统和科学实验系统。

1. 管理信息系统

管理信息系统指企业、机关内部的信息系统，如事务处理系统和办公自动化系统，这类系统主要用于经营和管理，为特定用户的工作和业务提供支持。管理信息系统提供的数据既可以来自终端用户的原始输入，也可以来自系统自身的二次加工处理的数据。这类系统的组织结构是专用的，数据通常是结构化的。

2. Web 信息系统

Web 信息系统包括互联网上的各种信息系统，如社交网站、社会媒体和搜索引擎等，主要用于构造虚拟的信息空间，为广大用户提供信息服务和社交服务。这类系统的组织结构是开放式的，大部分数据是半结构化或无结构化的。数据的产生者主要是在线用户。另外，电子商务和电子政务属于在 Web 上运行的管理信息系统。

3. 物理信息系统

物理信息系统指管理各种物理对象和物理过程的信息系统，如实时监控和实时检测，主要用于生产调度、过程控制、现场指挥和环境保护等。此类系统的组织结构是封闭的，数据由各种嵌入式传感设备产生，可以是关于物理、化学、生物等性质和状态的基本测量值，也可以是关于行为和状态的音频、视频等多媒体数据。

4. 科学实验系统

科学实验系统实际上也属于物理信息系统,但这类系统实验环境是预先设定的,主要用于学术研究,数据是有选择的、可控的,有时也可能是人工模拟生成的仿真数据。

从人-机-物三元世界观点看,管理信息系统和 Web 信息系统属于人与计算机的交互系统,物理信息系统属于物与计算机的交互系统。在人-机系统中,物理世界的原始数据是通过人实现融合处理的;而在物-机系统中,物理世界的原始数据需要通过计算机等装置进行专门处理。融合处理后的数据通常会被转换为规范的数据结构,输入并存储在专门的数据管理系统中(如文件或数据库),形成专门的数据集。

3.1.2 大数据采集过程

足够的数据量是大数据战略建设的基础,因此,数据采集就成了大数据分析的前站,是大数据价值挖掘重要的一环,其后的分析挖掘都建立在采集基础上的。

数据采集有基于物联网传感器的采集,也有基于网络信息的采集。例如,在智能交通中,数据采集有基于 GPS 的定位信息采集、基于交通摄像头的视频采集、基于交通卡口的图像采集和基于路口的线圈信号采集等。

互联网上的数据采集是对各类网络媒介,如针对搜索引擎、新闻网站、论坛、微博、博客和电商网站等的各种页面信息和用户访问信息进行的采集,采集的内容主要有文本信息、URL、访问日志、日期和图像等。之后需要清洗采集到的各类数据、进行过滤和去重等操作,并分类归纳存储。

数据采集过程涉及数据抽取(extract)、转换(transform)和加载(load)3 个子过程(即 ETL)。数据采集的 ETL 工具负责将分布的、异构数据源中的不同种类和结构的数据(如文本数据、关系数据,以及图像、视频等非结构化数据等)抽取到临时中间层后进行清洗、转换、分类和集成,最后加载到对应的数据存储系统(如数据仓库或数据集市)中,使之成为联机分析处理和数据挖掘的基础。

针对大数据的 ETL 处理过程同时又有别于传统的 ETL 处理过程,一方面,大数据的体量巨大;另一方面,数据的产生速度也非常快。例如,一个城市的视频监控头、智能电表每一秒都在产生大量的数据,对数据的预处理需要实时快速。因此,在选择 ETL 的架构和工具方面,需要采用如分布式内存数据库、实时流处理系统等现代信息技术。

现代社会各行业、部门中存在各种不同的应用、数据格式及存储需求,为实现跨行业、跨部门的数据整合(尤其是在智慧城市建设中,需要制定统一的数据标准、交换接口以及共享协议),这样不同行业、部门、格式的数据才能基于统一的基础实现访问、交换和共享。

3.2 大数据采集方法

数据采集系统整合了信号、传感器、激励器、信号调理、数据采集设备和应用软件。常用的数据采集方法归结为三类:传感器数据采集、系统日志数据采集和网络爬虫数据采集。

1. 传感器数据采集

传感器通常被用于测量物理变量，一般包括声音、温湿度、距离和电流等，将测量值转化为数字信号并传送到数据采集点，就可以让物体有触觉、味觉和嗅觉等感官，让物体慢慢变得活了起来。传感器数据采集如图 3.1 所示。

图 3.1　传感器数据采集

2. 系统日志数据采集

日志文件数据一般由数据源系统产生，用于记录数据源的各种操作行为，例如，网络监控的流量管理、金融应用的股票记账和 Web 服务器记录的用户访问等。

很多互联网企业都有自己的海量数据采集工具，这些工具多用于系统日志数据采集，均采用分布式架构，能满足每秒数百兆字节的日志数据采集和传输需求。系统日志采集方法如图 3.2 所示。

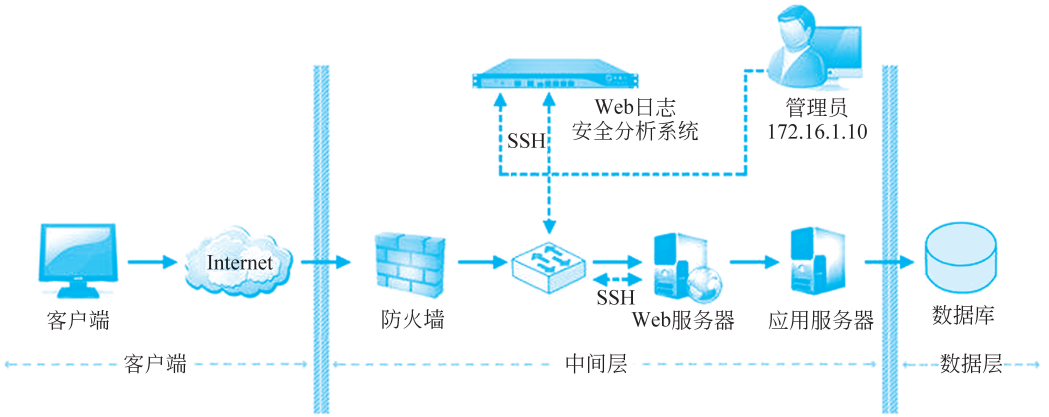

图 3.2　系统日志采集方法

3. 网络爬虫数据采集

网络爬虫指为搜索引擎下载并存储网页数据的程序，它是搜索引擎和 Web 缓存的主要数据采集方式。通过网络爬虫或网站公开 API 等方式可以从网站获取数据信息，该方法可以将非结构化数据从网页中抽取出来，将其存储为统一的本地数据文件，并以结构化的方式存储。它支持采集图像、音频、视频等文件或附件，并可将附件与正文自动关联。网络爬虫自动采集网页的过程如图 3.3 所示。

在采集企业生产经营数据（如客户数据、财务数据等保密性要求较高的数据）时，可以与数据技术服务商合作，使用特定系统接口等方式。

数据采集是挖掘数据价值的第一步，尤其当数据量越来越大时，可提取出来的有用信

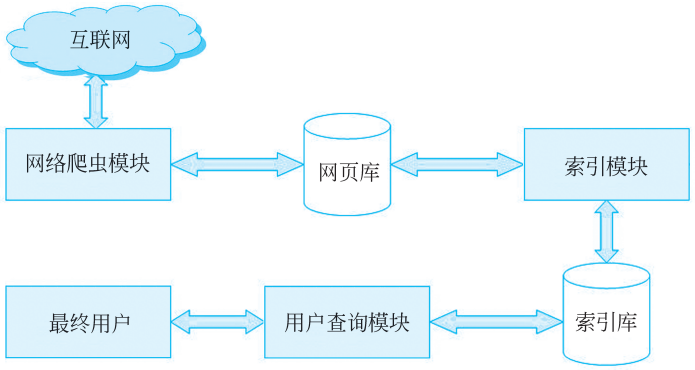

图 3.3 网络爬虫自动采集网页的过程

息必然会更多。只要善用大数据处理平台,就能够保证数据分析结果的有效性,助力企业实现数据驱动。

3.3 大数据导入/预处理

3.3.1 大数据导入/预处理的过程

大数据处理是将业务系统的数据抽取、清洗与转换,之后再将之加载到数据仓库的过程,目的是将企业中分散、零乱、标准不统一的数据整合到一起,为企业的决策提供分析依据。抽取、清洗与转换是大数据处理较重要的环节,通常情况下会花掉整个项目 1/3 的时间。

数据抽取是从各个不同的数据源抽取数据到处理系统中的过程,在抽取的过程中需要挑选不同的抽取方法,尽可能地提高运行效率。清洗、转换的过程通常耗时较长,一般情况下这部分的工作量花费的时间占该环节整个过程的 2/3。

1. 数据抽取

数据抽取需要在调研阶段做大量工作,首先要搞清楚以下几个问题:数据从几个业务系统中来?各个业务系统的数据库服务器运行什么数据库管理系统(database management system,DBMS)?是否存在手工数据?手工数据量有多大?是否存在非结构化数据?当采集完这些信息之后才可以设计数据抽取的方法。

(1) 与存放数据仓库(data warehouse,DW)相同的数据源处理方法。这一类数据源在设计时比较容易,一般情况下,数据库管理系统(包括 SQL Server、Oracle)都会提供数据库连接功能,在 DW 数据库服务器和原业务系统之间建立直接的连接关系的就可以使用 Select 语句直接访问。

(2) 与 DW 数据库系统不同的数据源的处理方法。针对这一类数据源一般情况下也可以通过开放数据库连接(open database connectivity,ODBC)的方式建立数据库连接,如 SQL Server 与 Oracle 之间的连接。如果不能建立数据库连接,那么可以用两种方式

完成：一种方式是通过工具将源数据导出成文本或者其他结构化数据格式的文件，然后再将这些源系统文件导入操作数据存储（operational data store，ODS）中；另一种方法是通过程序接口来完成。

（3）对于文件类型数据源（如 txt、xls 格式），可以利用数据库工具将这些数据导入指定的数据库，然后从指定的数据库抽取。

（4）增量更新问题。数据量大的系统必须考虑增量抽取功能。一般情况下，业务系统会记录业务发生的时间，该时间可以被用作增量的标志，每次抽取之前先判断 ODS 中记录最大的时间，然后根据这个时间去业务系统抽取大于这个时间的所有记录。

2. 数据清洗与转换

一般情况下，数据仓库分为操作数据存储和数据仓库两部分，通常的做法是从业务系统到 ODS 做清洗，将脏数据和不完整数据过滤掉，再从 ODS 到 DW 的过程中转换数据，进行一些业务规则的计算和聚合。

1）数据清洗

数据清洗的任务是过滤不符合要求的数据，将过滤的结果交给业务主管部门，确认是否需要将其过滤掉，并由业务部门修正之后再进行抽取。不符合要求的数据主要有不完整的数据、错误的数据和重复的数据三大类。

（1）不完整的数据。其特征是部分信息缺失，如供应商的名称、分公司的名称、客户的区域信息缺失，导致业务系统中主表与明细表不能匹配等。需要将这一类数据过滤出来，按缺失的内容分别写入不同数据文件并向客户提交，要求客户在规定的时间内补全，补全后再将之写入数据仓库。

（2）错误的数据。产生错误的原因往往是业务系统不够健全，在接收输入后没有对数据进行判断直接将之写入后台数据库造成的，例如，数值数据被输成全角数字字符、日期格式不正确、日期越界等。这一类数据也要分类处理，其中，包含全角字符、数据前后有不可见字符的问题只能靠写 SQL 查询语句的方式将之找出来，然后要求客户在业务系统修正之后抽取；类似日期格式不正确的或者日期越界的错误会导致 ETL 运行失败，这一类错误需要业务系统数据库用 SQL 查询的方式挑出来，交给业务主管部门要求限期修正，修正之后再抽取。

（3）重复的数据。需要将重复的数据记录的所有字段导出，让客户确认并整理。

数据清洗是一个反复的过程，不可能在几天内完成，只有不断地发现问题和解决问题才能逐步将之完善。过滤、修正等操作一般要求客户确认；过滤掉的数据可将之写入数据文件或者数据表，在 ETL 开发的初期可以每天向业务单位发送过滤数据的邮件，促使他们尽快地修正错误，同时也可以将其作为将来验证数据的依据。数据清洗时需要注意的是应避免将有用的数据过滤掉，每个过滤规则都应被认真验证，并要由用户确认。

2）数据转换

数据转换的任务主要是转换不一致的数据、数据的粒度和计算商务规则。

（1）转换不一致的数据。这是一个整合的过程，需要将不同业务系统的相同类型数据统一，例如，同一个供应商在结算系统的编码是 XX0001，而在 CRM 中的编码是

YY0001,在抽取之后应将其统一转换成一个编码。

(2) 转换数据的粒度。业务系统一般存储的是非常明细的数据,而数据仓库中的数据是用来分析的,不需要非常明细的数据。一般情况下,需要将业务系统数据按照数据仓库粒度进行聚合。

(3) 计算商务规则。不同的企业有不同的业务规则和数据指标,这些规则和指标有时不是简单地加减就能计算完成,可能需要在 ETL 中将这些数据指标计算好之后再存储到数据仓库中供分析使用。

3.3.2 数据清洗的过程

对于科研工作者、工程师、业务分析者这些和数据打交道的职业而言,数据分析是一项核心任务,其第一步就是清洗数据,例如,原始数据可能有如下不同的来源。

(1) Web 服务器的日志。

(2) 某种科学仪器的输出结果。

(3) 在线调查问卷的导出结果。

(4) 政府数据。

(5) 企业顾问准备的报告。

在理想世界中,所有记录都应该是整整齐齐的格式,并且遵循某种简洁的内在结构规律。但是实际并非如此,这些数据可能存在以下问题。

(1) 不完整的(某些记录的某些字段缺失)。

(2) 前后不一致(字段名和结构前后不一致)。

(3) 数据损坏(有些记录可能会因为种种原因被破坏)。

因此,必须经常使用清洗程序清洗这些原始数据,把它们转换成易于分析的格式,这种清洗又被称为数据整理(data wrangling)。

1. 识别不符合要求的数据

从名字上看,数据清洗就是把“脏”的数据“洗掉”。因为数据仓库中的数据是面向某一主题的数据的集合,这些数据可能是从多个业务系统中抽取而来,而且包含历史数据,因此难以避免错误数据的存在,有的数据相互之间还可能有冲突,这些错误的或有冲突的数据显然是人们不想要的“脏数据”。人们要按照一定的规则对这些“脏数据”进行处理,这就是数据清洗。数据清洗的任务是过滤那些不符合要求的数据,将过滤的结果交给业务主管部门确认是过滤掉,还是由业务单位修正之后再抽取。

2. 数据清洗的经验

数据清洗是一项需要逐步摸索、反复实践的工作,在设计数据清洗的规则时,设计者需要具备较为丰富的经验,也需要对数据本身有相当程度的了解。数据清洗的规则并非一蹴而就,其建立过程往往需要经历多次实验,需要设计者反复验证清洗结果后才能确定。

以下是一些清洗数据的经验。

(1) 不要默默地跳过记录。原始数据中有些记录是不完整的或者损坏的,所以清洗数据的程序只能将其跳过。默默地跳过这些记录不是最好的办法,因为很难保证数据不会被遗漏。因此,可以按照以下步骤处理。

① 打印出警告提示信息,这样就能够方便后续寻找错误源头。

② 记录总共跳过了多少记录,成功清洗了多少记录。这样做能够对原始数据的质量有大致的了解,例如,如果只跳过了 0.5%的数据,还属于正常范畴,但是如果跳过了 35%的数据,那么就该看看这些数据或者代码存在什么问题了。

(2) 把变量的类别以及类别出现的频次存储起来。数据中经常有些字段是枚举类型。例如,血型包括 A、B、AB 或者 O,但是如果某个特殊类别包含多种可能的值,尤其是出现始料未及的值,就不能及时确认了。这时,存储变量的类别以及类别出现的频次能够带来如下便利。

① 对于某个类别,假如碰到了始料未及的新取值时,就能够打印一条消息提醒设计者。

② 洗完数据之后为再次检查提供依据。例如,假如有人把血型误填成 C,再次检查时就能轻松发现了。

(3) 断点清洗。如果有大量的原始数据需要清洗,要一次清洗完就可能需要很长的时间,有可能是 5min、10min、1h,甚至几天。实际操作当中,经常在洗到一半时突然程序崩溃了。

假设有 100 万条记录,清洗程序在第 325 392 条处因为某些异常崩溃了,如果没有设计断点清洗机制,修改这个错误后重新清洗时程序就需要重新从 1 清洗到第 325 391 条,这是在做无用功。其实可以进行如下操作。

第一步,让清洗程序打印出来当前在清洗的记录号,如果程序发生崩溃,就能知道处理到哪条记录。

第二步,让程序支持断点清洗,这样当重新清洗时,就能从第 325 392 条直接开始。重新清洗的代码有可能会再次崩溃,只要再次修正错误,然后从再次崩溃的记录开始即可。

(4) 在一部分数据上进行测试。不要尝试一次性清洗所有数据。在刚开始写清洗代码和调试程序时,在一个规模较小的子集上先进行测试,然后扩大测试子集再测试。这样做的目的是让清洗程序能够很快地完成测试集上的清洗,节省反复测试的时间。

(5) 把清洗日志输出到文件中。在运行清洗程序时,把清洗日志和错误提示都输出到文件当中,这样就能轻松地使用文本编辑器来查看它们。

(6) 可选:把原始数据一并存储下来。在不用担心存储空间时这一条经验还是很有用的。这样做能够将原始数据保存在清洗后的数据中,清洗完后,如果发现哪条记录发生异常还能够直接看到原始数据,方便进行程序调试。

(7) 验证清洗后的数据。注意写一个验证程序验证清洗后得到的干净数据是否与预期的格式一致。人们虽然不能控制原始数据的格式,但是能够控制干净数据的格式。所以,一定要确保干净数据的格式符合预期的格式。

这一点其实非常重要,因为完成数据清洗后,这些干净数据将被用于下一步工作。因此,在开始数据分析之前要确保数据足够干净。否则,可能会得到错误的分析结果,到那

时就很难再发现很久之前的数据清洗过程中犯的错了。

3.3.3 数据清洗与数据采集技术

随着信息化进程的推进,人们对数据资源的整合需求越来越迫切。但整合分散在不同地区、种类繁多的异构数据库并非易事,要解决冗余、歧义等“脏数据”的清洗问题。仅靠手动清洗不但费时费力,而且质量也难以保证;另外,数据的定期更新也存在困难。所以整合业务系统数据是摆在大数据应用面前的难题。ETL 数据转换系统为数据整合提供了可靠的解决方案。

ETL 数据抽取、转换和加载过程负责将分布的、异构数据源中的数据(如关系数据、二维结构数据文件等)抽取到临时中间层后进行清洗、转换和集成,最后将其加载到数据仓库或数据集市中,成为联机分析处理、数据挖掘的基础。它可以批量完成数据抽取、清洗、转换和加载等任务,不但满足了人们整合种类繁多的异构数据库的需求,而且可以通过增量方式实现数据的后期更新。

主流 ETL 体系结构如图 3.4 所示。

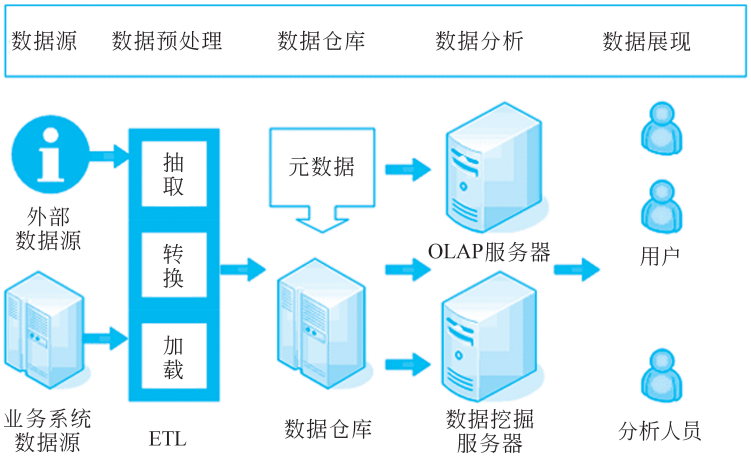

图 3.4 主流 ETL 体系结构

ETL 过程中的主要环节就是数据抽取、转换和加载。为了实现这些功能,各个 ETL 工具一般会进行一些功能上的扩充,例如,工作流、调度引擎、规则引擎、脚本支持和统计信息等。

数据抽取是从数据源中抽取数据的过程。实际应用中,不管数据源采用的是传统关系数据库还是新兴的 NoSQL 数据库,一般都有以下 3 种抽取方式。

1. 全量抽取

全量抽取是 ETL 在集成端初始化数据时,首先由业务人员或相关的操作人员定义抽取策略,选定抽取字段和定义规则后,由设计人员进行程序设计;处理数据后,直接读取整个工作表中的数据作为抽取的内容。与数据迁移类似,全量抽取是 ETL 过程中较简单的步骤,主要适用于处理一些对用户非常重要的数据表。

2. 增量抽取

增量抽取主要发生在全量抽取之后，用于抽取上次抽取过的数据源表中新增的或被修改的数据。增量抽取可以减少抽取过程中的数据量，提高抽取速度和效率，减少网络流量，同时，增量抽取的实现能准确把握异构数据源和数据库中数据的变化。

信息抽取不仅是从大量的文献集或数据集中找出满足用户需求的某篇文献或部分内容，而是抽取出真正满足用户需求的相关信息片段，并找出这些信息与原文献直接的参考对照。

3. 数据转换和加工

从数据源中抽取的数据不一定能完全满足目的库的要求，很容易出现数据格式不一致、数据输入错误和数据不完整等问题，因此人们还要对抽取出的数据进行数据转换和加工。

数据转换是真正将源数据库中的数据转换为目标数据的关键步骤，在这个过程中通过合并、汇总、过滤以及重新格式化和再计算等操作将操作型数据库中的异构数据转换成用户所需要的形式。对数据的转换和加工既可以在 ETL 引擎中进行，也可以在数据抽取过程中利用数据库的特性同时进行。

3.3.4　基于大数据的数据预处理

毫无疑问，数据预处理在整个数据挖掘流程中占有非常重要的地位，可以说大数据业务流程中 60％甚至更多的时间和资源都花费在数据预处理上了。

传统背景下数据预处理更多的是对数据库的清洗，这些数据有比较固定的模式，数据维度也不是很多，而且每一维度的数据类型（离散、连续数值和类标）以及包含的信息都很明确。

大数据背景下的数据预处理更倾向于对数据仓库的清洗。首先，数据都是异源（各种数据来源），异源数据的统一会带来极大的工作量；其次，数据可能没有固定的结构，或者为非结构化数据如文本；再次，数据量可能会很大，大到单机程序或小的分布式集群无法在给定时间范围内处理完毕；最后，数据量太大将导致很多有用的信息被噪声淹没，甚至都不知道这些数据的作用，分不清主次。

1. 为什么要预处理数据

（1）现实世界的数据是“脏”的（不完整，含噪声，不一致）。

（2）没有高质量的数据，就无法得到高质量的挖掘结果（高质量的决策必须依赖高质量的数据；数据仓库需要对高质量的数据进行一致的集成）。

（3）原始数据中存在的问题：不一致（数据结构不一致）、重复、不完整（缺少有价值的属性）、含噪声（数据中存在错误）、高维度或异常（偏离期望值）的数据。

2. 预处理数据的方法

(1) 数据清洗：去噪声和无关数据。

(2) 数据集成：将多个数据源中的数据结合起来存放到一个一致的数据存储容器中。

(3) 数据变换：把原始数据转换成适合数据挖掘的形式。

(4) 数据规约：主要方法包括数据立方体聚集、维度归约、数据压缩、数值归约、离散化和概念分层等。

3. 选取数据的参考原则

(1) 尽可能为属性名和属性值赋予明确的含义。

(2) 统一多数据源的属性编码。

(3) 去除唯一属性。

(4) 去除重复属性。

(5) 去除可忽略字段。

(6) 合理选择关联字段。

(7) 进一步处理：通过填补遗漏数据、消除异常数据、平滑噪声数据以及纠正不一致数据等操作去掉数据中的噪声、填充空值、丢失的值，并处理不一致的数据。

4. 数据预处理的知识要点

数据预处理相关的知识要点、能力要求和相关知识点如表 3.1 所示。

表 3.1 数据预处理的知识要点、能力要求和相关知识点

知 识 要 点	能 力 要 求	相关知识点
预处理数据的原因	(1) 了解原始数据存在的主要问题 (2) 明白数据预处理的作用和工作任务	(1) 数据的一致性问题 (2) 数据的噪声问题 (3) 原始数据的不完整和高维度问题
预处理数据的方法	(1) 掌握数据清洗的主要任务和常用方法 (2) 掌握数据集成的主要任务和常用方法 (3) 掌握数据变换的主要任务和常用方法 (4) 掌握数据规约的主要任务和常用方法	(1) 数据清洗 (2) 数据集成 (3) 数据变换 (4) 数据规约

5. 数据清洗的步骤

刚拿到的数据→与数据提供者讨论沟通→通过数据分析(借助可视化工具)发现“脏数据”→清洗“脏数据”(借助 MATLAB、Java 或 C++ 语言)→再次统计分析(可使用 Excel 的 Data Analysis 进行最大值、最小值、中位数、众数、平均值和方差等的分析，画出散点图)→再次发现“脏数据”或者与实验无关的数据(去除→最后实验分析→社会实例验证→结束)。

3.4　数 据 集 成

3.4.1　数据集成的概念

数据集成是指在不同应用系统、数据形式以及在原有应用系统未做任何改变的条件下进行数据采集、转换和存储的数据整合过程。数据集成领域已经有很多成熟的框架可以为人们利用。目前通常采用基于中间件模型和数据仓库等方法来构造集成的系统，这些技术在不同的着重点和应用上解决数据共享的问题并为企业提供决策支持。

数据集成的目的：运用技术手段将各个独立系统中的数据按一定规则组织成为一个整体，使其他系统或者用户能够有效地对数据进行访问。数据集成是现有企业应用集成解决方案中较普遍的一种形式。由于数据处于各种应用系统的中心，大部分的传统应用都是以数据驱动的方式进行开发。之所以进行数据集成，是因为数据分散在众多具有不同格式和接口的系统中，系统之间互不关联，所包含的不同内容之间也互不相通。因此，人们迫切地需要一种能够轻松地访问特定异构数据库数据的能力。

3.4.2　数据集成面临的问题

在信息系统建设过程中，由于受各子系统建设的具体业务要求、实施本业务管理系统的阶段性、技术性以及其他经济和人为因素等影响，导致在发展过程中积累了大量采用不同存储方式的业务数据。包括所采用的数据管理系统也大不相同，从简单的文件数据库到复杂的关系数据库，它们构成了企业的异构数据源，异构数据源集成是数据库领域的经典问题。

3.5　数 据 变 换

计算机网络的普及使数据资源的共享成为一个热门话题。然而，由于时间和空间上的差异，人们使用的数据源各不相同，各信息系统的数据类型、数据访问方式等也都千差万别。这就导致各数据源、系统之间不能高效地进行数据交换与共享，成为“信息孤岛”。

用户在具体应用时，往往又需要将分散的数据按某种需要进行交换，以便了解整体情况。例如，跨国公司的销售数据是分散存放在不同的子公司数据库中，为了解整个公司的销售情况，需要将所有子系统的数据集中起来。为了满足一些特定需要（如数据仓库和数据挖掘等）也需要将分散的数据交换集中起来，以达到数据的统一和标准化。异构数据的交换问题由此产生，并受到越来越多的重视。

用户在进行数据交换时面对的数据是千差万别的。产生这种数据差异的主要原因是数据结构和语义上的冲突。异构数据不仅指不同数据库系统之间的异构（如 Oracle 和 SQL Server 数据库），还包括不同结构数据之间的异构（如结构化的数据库数据和半结构化的数据）。源数据可以是关系型的，也可以是对象型的，更可以是 web 页面型和文本型

的。因此,要解决数据交换问题,一个重要的前提就是消除这种差异。随着数据的大量产生,数据之间的结构和语义冲突问题更加严重,有效地解决各种冲突问题是数据交换面临的一大挑战。

解决异构数据交换问题后,才需要对其他诸如联机分析处理、联机事务处理、数据仓库、数据挖掘和移动计算等提供数据基础。对一些应用,如数据仓库的建立,异构数据交换在这一过程中占据了十分重要的地位。数据交换质量的好坏直接影响交换后数据上其他应用能否有效执行。数据交换后,可以减小数据在存储位置上分布造成的数据存取开销;避免不同数据在结构和语义上的差异使数据转换出现错误;让数据存放更为精简有效,避免存取不需要的数据;向用户提供一个统一的数据界面等。因此,数据交换对信息化管理的发展意义重大。

3.5.1 异构数据分析

异构数据交换的目标在于实现不同数据之间信息资源、设备资源、人力资源的合并和共享。因此,分析异构数据、搞清楚异构数据的特点、把握异构数据交换过程中的核心问题是十分必要的,这样研究工作就可以做到有的放矢。

1. 异构数据

数据的异构性引发了应用对数据交换的需求。异构数据是一个含义丰富的概念,它指涉及同一类型但在处理方法上存在各种差异的数据。在内容上,异构数据不仅可以指不同的数据库系统之间的数据(如 Oracle 和 SQL Server 数据库中的数据);而且可以指不同结构的数据(如结构化的 SQL Server 数据库数据和半结构化的 XML 数据)。

总地来说,数据的异构性包括系统异构、数据模型异构和逻辑异构。

1) 系统异构

系统异构是指由于硬件平台、操作系统、并发控制、访问方式和通信能力等的不同而造成的异构,具体细分如下。

(1) 计算机体系结构的不同,即数据可以分别存在于大型机、小型机、工作站、PC 或单片机中。

(2) 操作系统的不同,即数据的操作系统可以是 Microsoft Windows、各种发行版的类 UNIX 等。

(3) 开发语言的不同,如 C、C++ 、Java 和 Python 等。

(4) 网络协议的不同,如 Ethernet、FDDI、ATM、TCP/IP 和 IPX\SPX 等。

2) 数据模型异构

数据模型异构指对数据库进行管理的系统软件本身的数据模型不同,例如,数据交换系统可以采用同为关系数据库系统的 Oracle、SQL Server 等作为数据模型,也可以采用不同类型的数据库系统——关系、层次、网络、面向对象或函数数据库等。

3) 逻辑异构

逻辑异构包括命名异构、值异构、语义异构和模式异构等。例如,语义的异构具体为用相同的数据形式表示不同的语义,或者同一语义由不同形式的数据表示。

以上情况构成了数据的异构性，数据的异构给行业单位和部门等的信息化管理以及决策分析带来了极大的不便。因此，异构数据交换是否迅速、快捷、可靠就成了制约行业、单位和部门信息化建设的瓶颈之一。

2. 冲突分类

在异构数据之间进行数据交换的过程中，实现严格的等价交换是比较困难的。主要原因是异构数据模型间可能存在结构和语义的各种冲突，这些冲突主要包括如下种类。

(1) 命名冲突。命名冲突指源模型中的标识符可能是目标模型的保留字，这时就需要重新命名。

(2) 格式冲突。同一种数据类型可能有不同的表示方法和语义差异，这时需要定义两种模型之间的变换函数。

(3) 结构冲突。如果两种数据库系统之间的数据定义模型不同，如分别为关系模型和层次模型，那么需要重新定义实体属性和联系，以防止属性或联系信息丢失。

由于目前主要研究的是关系数据模型间的数据交换问题，故根据解决问题的需要可将上述三大类冲突再次抽象划分为两大冲突：结构冲突和语义冲突。结构冲突指需要交换的源数据和目标数据之间在数据项构成的结构上存在差异。语义冲突指属性在数据类型、单位、长度和精度等方面的冲突。

3.5.2　异构数据交换策略

面向异构数据交换技术的研究始于 20 世纪 70 年代中期，经过多年探索，人们已取得了不少成果，并由相关研究学者提出了许多解决此问题的策略及方法，但究其本质可将其分成 4 类。

1. 使用软件工具进行转换

一般情况下，数据库管理系统都会提供将外部文件中的数据转移到当前数据库表中的数据装入工具。例如，Oracle 提供的将外部文本文件中的数据转移到 Oracle 数据库表的数据装入工具 SQL Loader，Powersoft 公司的 PowerBuilder 中提供的数据管道(data pipeline)。

这些数据转移工具可以多种灵活的方式实现数据转换，而且由于它们是数据库管理系统自带的工具，执行速度快，不需要开放数据库连接(open database connectivity，ODBC)软件支持，在计算机没有安装 ODBC 的情况下也可以被方便地使用。

但是，这些数据转换工具的缺点在于它们不是独立的软件产品，用户必须首先运行该数据库产品的前端程序才能运行相应的数据转换工具，这通常需要几步才能完成，且多用手工方式转换。如果目标数据库不是数据转换工具所对应类型的数据库，那么数据转换工具就无法发挥作用。

2. 利用中间数据库的转换

由于缺少工具软件的支持，在开发系统时可以中间数据库为桥梁，即在实现两个具体

数据库之间转换时,依据关系定义和字段定义从源数据库中读出数据并通过中间数据库灌入到目标数据库中。

这种利用中间数据库的转换办法所需转换模块少且扩展性强,但缺点是实现过程比较复杂,转换质量不高,转换过程长。

3. 设置传送变量的转换

借助数据库应用程序开发工具与数据库连接的强大功能,通过设置源数据库与目标数据库不同的传送变量同时连接两个数据库,实现异构数据库之间的直接转换。这种办法在现有的数据库系统下扩展比较容易,其可使转换速度和质量大大提高。

4. 通过开发数据库组件的转换

利用Java等数据库应用程序开发技术,通过源数据库与目标数据库组件来存取数据信息,实现异构数据库之间的直接转换。通过组件存取数据的关键是数据信息的类型,如果源数据库与目标数据库对应的数据类型不同,那么必须先进行类型的转换,然后双方才能实施赋值。

异构数据交换问题实质上就是:应用的数据可能要重新构造后才能和另一个应用的数据结构匹配,然后被写进另一个数据库。

3.5.3 异构数据交换技术

实现异构数据交换的方法和技术较多,本节列出XML、本体技术和Web Service等几项技术。

1. 基于XML的异构数据交换技术

1) 可扩展标记语言

可扩展标记语言(extensible markup language,XML)是一种用于标记电子文件使其具有结构性的标记语言。在计算机中,标记指计算机所能理解的信息符号,通过此种标记,计算机之间可以处理各种信息,如文章等。

XML提供了一种灵活的数据描述方式,其支持数据模式、数据内容和数据显示方式三者分离,可使同一数据内容在不同终端设备上的个性化数据表现形式成为可能,在数据描述方式上可以更加灵活。这使得XML可以为结构化数据、半结构化数据、关系数据库和对象数据库等多种数据源的数据内容加入标记,适于作为一种统一的数据描述工具,可扮演异构应用之间的数据交换载体或多源异构数据集成全局模式的角色。

2) 基于XML的异构数据交换的总体过程

由于系统的异构性,需要交换的数据可能具有多个数据源,不同数据源的数据模式也可能不同,这将导致源数据和目标数据在结构上存在差异。

在进行数据交换时,首先必须以统一的XML格式描述数据模型,通过文档类型定义(document type definition,DTD)这种特殊文档,从而将源数据库中的数据转换成结构化的XML文档,然后使用文档对象模型(document object model,DOM)技术解析XML文

档，这样就可以将 XML 文档中的数据存入目标数据库，实现异构数据的交换。

由于 DTD 文档定义的数据结构可以与源数据库中的数据结构保持一致，因此这样就保证了生成的 XML 文档与源数据库中数据的一致。

基于 XML 的异构数据交换的总体过程如图 3.5 所示。

图 3.5　基于 XML 的异构数据交换的总体过程

2. 本体技术

本体是对某一领域中的概念及其之间关系的显式描述，是语义网络的一项关键技术。本体技术能够明确表示数据的语义以及支持基于描述逻辑的自动推理，为解决语义异构性问题提供了新的思路，对异构数据集成来说有很大的意义。

本体技术也存在一定的问题：已有关于本体技术的研究都没有充分关注如何利用本体提高数据集成过程、提高系统维护的自动化程度、降低集成成本、简化人工工作。基于语义进行的自动集成尚处于探索阶段，故本体技术还没有真正发挥应有的作用。

3. Web Service 技术

Web Service 是近年来备受关注的一种分布式计算技术，它是在互联网(internet)或企业内部网(intranet)上使用标准 XML 和信息格式的全新技术架构。从用户角度上看，Web Service 就是一个应用程序，它只是向外界显现出一个可以调用的应用程序接口(application programming interface，API)。服务请求者能够用非常简便的、类似函数调用的方法通过 web 获得远程服务，服务请求者与服务提供者之间的通信遵循简单对象访问协议(simple object access protocol，SOAP)。

Web Service 体系结构由角色和操作组成。角色主要有服务提供者(service provider)、服务请求者(service requestor)和服务注册中心(service registry)。操作主要有发布(publish)、发现(find)、绑定(bind)、服务(service)和服务描述(service description)，Web Service 架构如图 3.6 所示。

其中，“发布”是为了让用户或其他服务知道某个 Web Service 的存在和相关信息，“发现”是为了找到合适的 Web Service，“绑定”则是在服务提供者与服务请求者之间建立某种联系。

在异构数据库集成系统中，用户可以利用 Web Service 具有的跨平台、完好封装及松散耦合等特性为每个数据源都创建一个 Web Service，使用 WSDL 向服务注册中心注册，然后集成系统就可以向服务注册中心发送查找请求并选择合适的数据源，以此通过 SOAP 从这些数据源获取数据。这样不仅有利于在数据集成中解决系统异构问题，同时也使数据源的添加和删除变得更加灵活，从而使系统具有松耦合、易于扩展的良好特性，能实现异构数据库的无缝集成。

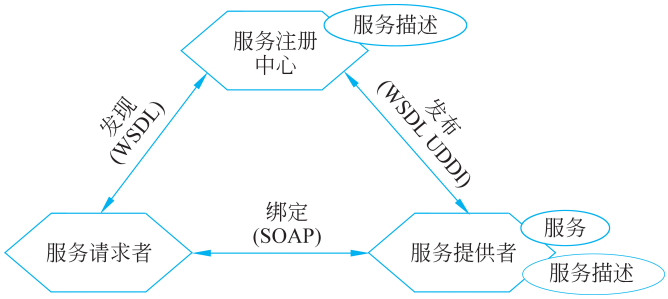

图 3.6 Web Service 架构

3.6 大数据应用案例：电影《爸爸去哪儿》大卖有前兆吗

2014 年 1 月 25 日,《爸爸去哪儿》(见图 3.7)首映发布会在北京举行。有媒体问郭涛：电影拍摄期只有 5 天,你怎么让观众相信,这是一部有品质的电影?

图 3.7 《爸爸去哪儿》

2014 年 1 月 27 日,某娱乐频道挂出头条策划——只拍了 5 天的《爸爸去哪儿》,值得走进电影院吗?

光线传媒总裁王长田解释：一般的电影只有 2～3 台摄影机,而《爸爸去哪儿》用了 30 多台摄影机,所以虽然只有 5 天的拍摄时间,却拍摄了 10 倍的素材量,在剪辑量上甚至比一般电影还大。

对这个话题的讨论看上去热度很高,但观众真在乎这件事情吗?《爸爸去哪儿》大卖,到底是让大家大跌眼镜的偶然事件还是早有前兆呢?

1. 真人秀电影 5 天拍完,观众真在乎吗

《爸爸去哪儿》是一部真人秀电影,制作流程在中国电影史上也没有可参照对象,但依然可能会有很多人贴标签——5 天拍完,粗制滥造。

本着危机公关心态出发,伯乐营销做了一次大数据挖掘,想看看到底有多少人在质疑《爸爸去哪儿》“圈钱”的事情,以及他们在提到这件事情时的看法,截至 2014 年 1 月 11 日,新浪微博仅有 536 人在讨论此话题(含转发人数),而其中还有近一半的人表示,“圈

钱”也要看，“圈钱”也无所谓的态度。比起对影片内容动辄数十万条的讨论和追捧，这个话题在所有相关话题中的讨论量简直是沧海一粟，比例关系如图 3.8 所示。

图 3.8　比例关系

反过来想一想，一部电影拍 5 年就值得观众进电影院了吗？商业电影环境里，让观众买单的是结果而不是努力的过程。

2. 原班人马出演，很重要吗

由热门电视剧改编的电影项目有很多，这里面有票房大卖的，如《武林外传》和《将爱情进行到底》，也有票房惨淡的，如《奋斗》《宫锁沉香》和《金太狼的幸福生活》，用比较粗暴的方式分类，前者基本是原班人马出演，而后者并非原班人马，虽然单看这几个项目就推出“热门电视剧改编电影＋原班人马＝票房大卖”未免太粗暴，但不得不说，原班人马对项目成功而言往往是加分因素。且通过大数据调查发现，对《爸爸去哪儿》大电影这个项目来说，原班人马更是至关重要。

在《爸爸去哪儿》大电影项目刚刚曝光时，还没有公布主演的两三天内，新浪微博上参与“原班人马”的讨论量便已经超过 2 万条。43.38％的网友表示，是原班人马就会看；47.06％的人表示，希望是原班人马出演；9.56％的网友表示，不是原班人马就不看。调查结果如图 3.9 所示。

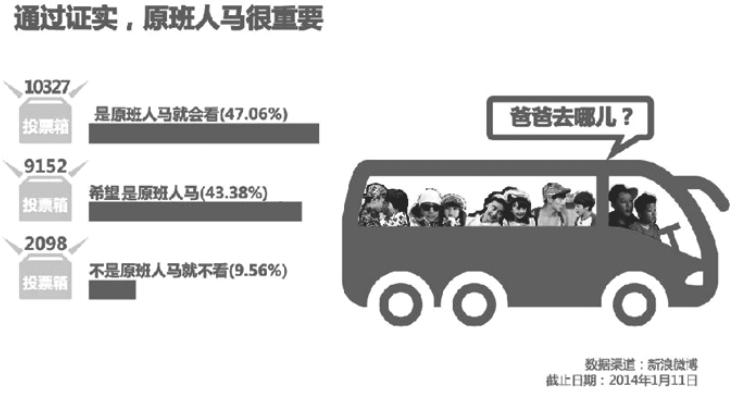

图 3.9　调查结果

与此同时,作者也调研了《武林外传》《将爱情进行到底》《宫锁沉香》和《奋斗》4 部电影在新浪微博上关于“原班人马”的讨论,虽然样本比《爸爸去哪儿》大电影小得多,但从分布比例上能明显看出,《奋斗》和《宫锁沉香》因没有使用原班人马而有失人心。成功案例和失败案例分别如图 3.10 和图 3.11 所示。

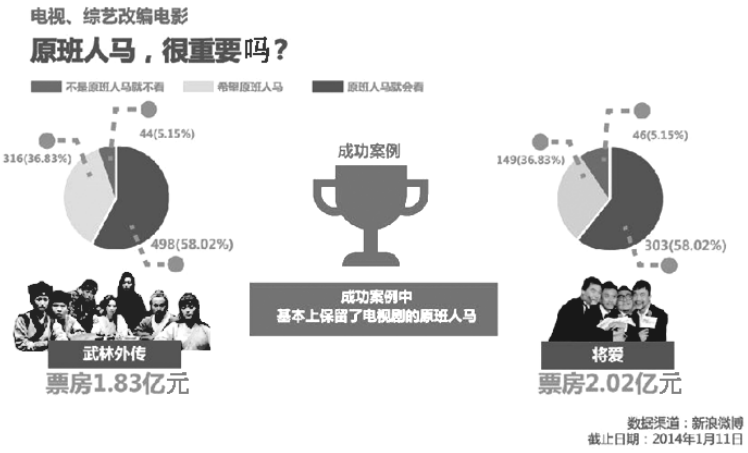

图 3.10 成功案例

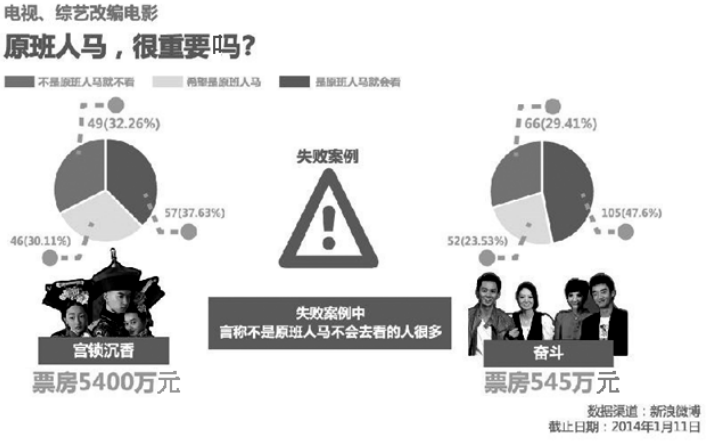

图 3.11 失败案例

3. 谁才是真正的“合家欢”

几乎每一部春节档公映的电影在宣传时都要给自己贴上“合家欢”的标签,那么,“合家欢”真的重要吗?

每年的春节档一般都是电影院的业务爆发期,而在这其中最受电影院欢迎的电影就是“合家欢”类型的电影,不难理解,适合全家一起看的电影能够带动更多消费,符合这样类型的电影必然会受电影院的欢迎。以春节档的《大闹天宫》《爸爸去哪儿》《澳门风云》作

为调研对象，在新浪微博上抽取以电影名和“全家”关键词作为讨论的条目，相较于《澳门风云》，《爸爸去哪儿》和《大闹天宫》有着绝对的优势，如图 3.12 所示。

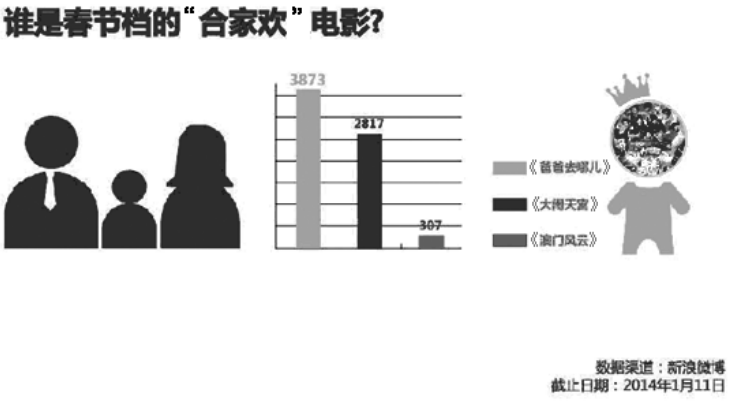

图 3.12　谁才是真正的“合家欢”电影

4. 有人注意到预告片播放量吗

数据公司随机抽取了在农历大年初一公映的 4 部电影自 2014 年 12 月以来发布的一款预告片和一款制作特辑，以腾讯、搜狐、新浪、优酷、土豆 5 家主流视频网站的播放量为调研对象进行分析，发现《爸爸去哪儿》和《大闹天宫》都是百万量级播放量，在 4 部电影中占尽优势。因为《大闹天宫》项目启动较早，考虑到预告片在各个平台上的长尾效应，此研究并没有选择《大闹天宫》的首场预告片，而是选择了与其他电影在密集宣传期相近时间主推的预告片。预告片播放量和花絮播放量如图 3.13 和图 3.14 所示。

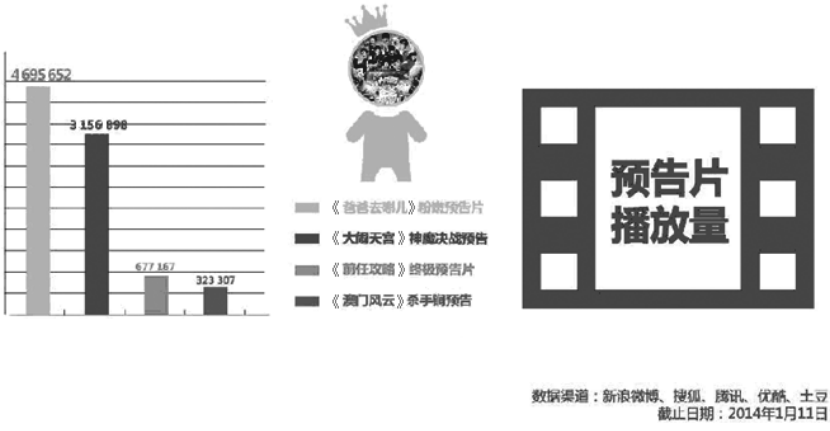

图 3.13　预告片播放量

图 3.14　花絮播放量

5. 春节档，人们到底最想看什么

其实论来论去，片方最关心的还是“想看”一部电影的人数，毕竟“想看”这个词直接和票房挂钩。但这个问题最复杂，因为与这个数据关联的维度非常多，有新浪微博上网友直接发出的声音，有业内营销人士非常关心的百度指数，也有像 QQ 电影票这样和用户购买行为直接相关的 App 服务商提供的数据，所以在这个问题上，笔者特地选择了几个不同的平台取样。

1）新浪微博

新浪微博数据图如图 3.15 所示。

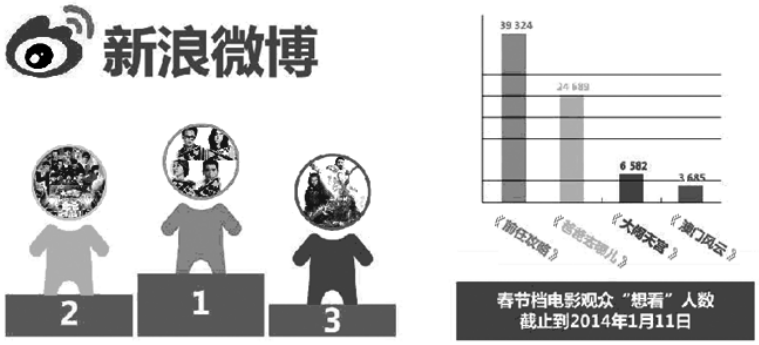

图 3.15　新浪微博数据图

从图 3.15 来看，在农历大年初一上映的 4 部电影中，提及“想看”和“期待”时，《前任攻略》的频率最高，《爸爸去哪儿》次之，可能这个结果与大家对未来各个电影在票房上的期待不符，但它的确反映了在微博这个阵地上粉丝为各个电影带来的话题讨论量。

2）百度指数

百度指数数据图如图 3.16 所示。