编 号

江南大学

本科生毕业设计（论文）

题目： 复杂网络技术在数据挖掘中的应用

物联网工程 学院 计算机科学与技术 专业

学　　号 1030414623

学生姓名 汪建海

指导教师 方伟 副教授

二〇一八年六月

摘　要

目前，数据挖掘和机器学习领域正在面临新的挑战，因为收集和需要处理的信息量很大. 许多复杂的学习方法不能简单地处理庞大而复杂的领域，因为执行时间难以管理，或者当领域变得更加复杂时发生的预测和通用性能力的丧失. 因此，为了应对当前现实世界问题的信息量，需要推进复杂数据挖掘技术的界限.

数据聚类是数据挖掘领域亟待研究重点与难点，对其研究在理论上和实际应用上都有重要的意义. 随着社交网络的崛起，比如新浪微博、知乎等网站蕴含着巨大的信息量，从大量的数据中挖掘有用的信息是一件极其不简单的事情. 近年来，很多领域都应用到了数据挖掘，比如金融的风险投资、广告精准投放、预测用户的购买行为等，数据挖掘为人们的生活提供了巨大的便利，也创造了巨大的经济和社会价值.

通过对复杂网络的研究，人们可以量化和预测模糊世界，而且能够在一定范围内预测事物的发展和运作，并能够预测网络崩溃. 目前有大量实际可用的模型，并且这些模型已经在实际生产和组织结构中的大量应用中使用，并取得了大量的实际成果.

本文对知乎用户资料进行数据挖掘，采用 Scrapy 框架对数据进行有效的数据采集，并应用知乎 API 采集特定的信息进行简单数据可视化. 通过复杂网络的的相关理论知识，结合特定应用领域的具体特性，对特定应用领域的复杂网络特性进行分析研究. 利用特定应用的数据相互关系，通过计算用户间关系相异度，并采用聚类算法和社团发现算法，对数据之间的关系进行分析，得出数据实体的关系聚类图，分析方法的实际效果.

关键词： 数据挖掘；数据聚类；复杂网络；聚类算法；网络模型

ABSTRACT

Currently, the data mining and machine learning fields are facing new challenges because of the amount of information that is collected and needs processing. Many sophisticated learning approaches cannot simply cope with large and complex domains, because of the unmanageable execution times or the loss of prediction and generality capacities that occurs when the domains become more complex. Therefore, to cope with the volumes of information of the current realworld problems there is a need to push forward the boundaries of sophisticated data mining techniques.

Data clustering is a key and difficult point in the field of data mining, and its research has important significance in the theory and practical application. With the rise of social network, websites such as Sina Weibo and Zhihu have a huge amount of information, and finding useful information from the amount of data is extremely difficult. In recent years, many fields have been applied to data mining, such as financial risk investment, accurate placement of advertisements, and prediction of users' puchase behavior. Data mining provides greate comvenience for people's lives and also creates greate economic and social value.

Through the research on complex network, people can quantify and predict the fuzzy world. At present, only the research results based on complex networks can predict the development and operation of things within a certain range, and can predict the network crashes.

At the same time, a large number of practically available models are prodeced during the process of research on complex networks, and these models have been used in a large number if application in actual production and organizational Structures, and a great deal of practical results have been achieved.

This paper conducts data mining for user data and uses Scrapy Framwork for effective data collection. It also uses ZHIHUAPI to collect specific information for simple data visualization. Through the relevant theoretical knowledge of complex networks, combined with the specific characteristics of specific application areas, the complex network characteristics of specific application areas are ananluzed and studied. Using the data relationshaip of specific applications, by caculating the dissimilarity between users, and using clustering algorithm and community discovery algorithm, the relationship between data is analyzed to obtain the relationship clustering diagram of data entities.

Keywords: Data mining; Data clustering; Complex network; Clustering Algorithm; Network model

目 录

第 1 章 绪论

1.1 选题背景与意义

计算机和信息技术的广泛使用导致了从广泛的应用领域创建大量数据库[1]. 如果适当的知识发现机制被用于提取嵌入数据中的隐藏的但潜在有用的信息[2]，那么这样庞大的数据库可以对未来的决策做出重大贡献.

DM 是提取数据丰富，不完整，模糊和随机的有用信息和知识的过程[3–5]. DM 被定义为大型复杂数据集的自动或半自动探索性数据分析，可用于揭示数据中的模式和关系，并强调大型观测数据库[6]. 现代统计和计算技术被应用于这个问题，以便找到隐藏在大型数据库中的有用模式[7,8]. 为了发现隐藏的趋势和模式，DM 使用了明确的知识库，复杂的分析技能和领域知识. 实际上，通过 DM 的趋势和模式形成的预测模型使分析师能够从现有数据中产生新的观察结果. DM 方法也可以被看作是统计计算，人工智能（AI）和数据库方法. 但是，这些方法并没有取代现有的传统统计数据; 实际上，它是传统技术的延伸. 例如，其技术已被应用于发现隐藏的信息并预测金融市场的未来趋势. DM 在商业和金融方面取得的竞争优势包括增加收入，降低成本，提高市场反应和意识[9]. 它也被用于推导新的信息，这些信息可以集成到决策支持，预测和估计中，以帮助企业获得竞争优势. 在高等教育机构中，DM 可以用于发现隐藏的趋势和模式，帮助他们预测学生的成绩.

所以说对挖掘技术的研究必然可以提高有用信息的获取，从而直接或间接推动这些应用的性能和实用性的提高，具有重要的研究意义和价值.

1.2 数据挖掘的发展与现状

1.2.1 数据挖掘的历史及发展

数据挖掘是一门有着悠久历史的学科，它从早期的数据挖掘方法贝叶斯定理（1700 年)[10] 和回归分析（1800 年)[11] 开始，这些分析主要是识别数据中的模式. 从下面的时间顺序表中简要地看到数据挖掘历史的主要里程碑:

数据挖掘是从不同角度分析大数据集（大数据）的过程，揭示相关性和模式以将其汇总为有用的信息. 现在它融合了许多技术，如人工智能，统计学，数据科学，数据库理论和机器学习等[12].

技术的日益强大和数据集的复杂性使得数据挖掘公司从静态数据交付演变为更加动态和主动的信息交付; 从磁带和磁盘到高级算法和海量数据库. 在 80 年代后期，统计学家，数据分析师和管理信息系统（MIS）社区开始了解和使用数据挖掘术语. 到了 20 世纪 90 年代初，数据挖掘被认为是一个子过程或者是一个称为数据库知识发现（KDD）的更大过程中的一个步骤，这实际上使得它成为“受欢迎的人”. KDD 最常用的定义是“识别数据中有效，新颖，潜在有用且最终可理解的模式的非平凡过程”（Fayyad，1996)[13]. 在过去的十年里，数据挖掘的普及率一直在迅速增长.

从 20 世纪 70 年代，数据挖掘技术主要经历了四个阶段，总结如图 (1-1) 所示:

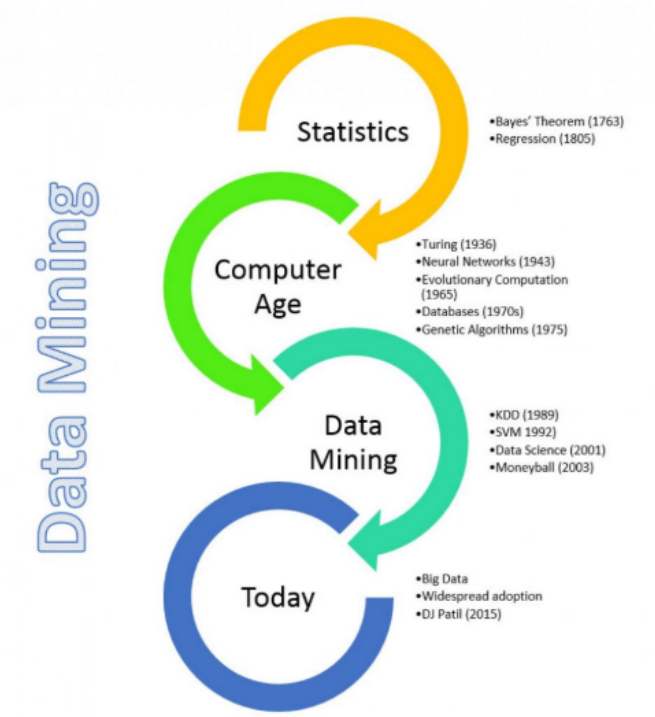


图 1-1 数据挖掘发展历程

发展阶段	商务模式	特征	计算类型	系统类型
20世纪70年代 第一阶段	电子邮件	作为一个独立的应用	单独设备	独立系统
20世纪90年代 第二阶段	信息发布	集成数据库 数据仓库	同质、局部区域的 计算机集群	数据管理系统 包含数据库
21世纪初 第三阶段	电子商务	集成预言与模型系统	Interanet/Extranet 网络计算	数据管理与 预言模型系统
现在 第四阶段	全程电子商务	集成移动数据与 各种计算设备的数据	移动与各种计算设备	数据管理预言 模型移动系统

图 1-2 数据挖掘发展

1.2.2 数据挖掘的研究现状

目前我国在数据挖掘技术研究上已经取得了巨大的成果，常用的数据挖掘模型包括神经网络模型、决策树模型、遗传算法模型、粗糙集模型、模糊集模型、关联规则模型等[14].

近年来，数据挖掘取得的了很多最新研究成果，本文选择了四篇研究大数据挖掘领域的论文，这些论文提供了数据挖掘领域的概述而且对未来该邻域进行广泛的概述.

(1) 由 Ryaboy 和 Dmitriy 写的名为“Scaling Big Data Mining Infrastructure: The Twitter Experience”[15] 的论文中介绍到: 有关大数据挖掘基础架构的见解，以及在 Twitter 上进行分析的经验. 它表明，由于数据挖掘工具的当前状态，执行分析并不简单. 大部分时间都用于数据挖掘方法应用的准备工作，并将初步模型转化为可靠的解决方案.

(2) 由 Yizhou Sun 和 Jiawei Han 写的名为“Mining Heterogeneous Information Networks: A Structural Analysis Approach”[16] 的论文中表明: 挖掘异构信息网络是大数据挖掘研究中一个新的有前景的研究前沿，它将互连的多类型数据（包括典型的关系数据库数据）视为异构信息网络. 这些半结构化的异构信息网络模型利用网络中类型化节点和链接的丰富语义，可以从互联数据中发现令人惊讶的丰富知识.

(3) 由 U Kang 和 Christos Faloutsos 写的名为“Big graph mining:algorithms and discoveries”[17] 的论文中介绍了挖掘大图的概述，重点介绍了 Pegasus 工具的使用，并展示了 Web Graph 和 Twitter 社交网络中的一些发现. 本文为大图挖掘提供了鼓舞人心的未来研究方向.

(4) 由 Xavier Amatriain 和 Christos Faloutsos 写的名为“Mining Large Streams of User Data for Personalized Recommendations”[18] 的论文中介绍了 Netflix 奖学习的一些经验教训，并讨论了 Netflix 中使用的推荐人和个性化技术. 它讨论了最近的重要问题和未来的研究方向. 文中第 4 节包含了一个有趣的讨论，关于我们是否需要更多数据或更好的模型来改进我们的学习方法.

有关大数据挖掘中的其他重要工作可以在主要会议上找到，如 KDD、ICDM、ECMLPKDD 或期刊例如“数据挖掘和知识发现”或“机器学习”等.

1.3 论文主要工作

本文围绕研究将数据挖掘中的聚类知识与复杂网络有关模型进行有机融合，提出了基于复杂网络的数据挖掘的应用，并就特定的用户数据进行研究，本文的主要工作有下面几个方面.

(1) 通过查找文献了解复杂网络和数据挖掘的定义和具体实现方法，对现有相关研究做一定的综述工作.

(2) 研究知乎用户数据并采用特定的方法对数据进行爬取，并对知乎用户数据进行简单的数据可视化.

(3) 通过研究特定的聚类算法并实现，分析实习的效果并分析他的优缺点，并应用算法对数据进行聚类，分析聚类效果.

(4) 用社团发现算法并结合复杂网络模型对数据进行分析，通过计算用户间关系相异度，分析数据以获得数据关系聚类图，分析方法的实际效果.

1.4 论文结构安排

本论文总共分为五个章节，每个章节的内容概括如下:

(1) 第一章为绪论，首先介绍了课题的研究背景，然后介绍了该数据挖掘领域中的历史发展和研究现状，并且简单描述了本文的研究内容，最后给出了全文的结构安排.

(2) 第二章对复杂网络的相关知识进行的说明，介绍复杂网络的网络特性和网络模型.

(3) 第三章介绍数据抓取的细节，并对数据做出一些清洗，为后文提供数据.

(4) 第四章为聚类算法的研究，详细说明了算法的实现的具体细节并分析实现效果，

说明了算法的优缺点，并用复杂网络技术对数据进行聚类分析.

(5) 第五章为结论与展望，对本文主要工作进行了总结，对存在的不足进行了展望.

第 2 章 复杂网络

我们在现实生活中经常发现你的朋友的朋友可能也是你的朋友，现实生活中很多网络表现除了很强的聚类性，尤其是社交网络. 社交网络是一组人或一群人，他们之间有一些互动模式或“关系”. 一群人之间的由友谊，公司之间的业务关系以及家庭的通婚都是过去研究过的网络例子. 所以本章节主要介绍有关复杂网络的基本理论和体现网络特征的参数，为后面章节打下了坚实的理论基础.

2.1 复杂网络的定义

复杂网络是由节点和节点之间许多复杂连接组成的网络结构，它是通过边连接的节点组成，节点可以任意分配一定的信息的网络系统的基本单元，边则是表示基本单元之间的关系或着相互作用.

但是，到目前为止，复杂网络还没有统一的定义，复杂网络一般具有以下特征: (1) 它是大量真是系统的拓扑抽象；(2) 它的统计特征介于规则网络和随机网络之间. 钱学森给[19] 出了复杂网络一个较严格的定义：

定义 2-1 具有自组织、自相似、吸引子、小世界、无标度中部分或全部性质的网络成为复杂网络 (Complex neworks).

复杂网络简而言之即呈现高度复杂性的网络. 其复杂性主要表现在一下几个方面：① 结构复杂：体现在巨大的节点数量，复杂的网络结构；② 网络进化：体现在节点或者节点之间的连接与消失；③ 连接多样性：节点之间的边的权重不一样，而且边具有不同的方向；④ 动力学复杂性：数据节点集属于非线性动力学系统；⑤ 节点多样性：复杂网络中的节点可以代表任何事物；⑥ 多重复杂性融合：就是以上几点之间的交互，将会导致更加不可预测的结果

在数学上，网络用图 (Graph) 来表示，所以不需要关心节点的位置，边的长度和形状，只需要关心节点之间是否相连接.

2.2 复杂网络的网络特征

2.2.1 网络概述

节点的度简单说就是与该节点相连边的个数.

在图 2-1 中，复杂网络中度为 0 的节点称为孤立节点. 如图 2-1 所示是一个无向图，节点 A 的度数为 2，节点 B 为一个孤立节点.

一个节点最简单最重要的局部特征就是节点的度，网络中所有节点度的平均值称为网络的平均度，用 P_k 来表示:

$$P_k = \frac{1}{N}\sum_{v\in V} d(v). \tag{2-1}$$

式中中 $\sum_{v\in V} d(v)$ 表示所有节点度数之和.

平均加权度是在统计节点度是，同时也考虑边的权重. 可以简单理解为在平均度的

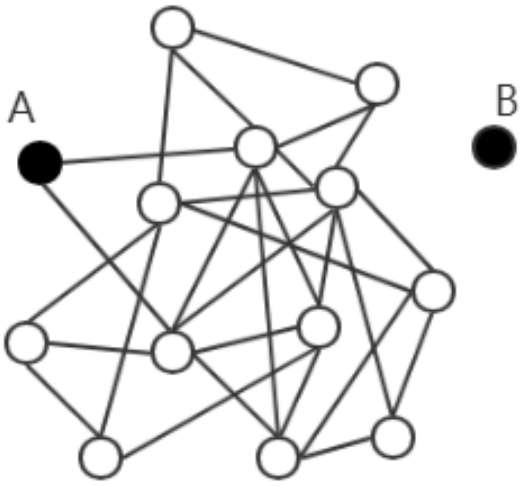


图 2-1 节点度的示意图

计算中，将边的权重全为当作 1 来计算，而在计算平均加权度的时候则是要根据实际的边的权重计算节点的度，并根据加权的度来计算平均的度.

网络中节点度值分布特征是网络的重要几何性质，度分布 (Degree distribution) 是用来体现网络中节点度分布状况的函数，一般用 $P(k)$ 表示.

在规则网络中，每个节点都有相同的度数，用 k_0 表示每个节点的度数，则其网络度分布服从 δ 分布，如图 2-2 所示：

$$P(k) = \delta_0(k) = \begin{cases} 1, & k = k_0, \\ 0, & k \neq k_0. \end{cases} \tag{2-2}$$

在随机网络[20] 中，度分布服从泊松分布，如下图 2-2 所示：

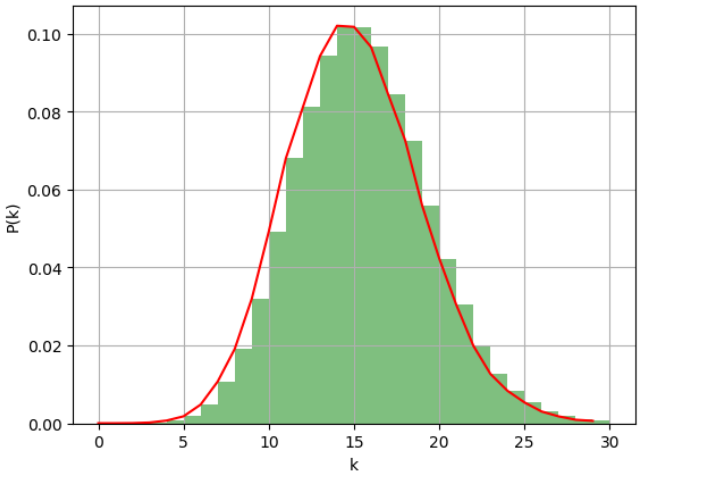
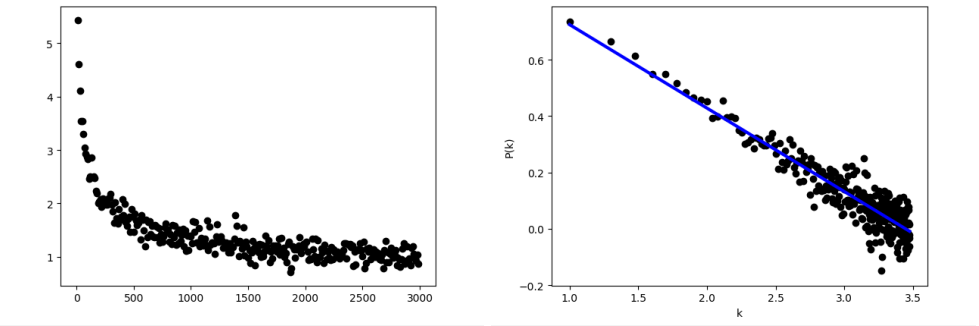


图 2-2 泊松分布

但是 Barabási 和 Albertlászló 等人[21] 人研究了一个包含 325739 节点的万维网子网，

他们通过实验发现发现万维网的度分布不像规则网络和小世界网络[22,23] 那样是对称的泊松分布，而是幂律分布 (Power law distribution). 如图 2-3 所示为幂律分布函数图，从这个图中可以很容易地看出，随着 k 的增加，曲线中没有出现峰值，而且还可以从图中看出较多的节点度数很小，极个别的节点度数很高（这些节点我们成为中枢点），并且网络中的节点具有很强的异质性.



(a) (b)

图 2-3 幂律分布

网络中的度分布[24,25] 是否遵循松分布或幂律分布关键取决于网络中的节点是反映同质还是异质. 与此同时还可以确认网络的性质，我们从数学关系上可以简单的知道指数函数是半对数坐标系中的一条直线，而幂律分布在双对数坐标系中是直线. 因此，为了确认度分布函数之间的确切关系只需要在半对数或者双对数坐标系绘制分布函数，即可简单的区分开它们.

网络直径是统计的边的连接特性，统计网络直径后，得出的值是一个网络整体的. 模块化是根据图的连接关系对节点做归类，类型相同的节点会增加一个字段，用相同的数字表示. 模块化在社会学中可以用于社区发现.

2.2.2 节点概述

聚类系数 (Clustering coeffocient)[26] 是指一个节点一度连接的节点中，实际的边数与最大边数之比. 聚类系数的定义如下：

$$C_i = \frac{d_1}{C_{k_i}^2} = \frac{2d_1}{k_i(k_i-1)} \tag{2-3}$$

式子中，d_i 表示与节点 A 相连的节点数目，$C_{k_i}^2$ 表示与 A 节点相连节点之间最大的连接数目.

网络的平均聚类系数 (图中所有节点聚类系数的平均值)：

$$C = \frac{1}{N}\sum_{i=1}^{N} C_i. \tag{2-4}$$

式子中，节点总数表示为 N.

计算过程如下：计算节点 a 的聚类系数，图中 a 节点度为 1 的节点有 b、c、d，这

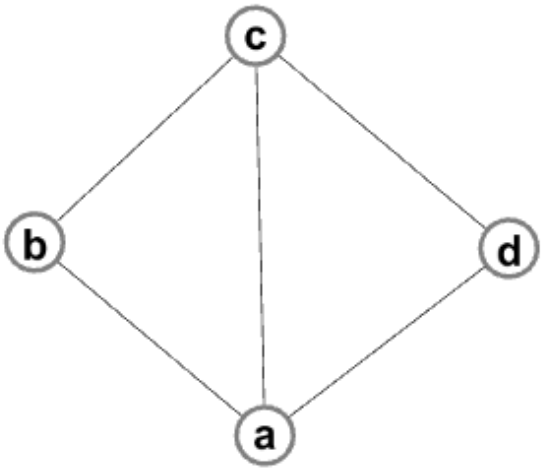


图 2-4 聚类系数

三条节点只有两条边 (b-c,c-d)，而这三个节点之间最多可以有 3 条边 (b-c,c-d,d-b). 因此 a 节点的聚类系数约为 0.67，同理可得节点 b、c、d 的聚类系数为 1.00，0.67，1.00，故该图的平均聚类系数为（1.00*2+0.67*2）/4=0.8333.

特征向量中心度[27] 也是能够表示节点重要性的一个参数. 核心节点不仅与大量的其他节点相连，而且也会与其他重要的核心节点相连，这是特征向量中心度的主要思想. 特征向量对于有向网络和无向网络都适用，在无向网络效果更好，但在有向网络中使用，否则讨论特征向量中心度是没有意义的.

2.2.3 边概述

描述边的特性一共有三个，分别是最短路径、网络直径、平均路径长度. 其中最短路径是指在一个网络中，两个节点连接最短的路径；最短路径的值就是最短路径中边的个数. 在复杂网络中，所有最短路径的最大值就是网络直径；所有网络最短路径之和的平均值等于这个网络的平均路径长度，这两个特性是整个网络的指标.

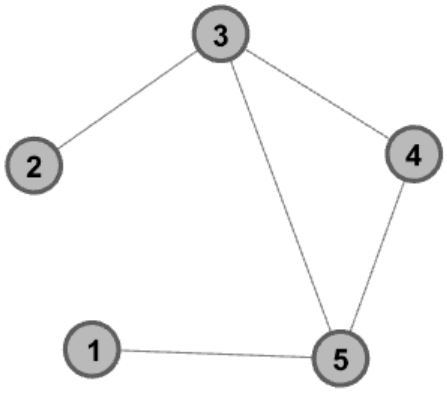
定义 2-2　平均最短路径长度：指一个网络中，节点的数量除以所有两个节点最短路径之和，记为 d_a.

$$d_a = \frac{1}{\frac{N(N-1)}{2}} \sum_{i \le N} \sum_{j>i} d_{(i,j)}. \tag{2-5}$$

其中 i，j 表示为网络中的两个节点，并定义节点 i 到自身的距离为 0.

这里以图 2-5(a) 中节点 2 到节点 1 的路径为例，在节点 2 与节点 1 之间共有两条路径可以连通，第一条是：$2 \to 3 \to 4 \to 5 \to 1$；第二条是：$2 \to 3 \to 5 \to 1$. 第二条路径短，所以最短路径就是第二条，数量为 3. 图 2-5(b) 中所示可以看出节点 1 到节点 2 的值最大（为 3），网络直径也就是 3. 由图 2-5(c) 和图 2-5(d) 可以得出最短路径长度为

16/10=1.6.



(a) 网络边

各节点间最短路径					
	节点 1	节点 2	节点 3	节点 4	节点 5
节点 1	-	3	2	2	1
节点 2	3	-	1	2	2
节点 3	2	1	-	1	1
节点 4	2	2	1	-	1
节点 5	1	2	1	1	-

(b) 最短路径

各节点间最短路径之和						
	节点 1	节点 2	节点 3	节点 4	节点 5	小计
节点 1	-	3	2	2	1	8
节点 2		-	1	2	2	5
节点 3			-	1	1	2
节点 4				-	1	1
节点 5					-	
网络中最短路径之和						16

(c) 最短路径之和

各节点间最短路径值的数量						
	节点 1	节点 2	节点 3	节点 4	节点 5	小计
节点 1	-	3	2	2	1	4
节点 2		-	1	2	2	3
节点 3			-	1	1	2
节点 4				-	1	1
节点 5					-	
网络中最短路径的数量之和						10

(d) 最短路径的数量

图 2-5 边特性

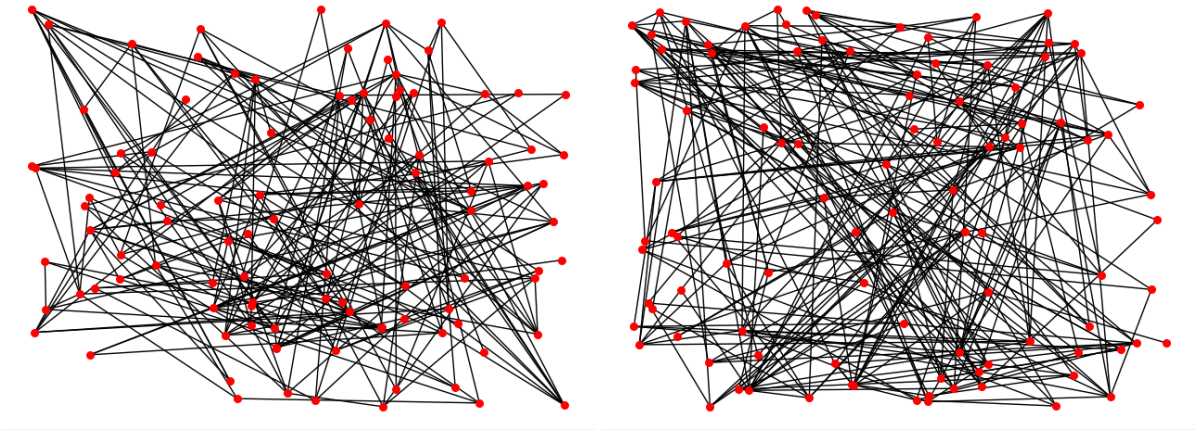
2.3 复杂网络模型

2.3.1 随机网络模型

20 世纪 60 年代，随机网络模型由埃德加 · 纳尔逊 · 吉尔伯特（Edgar Nelson Gilbert，1923-2013）[28] 独立介绍，同年匈牙利数学家 Erdos 和 Renyi 发表了他们的第一篇论文提出了 ER 随机网络模型[20]. 然而，Erdős 和 Rényi 的作品的影响如此之大以至于他们被正确地视为随机图论的创始人. 最简单的随机网络模型是 Erdös-Rényi 随机网络（ER 随机网络），其中所有边都是独立的.

对于 ER 随机网络有两种生成机制，一种是 G(N，L) 模型，N 个标记的节点与 L 个随机放置的链接链接. Erdős 和 Rényi 在他们的随机网络论文中使用了这个定义[20,29]. 如图 2-6(a) 所示. 另一种是 G(N,P), 对于节点总数为 N 的网络，任意两个节点之间的连接概率为 p. 这是 Gilbert 引入的一个模型[28]. 如图 2-6(b) 所示. 两种机制的关系是

$L = p \times \frac{N(N-1)}{2}$，$p = \frac{2L}{N(N-1)}$.



(a) N = 100，L = 200　　(b) N=100，P=0.05

图 2-6 随机网络模型

图 2-6(a) 中的，节点数为 100，边数为 200，两个节点连接概率为 0.0404；平均度和平均加权度都为 1.99（因为每条边的加权值都为 1，和平均度值一样），网络直径为 7，图密度为 0.04，平均聚类系数为 0.025，迭代 100 次的网络特征向量中心度为 0.01405，平均路径长度为 3.392. 图 2-6(b) 中，节点数为 100，两节点连接概率为 0.05，边数为 269；平均度和平均加权度为 2.69，网络直径为 6，图密度为 0.027，平均聚类系数为 0.047，迭代 100 次的网络特征向量中心度为 0.00553，平均路径长度为 2.84.

2.3.2 小世界网络模型

小世界现象，也被称为六度分离，长期以来一直令广大公众着迷. 它指出，如果你选择地球上任何地方的任何两个人，你会发现他们之间至多有六个熟人的路径. 居住在同一个城市的个人只有几次握手的事实并不令人意外. 小世界网络是许多现实世界网络的两个属性是任何节点对之间的距离相对较小，同时传递性或聚类的水平相对较高. 小世界的概念指出，即使是地球另一端的个人也可以通过一些熟人与我们联系.

在 1998 年， Duncan Watts 和 Steven Strogatz 提出了随机网络模型的扩展[30]：（1）小世界属性：在实际网络中，两个节点之间的平均距离取决于 N 的对数关系，而不是遵循正则格的期望多项式. （2）高聚类: 对于类似的 N 和 L 的随机网络，实际网络的平均聚类系数远高于预期.

Watt 和 Strogatz 开发了一个模型，将网格模型的传递性与随机网络模型的低路径长度相结合，创建了一个称为小世界网络的模型. 他们从具有高聚类和高平均路径长度的晶格模型开始. 然后，向模型中添加概率 p 边缘重新布线，意味着边缘与其中一个节点断开连接，然后随机连接到网络中任何位置的另一个节点. 选择每个边缘以概率独立重新布线 p.

如图 2-6 所示节点总数为 100，节点与邻近四个节点相连接，连接概率为 0.04 的 WS 模型. 平均度为 4，网络的聚类系数为 0.4127.

Watts-Strogatz 模型 (也称为小世界模型) 重新生成高聚类和小世界现象共存. 作为

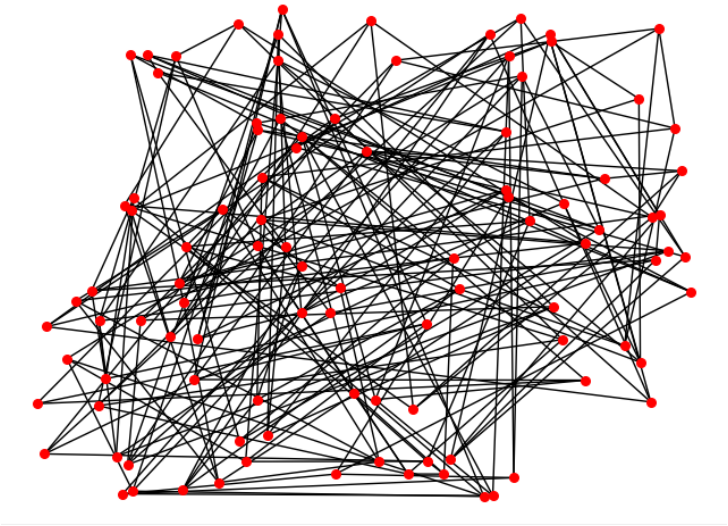


图 2-7 WS 小世界网络模型 (N=100，K=4，p=0.04)

随机网络模型的扩展，WattsStrogatz 模型预测了一个类似泊松的有界度分布. 理解高聚类的小世界属性的共存必须从网络的正确度分布开始.

2.3.3 BA 无标度网络模型

现实世界网络的一个共同特征是中枢节点或与网络中其他节点高度连接的少数节点. 中枢点的存在会给度分布一个长尾巴，表示中枢点的个数很少但度数却很高，而大部分节点度数少但个数且很多.

1999 年，当 Barabasi 和 Albert 在研究万维网的时候发现，万维网的都度分布不符合随机网络和 WS 模型，因而提出了 BA 网络模型[31]（也就是无标度网络模型）. 无标度网络模型符合幂率分布.

无标度网络模型 (scale-free model) 是一种以高度数中枢点为特征的网络模型，其度分布符合幂率分布，可以将其度分布写为:

$$P_d(k) \propto k^{-\gamma} \tag{2-6}$$

式中 γ 是指数，度分布 $P_d(k)$ 随着度数 k 的增减而衰减越来越缓慢，增加了找到非常大程度的节点的可能性.

图 2-8 说明了节点数目为 10000，幂律指数 $\gamma = 2$ 的无标度网络的度分布，其平均度大约为 7，但是 3/4 的节点的度数小于等于 3，在第一个条形图中，看不见有大于 100 度的节点，但用第二个中绘制条形高度会显示度分布的长尾. 尽管大多数节点的度数很少，但是有几个节点度数可以超过 500，这些中枢点的度数数量级比大多数节点的数量级要大，这就是幂律网络的一个特点. 我们可以清楚的看到，度分布绘制在双对数坐标上，散点图将倾向于沿着一条直线下落，这也是幂律分布的特点.

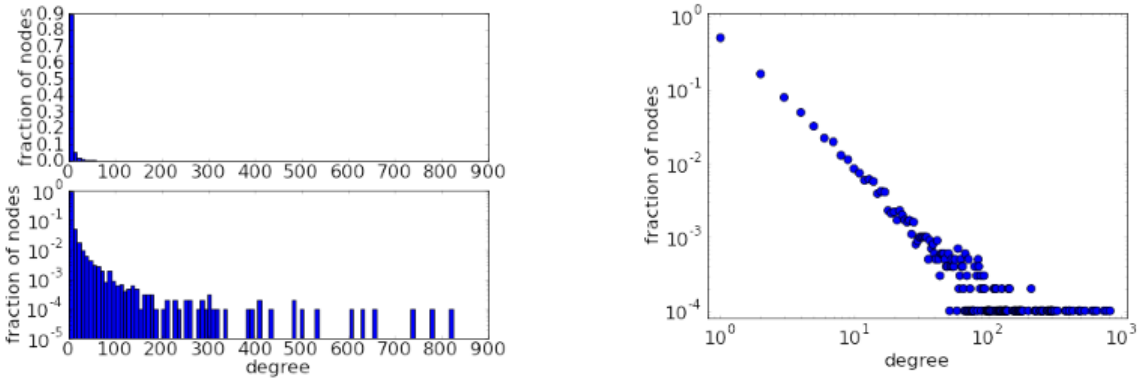

fraction of nodes
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
0 100 200 300 400 500 600 700 800 900
fraction of nodes
10^{0}
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
0 100 200 300 400 500 600 700 800 900
degree
fraction of nodes
10^{0}
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{0} 10^{1} 10^{2} 10^{3}
degree

图 2-8 BA 模型

第 3 章 用户信息采集

如果要对用户关系之间的网络进行分析，首先就要对用户信息进行采集. 本章节将介绍知乎用户的采集采集过程，以及采集特定数据的方法，并对数据进行简单的可视化工作.

3.1 网站选取

知乎是一个中国消费者的平台，其产品形态与 Quora(美国一家问答网站) 并无二致，知乎用户可以提出问题并可以从数据库中接受答案以及获取其他用户的答案。该平台从 2011 年 1 月 26 日创建，截至 2017 年 9 月份, 知乎官方宣布知乎注册用户超过 1 亿人次，其中每天知乎有 2600 万用户至少花费 1 个小时时间在知乎平台上浏览，月浏览量高达 180 亿次点击。知乎气氛浓厚，其内容得到广泛认可和信赖。基于用户、话题和其他元素，平台上的高质量内容可以快速传播到其他相关社区。“有问题请知乎”是总结知乎平台的最佳句子. 其凭借高质量的内容，知乎在互联网信息发布过程中的信息达到了上游位置。

知乎用户的主要特点是收入高、消费能力强、学历高等。其中本科及以上学历的用户占比 80.1%，中高收入和小康用户是知乎的主要人群，占比 76.0%. 这些知识渊博的人利用他们的批判性思维技能来分享他们在各个邻域的知识，从而创建一个具有社会认可度强大的“个性化品牌”。当人们越来越依赖专业人员的帮助和建议时，这些专业人员时创造热门话题和影响他人决策的关键。知乎注册用户破亿是知乎平台的一个里程碑，在这样的环境下知乎网站应该进一步从用户主体、用户体验、界面优化和使用规则等各个方面进行改善，进而构建一个全民的知识性的平台。

知乎尽管不进行新闻报道，但是知乎也可以成为关注的焦点，比如魏则西事件，雷洋事件、豫章事件等；发布在知乎上的内容，大家都可以分享，评论和发表个人的看法，能够推动事件的发展；另外知乎算法更加倾向于专家给出的答案，而且专业也会相互评论和相互交流看法，增加了网站的影响力和可信度。这说明知乎越来越具有影响力，在巨大的信息量面前必定隐藏了丰富的内容和有用的价值，因此对知乎的信息挖掘是有意义的。

3.2 实验环境

3.2.1 环境搭建

数据爬取的实验环境如下表 3-1 所示:

表 3–1 实验环境

类别	描述	类别	描述
开发语言	Python3.6	开发框架	Scrapy1.5
操作系统	Windows10 pro	系统环境	Anaconda4.5
CPU 型号	Intel i5-4210M	内存大小	4GB

3.2.2 框架介绍

本文采集知乎用户信息用的 *Scrapy* 框架，*Scrapy* 是一个用 *Python* 实现的开源并可以免费使用的跨平台的网络爬虫框架，它主要目的是为爬取网络提供支持；并支持 *JSON*、*CSV*、*XMl* 等多种格式输出；具有内置支持，通过 *XPATH* 或 *CSS* 表达提取源代码；支持一步并自动爬取数据；易扩展、爬取高效、程序健壮等优点

Scrapy 于 2008 年 6 月 2 日首次发布，截至 2018 年 3 月，最新版本为 *Scrapy* 为 1.5 版本且与 *Python*3.6 兼容. 目前有许多公司比如 *CareerBuilder*，*DayWatch*，*PriceWiki* 和 *Tarlabs* 等都在使用 *Scrapy* 作为爬取网站的框架.

Scrapy 是一个集成系统，包括一个控制所有组件之间数据流的引擎 (*ScrapyEngine*)，一个接收请求的调度程序 (*Scheduler*)，一个获取网页的下载程序 (*Downloadermiddlewares*)，一个 Spider 中间件 (*Spidermiddlewares*)，以及用户编写的用于解析响应 (*ItemPipeline*) 和提取项目的自定义类 (*Spiders*).

从网页上捕获和抓取数据的任务常常分为两个不同的阶段来执行：爬行和任务的抓取部分. 对于抓取知乎用户的详细信息，*Scrapy* 是一个不错的选择，因为它是一个基于 *Python* 的框架，提供了用于爬行和抓取数据的工具；此外，*Scrapy* 作为开源产品，拥有强大的社区并能够为用户提供有效的帮助. 接下来本文用图 3-1 展现 *Scrapy* 的架构，包括各种组件以及在系统中发生的数据流的展示 (绿箭头).

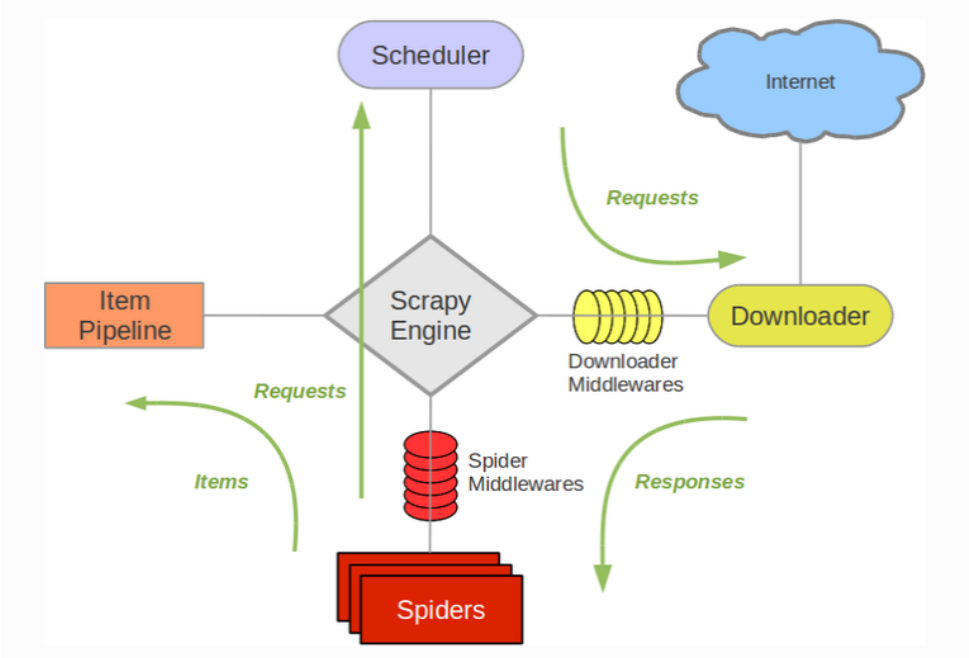


图 3-1 *Scrapy* 框架

了解更多详细信息请参考 **官方文档**

3.3 知乎用户详细资料抓取过程

3.3.1 采集流程

下面给出本文采集数据的思路，如图 3-2 所示：

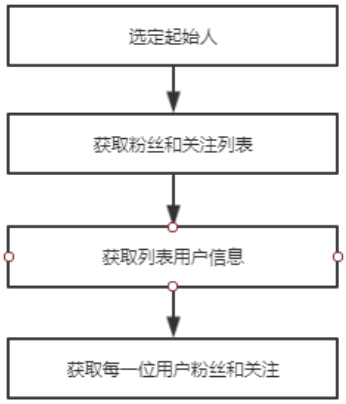


图 3-2 设计思路

选定初始人就是选定以为关注数或者粉丝数多的大 V 作为爬取起始点.(本文选取的是 *vczh*) 获取粉丝和关注列表就是通过知乎接口获得该大 V 的粉丝列表和关注列表. 获取列表用户信息就是通过知乎接口获得列表中每一位用户的详细信息. (获取用户详细信息的链接是获取 json 格式的资料，这样不用加载其他资源，访问速度快)，n 是数据集中包含数据点的个数，获取每一位用户粉丝和关注，进一步对列表中的每一个用户，获取他们的粉丝和关注列表，实现递归爬取.

下面列举一些采集过程中用的技巧，并列出本文在爬取时的参数 (在项目中 settings.py 中可以设置):

```
# 可以提高下载速度，建议夜里设置为False，可以减少服务器负担
ROBOTSTXT_OBEY = False
# 设置并发请求数量，默认16，为了提高CPU利用率，可以适当调节
CONCURRENT_REQUESTS = 32
# 设置网址禁止记录cookies可以提高CPU和内存利用率
COOKIES_ENABLED = False
# 设置日志级别，降低级别可以有利于提高性能
LOG_LEVEL = 'INFO'
# 设置禁止重试，当网站响应慢时可提高爬行效率
RETRY_ENABLED = False
# 设置下载超时，对于下载超时的网站，可以快速放弃该站点并释放资源
DOWNLOAD_TIMEOUT = 15
# 设置下载延迟，为了防止反爬虫，缓解服务器压力
DOWNLOAD_DELAY = 0.25
DOWNLOADER_MIDDLEWARES = {
  # 设置HTTP代理
  'scrapy.contrib.downloadermiddleware.httpproxy.HttpProxyMiddleware
     ':543,
  # 设置高度可逆代理，为了防止知乎对IP限流，采用的时动态IP代理
```

```
    'zhihuuser.middlewares.ProxyMiddleware': 125,
    # 设置禁止本地记录浏览器用户代理，可以随机使用下面的用户代理
    'scrapy.downloadermiddlewares.useragent.UserAgentMiddleware': None,
    # 设置用户代理，模仿浏览器访问网站，可以有效的反爬虫
    'zhihuuser.middlewares.ZhihuuserUser_agentMiddleware': 126,
}
```

为了提高爬虫速度还可以利用分布式爬取网站，只需要维护一个请求队列即可. 还有一些其他处理技巧，就不在一一列举具体的项目细节可以参考 **zhihuuser**

3.3.2 爬取用户关注列表

爬取用户的详细的关注列表或者时粉丝列表可以使用知乎 API 该框架的使用最重要的类就是 ZhihuClinet，想要获取知乎的数据就就必须先创建 ZhihuClinet 对象并登录。

```
# 创建了ZhihuClient对象
client = ZhihuClient()
# 如果有token.pkl用户登陆记录可直接加载登陆资料，
client.load_token('token.pkl')
# 如果没有请登陆并保存登陆记录，请使用下面方法:
try:
  client.login('email_or_phone', 'password')
except NeedCaptchaException:
  # 保存验证码并提示输入，并登录
  with open('a.gif', 'wb') as f:
      f.write(client.get_captcha())
  captcha = input('please input captcha:')
  client.login('email_or_phone', 'password', captcha)
# 上面可以简单一点写成:
client.login_in_terminal('email@example.com', 'password')
# 保存登陆记录
client.save_token('token.pkl')
```

要想获取特定用户的粉丝或者关注列表就必须确定该用户的 url_token，由于上节已经获取到了用户的详细资料中含有 url_token，本节就直接使用. 具体关键代码代码如下:

```
# 读取用户的id
for line in open('url_token.txt'):
  line = f.readline()
  result.append(line.strip('\n'))
f.close()
# 获取关注列表并写入列表
for p in result:
    try:
        people = client.people(p)
```

```
        f = open('following.txt', 'a')
        for following in people.followings:
            if following.over:
                continue
            try:
                f.write(people.name + ',' + following.name + '\n')
            except Exception as e:
                continue
        f.close()
    except Exception as e:
        continue
print("finish")
```

3.3.3 爬取数据及数据可视化

爬取的知乎信息字段如下表 3-2 所示：

表 3-2 数据字段

Field	类型	含义	Field	类型	含义
_id	Object	唯一标识用户	name	String	用户名称
gender	Int32	性别 (1 男,0 女)	deadline	String	用户一句话介
location	String	用户居住地	educations	String	教育经历
url_token	String	用户表示字段	answer_count	Int32	回答数目
thanked_count	Int32	获得感谢数目	question_count	Int32	提出问题数目
follower_count	Int32	粉丝数目	artticles_count	Int32	文章数目
following_count	Int32	关注数目	following_topic_count	Int32	关注话题数目
following_columns_count	Int32	关注专栏数目	following_question_count	Int32	关注问题数目

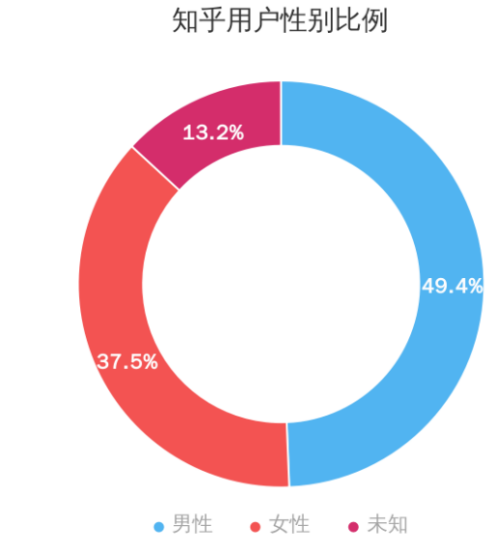
由于爬取的数据字段有 50 个，表 3-2 只列举常用字段.

本文爬取数据耗时 6 个小时共爬取数据 35825 条数据. 爬取用户粉丝列表如下格式：

表 3-3 数据字段

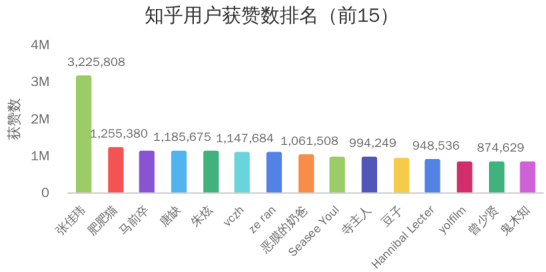
用户	关注用户	用户	关注用户
vczh	安然	TonyViceCity	vczh
vczh	米-格 MiGr	TonyViceCity	游公子
vczh	Roselyne LQ	TonyViceCity	丸子酱

对这些数据做一些简单的数据可视化如下图所示：

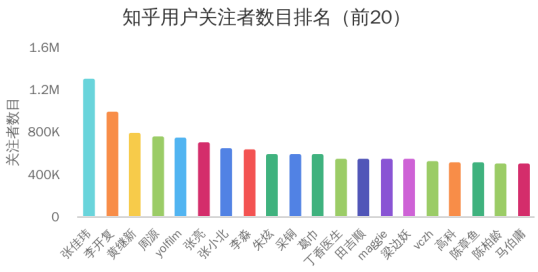


(a) 性别比例

(b) N 地区分布



(c) 赞数排名



(d) 关注数目排名

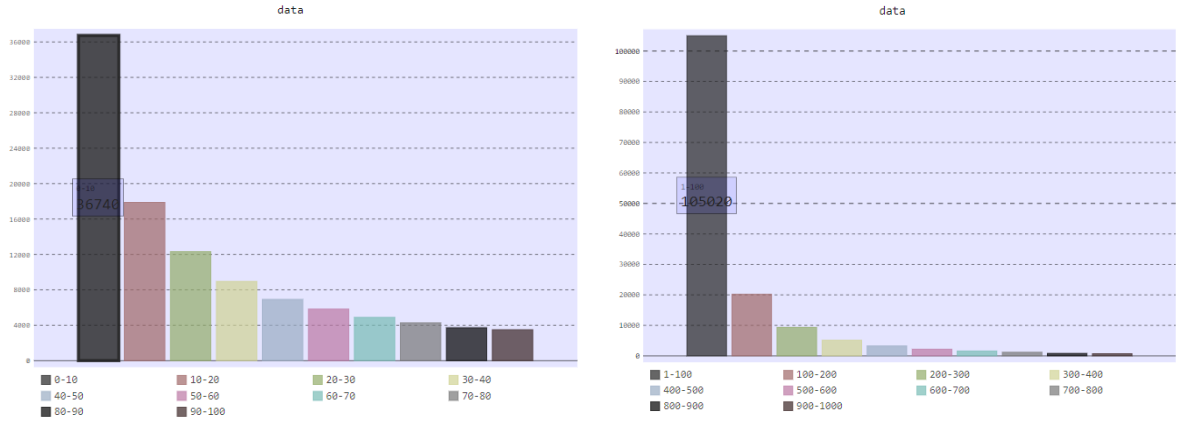
图 3-3 知乎用户数据可视化

第 4 章 知乎用户的复杂网络研究

本章节主要对知乎用户信息资料进行分析，通过采用聚类算法和复杂网络技术对数据之间的关系进行分析，得出数据实体的关系聚类图，分析算法的实际效果.

4.1 知乎用户资料的特征分析

对知乎用户资料的分析，为了判断该用户的重要程度，首先对用户的关注数、粉丝数 (关注者数)、回答数、赞同数、感激数的的平均数、中位数以及标准差进行了简单的统计，一共统计知乎用户数量 153913，统计结果如下所示：



(a) 关注数 100 以内的统计结果　　(b) 所有统计结果

图 4-1 following 统计结果

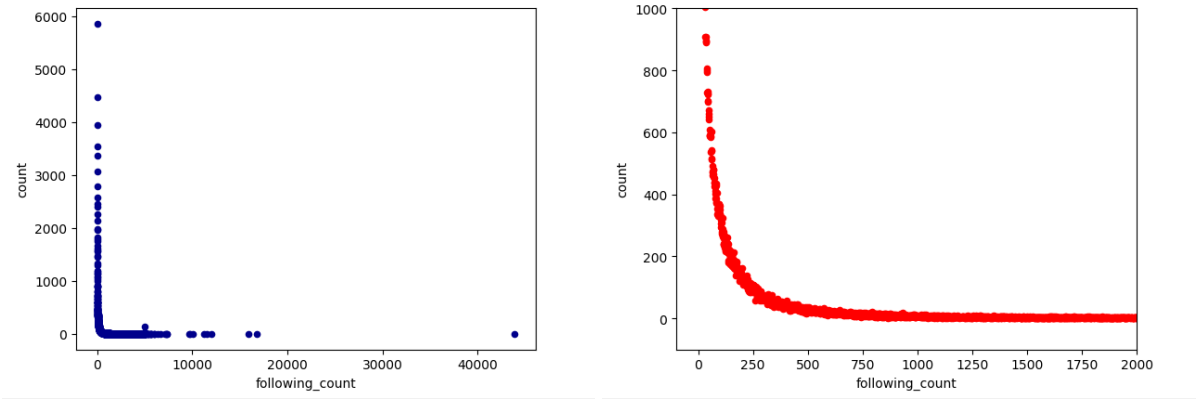
表 4-1 统计结果

type	mean	median	Std
following	167.9845	42	470.62
follower	1277.1270	9	14022.77
answer	38.2021	3	187.93
voteup	2992.0210	3	27399.50
thanked	513.0559	1	4438.83

在知乎用户之间一共存在三种关系，一种是用户甲关注用户乙，此时甲可以收到乙回答问题的信息推送和乙的动态，第二种关系是用户甲被用户乙，此时乙就变成甲的粉丝第三种是用户甲和用户乙相互关注，两人就变成了简单的好友关系，可以相互收到对方的信息. 根据图论知识我们可以简单知道，可以把用户当作节点，用户之间的关系可以简化为无权有向图.

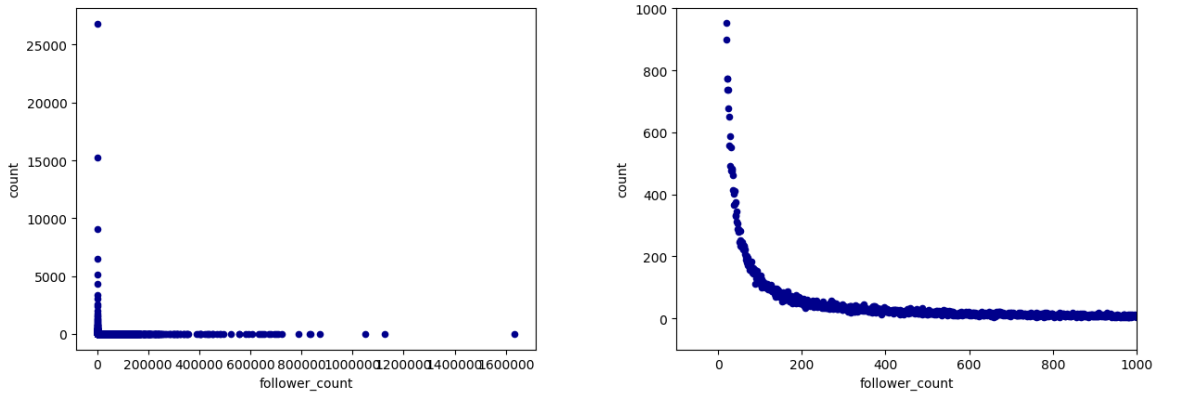
要对知乎用户信息进行复杂网络特征分析，就自然而然的想得到了第二章所介绍到的聚类系数和度分布，下图是知乎用户的入度和出度分布：

由于图 5-1(a) 用户数量太多，用户之间的差距太大，所以用户之间的关系不是很明显，因此把坐标的范围缩小到合适的大小后得到了 5-1(b) 的图像，入度分布处理和出度



(a) (b)

图 4-2 知乎用户出度分布



(a) (b)

图 4-3 知乎用户入度分布

分布处理一样.

根据图的分布我们很自然的想到了幂律分布，为了进一步得到明显的拟合曲线我们将幂律公式做以下处理：

根据幂律分布公式：

$$y = cx^{-r} \tag{4-1}$$

对式子两边同时取对数，可以得到：

$$lny = lnc - rlnx \tag{4-2}$$

在令两个 *lny* 和 *lnx* 分别为两个变量，在图形上很明显就可以看出变量是否符合幂律分布模型.

由此就可以轻易地看出，知乎用户网络的度分布符合幂律分布. 现在我们需要对数据进行线性回归分析，本文用到的工具是 python 的 Statsmodels 统计包. 本文重点介绍

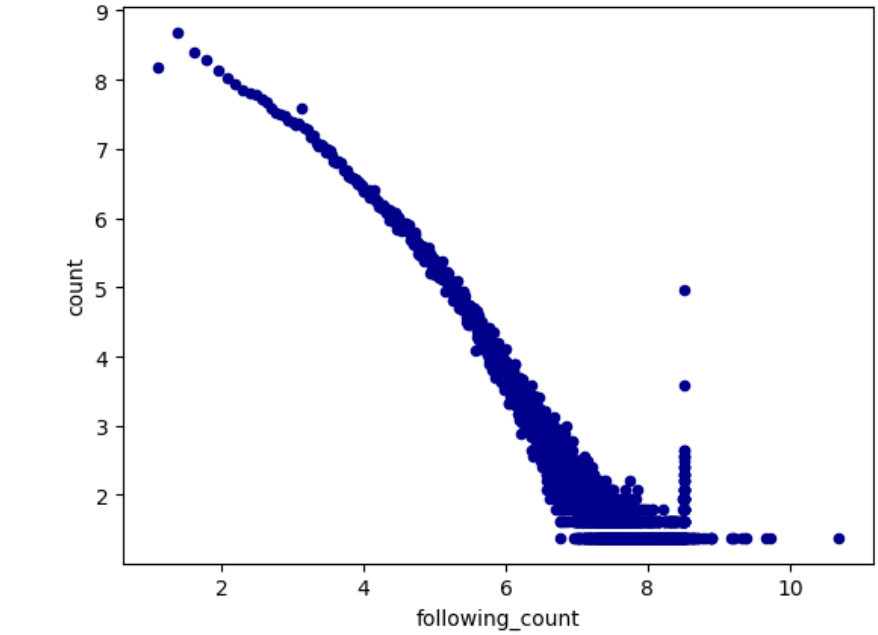


图 4-4 取对数度分布

回归信息中最常用的 OLS(ordinary least square) 功能.

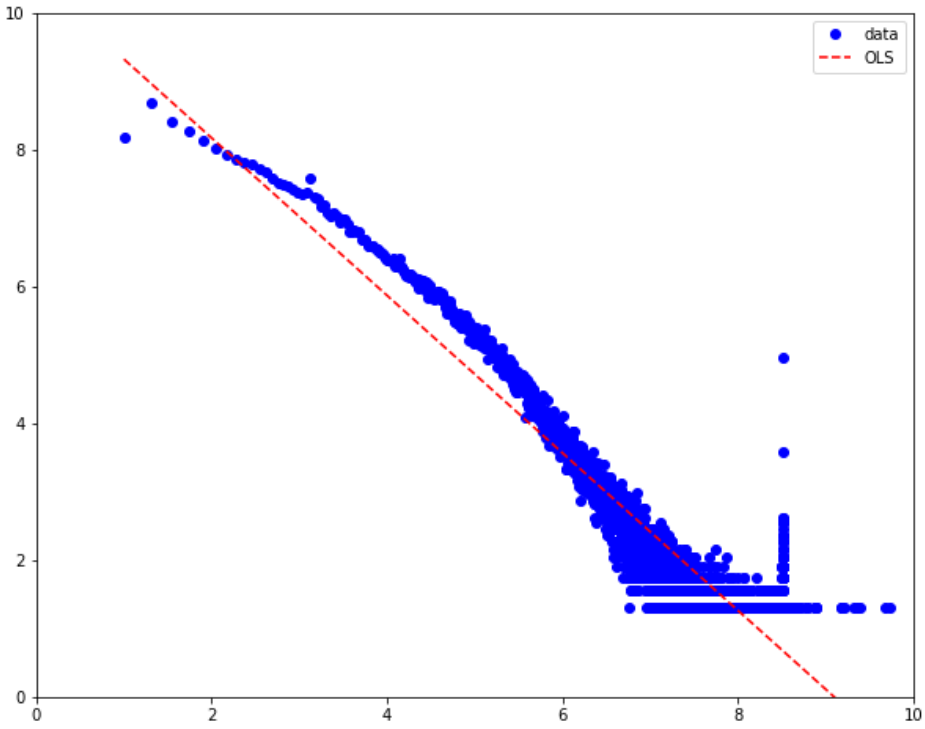


图 4-5 拟合曲线

针对知乎用户的出度分布，采用 OLS 方法对数据进行拟合得到结果如下表所示：

由表我们可以得到拟合曲线为 $lny = 8.804 - 2.7128lnx$，由此可以得到拟合的幂律公式为 $y = 6660.83 \times x^{-2.7128}$，拟合效果 (R-squared:) 为 0.982.

可以从图中看出，虽然统计时出度为 0 的数目为 3541，其中不乏有大量的僵尸粉

OLS Regression Results

Dep. Variable:	cnts	R-squared:	0.982
Model:	OLS	Adj. R-squared:	0.982
Method:	Least Squares	F-statistic:	1.080e+04
Date:	Sat, 12 May 2018	Prob (F-statistic):	2.26e-173
Time:	17:35:58	Log-Likelihood:	153.39
No. Observations:	198	AIC:	-302.8
Df Residuals:	196	BIC:	-296.2
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	10.3852	0.043	239.549	0.000	10.300	10.471
key	-1.0085	0.010	-103.930	0.000	-1.028	-0.989

Omnibus:	25.195	Durbin-Watson:	0.626
Prob(Omnibus):	0.000	Jarque-Bera (JB):	33.398
Skew:	-0.801	Prob(JB):	5.59e-08
Kurtosis:	4.218	Cond. No.	25.5

图 4-6 拟合效果分析

对拟合效果造成了一定的影响，但是总体上知乎用户关系之间的网络还是具有复杂网络具有的特征，符合 BA 无标度网络模型.

4.2 K-means 算法

4.2.1 算法介绍

K-means 是典型的基于划分思想的聚类. 那么什么是聚类呢？《周易 · 系辞上》说:“方以类聚，物以群分，吉凶生矣.”聚类就是是把数据对象的集合划分成不同簇，使簇内对象批次相似，簇间对象不相似的过程，是大数据分析的基本工具.

通过查阅文件和上网查找资料，总结了聚类的发展阶段如下图所示:

图 4-7 聚类发展

4.2.2 算法思想和算法流程

K-meas 的算法[32] 思想是基于划分的聚类方法，将数据集按照算法进行划分若干组，采用贪心迭代求解. 具体做法如下：先选定分成的多少簇，根据目标函数对数据集进行划分，不满足目标函数时，继续重新选择代表点重复上述构成直到收敛.

目标函数通常选择点和点之间的距离之和作为度量准则，这种方法可以保证算法以更快的速度达到收敛. 距离我们采用的欧几里得距离 (L2) 范式. 对数据集合 $D=\{x_1,x_2,\cdots,x_n\}$，进算法进行划分后集合为 $C=\{C_1,C_2,\cdots,C_k\}$. 目标函数计算如下

$$SSE(C)=\sum_{k=1}^{n}\sum_{x_i\in C_k}\parallel x_i-c_k\parallel^2 \tag{4-3}$$

式中，c_k 是簇 C_k 的中心点，计算方法如下所示：

$$c_k=\frac{\sum_{x_i\in C_k}^{n}x_i}{\mid C_k\mid} \tag{4-4}$$

算法目标就是通过不断的迭代选择能够最小化 SSE 的聚类结果. 选择均值作为 SSE 的最好中心点的原因如下推导：

$$SSE(C)=\sum_{k=1}^{k}\sum_{x\in C_k}(c_k-x_i)^2 \tag{4-5}$$

对式求导，令导数等于 0.

$$\frac{\partial SSE}{\partial c_j}=\frac{\partial\sum_{k=1}^{k}\sum_{x\in C_k}(c_k-x_i)^2}{\partial c_j}=\sum_{k=1}^{k}\sum_{x\in C_k}\frac{\partial(c_k-x_i)^2}{\partial c_j}=\sum_{x_i\in C_j}2\cdot(c_j-x_i)=0 \tag{4-6}$$

$$\sum_{x_i\in C_j}2\cdot(c_j-x_i)=0\Rightarrow|C_j|\cdot c_j=\sum_{x_i\in C_j}x_i\Rightarrow c_j=\frac{\sum_{x_i\in C_j}x_i}{|C_j|} \tag{4-7}$$

式子中 C_k 为第 k 个簇，x_i 是从属于 C_k 的数据点，c_k 是 C_k 中所有数据点的均值点.

算法流程图如下：

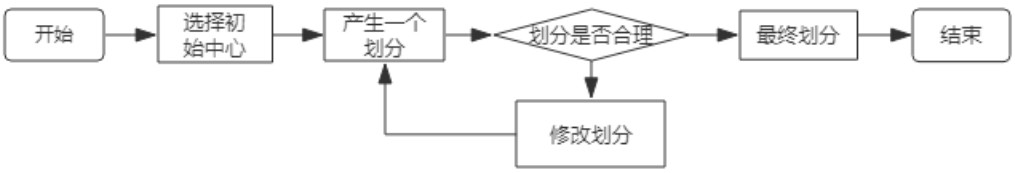


图 4-8 算法流程图

算法步骤：

Algorithm 4.1 $K-means$

Input:

All the set of point, A;

The number of clusters, k.

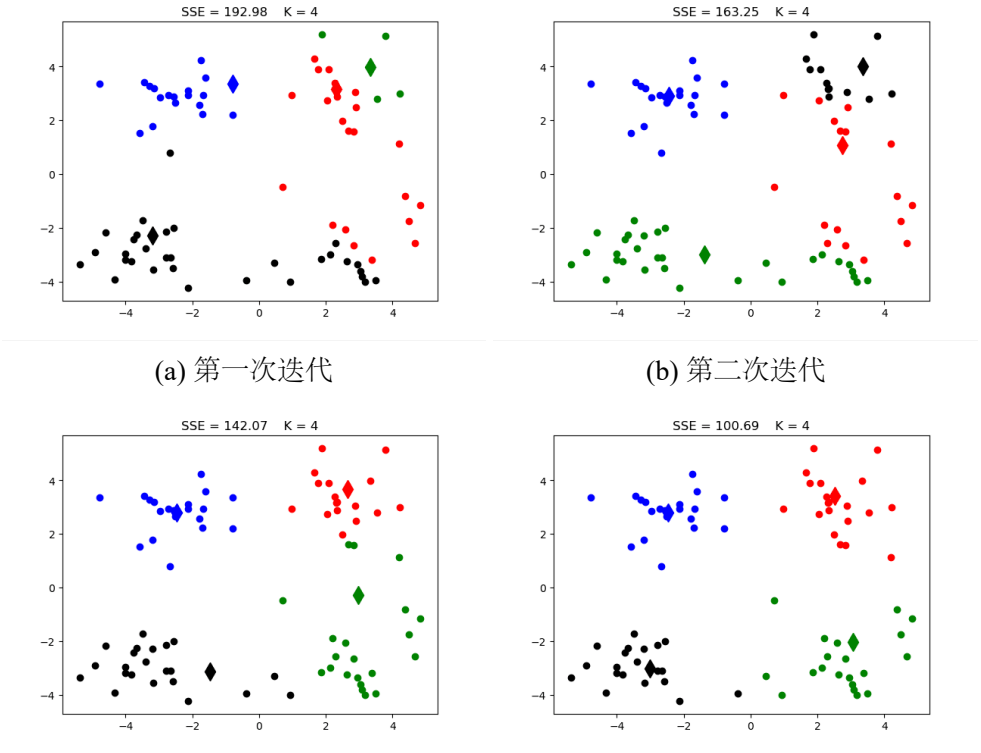
Output:

k cluster center points.

1: Randomly select k initial center points;
2: **repeat**
3: Calculate the distance between each point and its own center point;
4: Assign the points to other clusters with the closest distance to your center;
5: By formula (4-5) obtained c_k, update the cluster center point;
6: **until** The center point does not change;
7: **return** k coordinate.

4.2.3 性能分析

为了更好的分析算法性能，现在对一组数据进行迭代分析，迭代过程如下图所示：



(a) 第一次迭代　(b) 第二次迭代

(c) 第三次迭代　(d) 第四次迭代

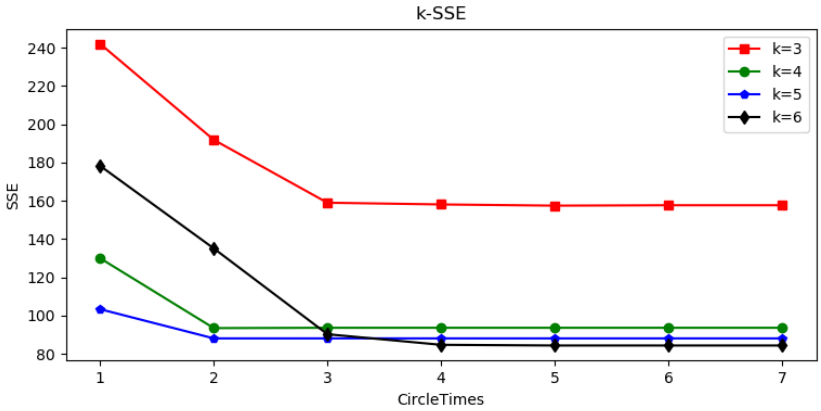
图 4-9 $K-means$ 算法执行过程

下面对算法的时间复杂度和空间复杂度进行分析：

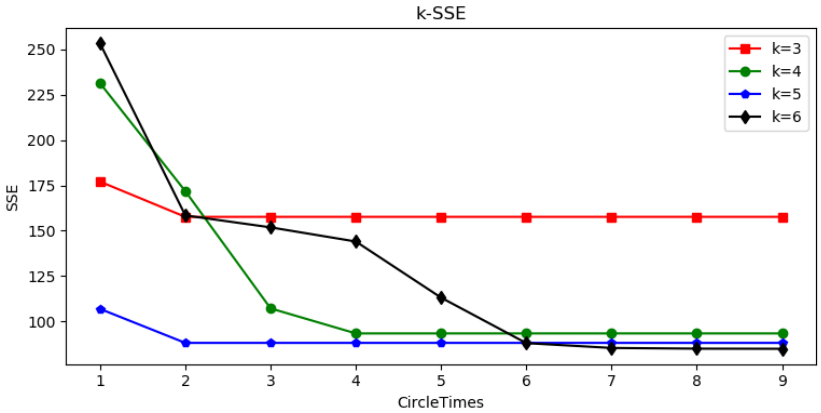
时间复杂度：$K-means$ 算法的时间需求大体上与数据点的个数成线性相关. 具体来说，其时间复杂度为 $O(i*k*n*m)$，k 是簇的个数，i 是迭代次数，n 是数据集中数据点的个数，m 是每个数据的属性数. 由于收敛大多发生在早期阶段，i 的值通常比较小，如上图所示，i 为 4 就收敛了. 这样，只要簇的个数 k 远小于 n，算法的计算时间就与 n 成线性相关.

空间复杂度：$K-means$ 算法需要存放的类容只有数据点和每个簇的中心点数据. 具体来说，其空间复杂度为 $O((k+n)*m)$，这几个量的定义与计算时间复杂度时的定义一样. 一般可以认为，i，k 和 m 都是常量. 这样的话，算法的时间复杂度和空间复杂度都可以简化为 $O(n)$，也是线性的.

故可以看出该算法是一种计算简单而有效的聚类算法.



(a) 实验 1 k-SSE 迭代图



(b) 实验 2 k-SSE 迭代图

图 4-10 性能分析

4.3 实验过程

4.3.1 实验环境

本章节主要使用的开发软件是 Gephi，Gephi 继承了图论中对图的定义，也使用了相同的概念和术语，也有部分相同的研究内容，且包含了网络科学中对网络研究的模式和方法. Gephi 是在 Netbeans 平台上开发，语言是 JAVA，并且使用 OpenGL 作为它的可视化引擎. 本文实验环境如下表所示：

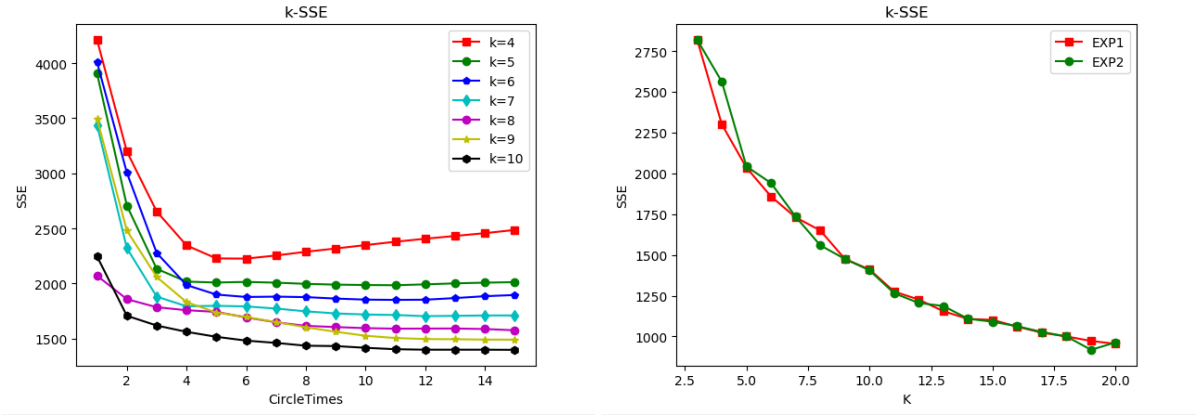
4.3.2 实验结果

用户关注的人可以体现出用户的兴趣所在，对用户的出度进行聚类分析，聚成的类即表示用户的兴趣，取采集数据的前一千数据进行分析. 由于 K-means 的聚类方法中的聚类个数不能确定，为了防止实验误差，做了两次实验:

对图中的函数关系可以看出，随着 K 值的增大，SSE 越来越小，说明聚类个数越

表 4-2 实验环境

类别	描述	类别	描述
开发语言	Python3.6	开发软件	Gephi0.92
操作系统	Windows10 pro	系统环境	Anaconda4.5
CPU 型号	Intel i5-4210M	内存大小	4GB
GPU 型号	Intel(R)HD Graphics 4600	显存大小	2GB



(a) K-means 实验过程　　(b) K-meansK 的 K 值选择

图 4-11 聚类分析

多聚类效果越好 (SSE 越小)，但是为了选取使 SSE 最小的 K 值，那么就是说 K 取最大值即每个数据为一个类，显然这样做是毫无意义的. 因此我们应该采取适当的方法来确定合适的 K 值，从图像中可以看出，当 K 很小时，SSE 很大，随着 K 的增大 SSE 迅速减小. 所以我们可以选取趋于平缓的 K 值作为聚类结果即可，这里的 K 可以取大于 12 的值，都可以取得很好的聚类效果. 选取 K 值为 15 时的聚类结果如下图所示:

4.3.3 实验改进

从上一节的聚类结果来看，聚类结果并不是很显著，因此对实验进行改进.

算法采用的是复杂网络中的社团发现算法，同 K-means 聚类算法一样，不能事先确认社区的数目，于是必须有一种度量的方法，本文采用的是模块度的度量方法进行的实验. 用模块度来衡量社区的划分是不是相对比较好的结果. 好的聚类结果在社区内部的节点相似度比较高，而在社区外部节点的相似度较低. 通常来说，当模块度值越大说明网络的划分比较理想，模块度值范围一般在 0 到 1 之间，值越大则可以说明社区结构准确度越高，在实际网络中，比较理想的模块度的值一般介于 0.3 到 0.7 之间.

现在对第三章爬取的数据为数据集，选取 11109 个实验节点，边 13487 条，实验结果如下:

实验结果分析：聚类结果一共形成了 24 类，并以不同的颜色表示出来，其中用户 vczh 和 Stuff 共同关注 27 个人，这些人大都喜欢摄影和旅游，然后通过他们的主页观察他们的赞助的 Live，关注的话题，关注的专栏，可以验证分析他们的兴趣确实有摄

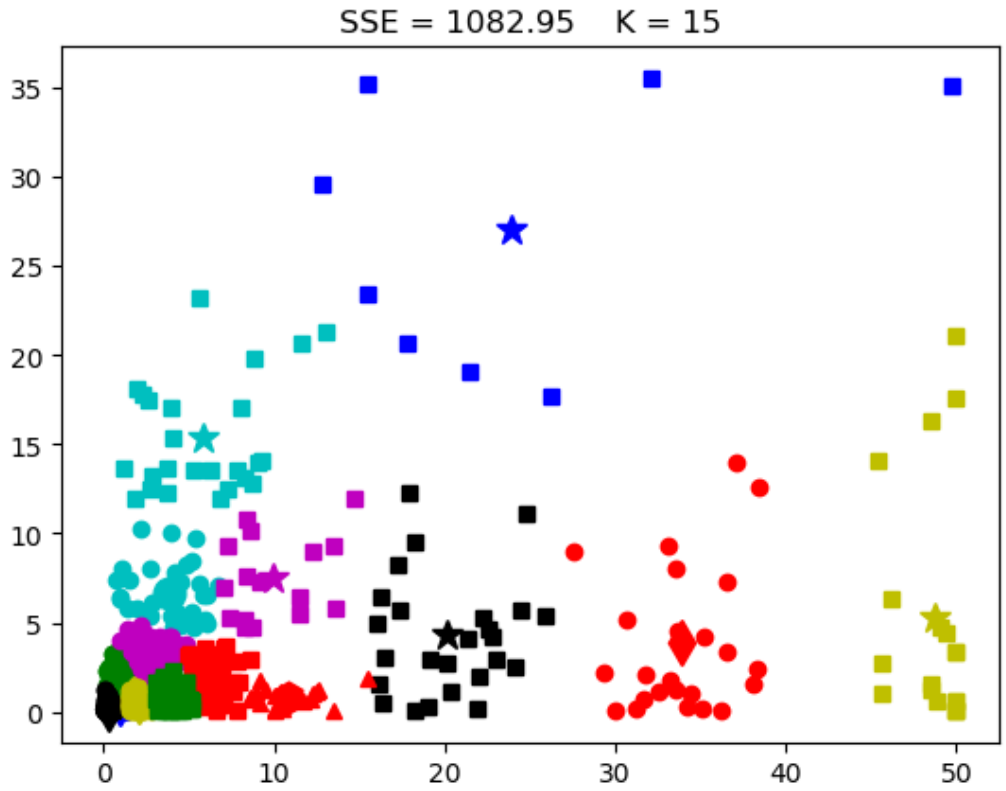


图 4-12 聚类结果

影和旅行. 实验取得了良好的聚类效果. 网络特征如下:

表 4-3 实验结果

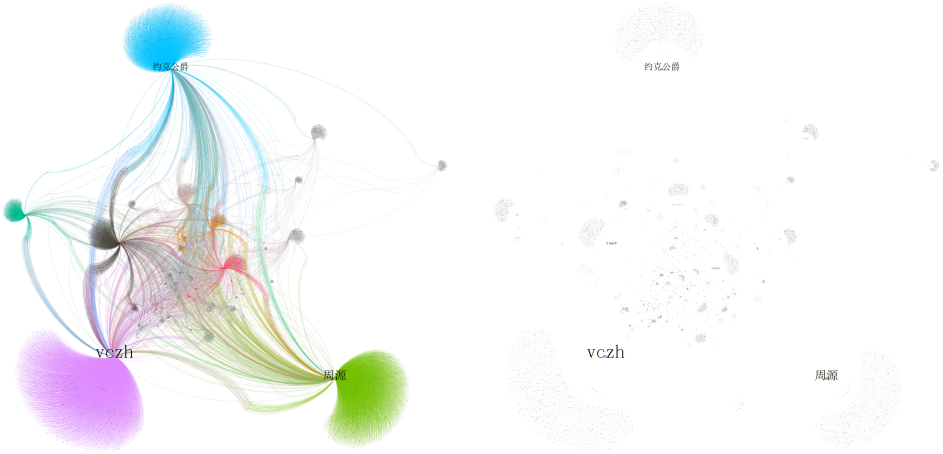
网络特征	参数	网络特征	参数
平均度	1.214	模块度	0.738
平均加权度	1.223	网络直径	6
平均聚类系数	0.028	平均路径长度	2.85

从数据结果来看，模块度的值为 0.738，大于 0.3 说明聚类结构还是比较理想的.

4.4 小结

通过实验可以看出 K-means 算法的优点：算法可以使用于各种数据；当数据的簇是凸的，簇与簇之间大小相近且差异明显时，算法可以表现出良好的聚类效果；对于大规模数据集合，该算法非常高效且伸缩性较好.

本章节对用户之间的网络关系进行分析，通过复杂网络技术对用户进行聚类，得出用户的实体聚类图，取得了良好的效果，得出了知乎用户关系复合复杂网络中的 WS 小世界模型，具有复杂网络的特性，并计算出了复杂网络的特征.K-means 算法还存在一些不足，该算法在处理具有分类属性的数据就无从下手，初始值的选取对算法比较重



(a) (b)

(c) (d)

图 4-13 实验结果

要，算法对噪声点和离群点比较敏感等缺点.

第 5 章 结论与展望

本文主要对基于复杂网络的知乎用户数据挖掘惊醒总结，阐述本论文的主要成果和优点，并总结论文中的方法还存在改进和完善的地方.

5.1 结论

数据挖掘可以通过预测未来的趋势，做出前瞻性的，具有理论依据的决策具有重要意义. 从大量信息中发现隐含的，有价值的信息是数据挖掘的目标，而聚类是数据挖掘中的重要一环. 本文数据挖掘的主要任务是对数据进行采集，并基于复杂网络技术对用户之间的关系进行分析，得出用户实体聚类图，取得了不错的效果

总结来说，本文的主要成果如下:

(1) 对复杂网络技术和知识以及复杂网络特征进行简单的分析，通过研究文献对相关研究做出了一定的综述工作.

(2) 针对特定的挖掘数据进行大量的采集，并对用户数据进行简单的可视化工作.

(3) 针对本文采用的聚类算法进行充分的研究，研究算法的优势和缺点，并对算法进行了适当的改进.

(4) 通过对知乎用户之间的网络进行分析，得出用户之间的网络具有复杂网络的特征，同时采用复杂网络技术对知乎用户之间进行分析，计算用户之间的相异度，得出用户之间的实体聚类图并分析实际效果

5.2 不足与展望

因为数据挖掘本身就是一件复杂的研究领域，所以由于条件和时间等方面的原因，本文的方法还存在需要改进和完善的地方，主要如下:

(1) 爬取用户资料没有使用知乎 API，而是采用的 Scrapy 框架，爬取效果并不是很好，速度有点慢. 以后再进行大数据爬取是可以采用 API 爬取或者使用分布式，可以大大提高采集速度.

(2) 在对知乎用户资料分析时，采用的算法在时间和空间开销存在着很大的优化空间，而且在针对大量的数据时，会消耗大量的资源和时间，一部分原因可能存在电脑配置不够高的结果. 在未来的研究生涯中，可以针对优化算法的时间复杂度和资源优化可以进一步改善.

(3) 在分析知乎用户之间的复杂网络的关系，只能对用户之间的关系进行简单的分析，下一步工作对用户的行为进行分析，采集大量的用户行为信息 (比如用户的回答，发布的文章，评论等) 这些都可以体现用户的兴趣所在，因此在未来的研究中，可以对文本信息进行文本划分以及文本分类，进而构建用户兴趣建模，更加深度的对用户进行分析.

参考文献

[1] LI C, BISWAS G. Unsupervised Learning with Mixed Numeric and Nominal Data.[J]. Knowledge & Data Engineering IEEE Transactions on, 2002, 14(4): 673 – 690.

[2] AHMAD A, DEY L. A k -mean clustering algorithm for mixed numeric and categorical data[J]. Data & Knowledge Engineering, 2007, 63(2): 503 – 527.

[3] 李佩, 刘玉树. Agent-Based Data Mining Framework for the High-Dimensional Environment[J]. 北京理工大学学报 (英文版), 2005, 14(2): 113 – 116.

[4] PAN D, SHEN J Y, ZHOU M X. Incorporating Domain Knowledge into Data Mining Process: An Ontology Based Framework[J]. 武汉大学学报 (自然科学英文版), 2006, 11(1): 165 – 169.

[5] QIAN X, WANG X. A New Study of DSS Based on Neural Network and Data Mining[C] // International Conference on E-Business and Information System Security. 2009: 1 – 4.

[6] MILLER L D, SOH L K, SAMAL A, et al. A Comparison of Educational Statistics and Data Mining Approaches to Identify Characteristics That Impact Online Learning.[J]. Journal of Educational Data Mining, 2015, 7(3): 737 – 741.

[7] TSANTIS L, CASTELLANI J. Enhancing Learning Environments through Solution-based Knowledge Discovery Tools: Forecasting for Self-Perpetuating Systemic Reform[C] // . 2001: 39 – 52.

[8] SHARMA M, GOYAL A. An application of data mining to improve personnel performance evaluation in higher education sector in India[C] // Computer Engineering and Applications. 2015: 559 – 564.

[9] ZHANG D, ZHOU L. Discovering golden nuggets: data mining in financial application[M]. [S.l.]: IEEE Press, 2004: 513 – 522.

[10] 邵金楠. 经典贝叶斯决策理论概述 [J]. 小作家选刊, 2017(9).

[11] 陈希孺. 最小一乘线性回归 (下)[J]. 数理统计与管理, 1989(6): 48 – 55.

[12] 黄河燕, 曹朝, 冯冲. 大数据情报分析发展机遇及其挑战 [J]. 智能系统学报, 2016, 11(6): 719 – 727.

[13] FEYYAD U M. Fayyad, U.M.: Data mining and knowledge discovery: making sense out of data. IEEE Expert 11(5), 20-25[J]. IEEE Expert, 1996, 11(5): 20 – 25.

[14] 任新社, 陈静远. 关于数据挖掘研究现状及发展趋势的探究 [J]. 信息通信, 2016(2): 171 – 172.

[15] RYABOY D, RYABOY D. Scaling big data mining infrastructure: the twitter experience[M]. [S.l.]: ACM, 2013: 6 – 19.

[16] SUN Y, HAN J. Mining heterogeneous information networks: a structural analysis approach[J]. Acm Sigkdd Explorations Newsletter, 2012, 14(2): 20 – 28.

[17] KANG U, FALOUTSOS C. Big graph mining:algorithms and discoveries[J]. Acm Sigkdd Explorations Newsletter, 2013, 14(2): 29 – 36.

[18] AMATRIAIN X. Mining large streams of user data for personalized recommendations[J]. Acm Sigkdd Explorations Newsletter, 2013, 14(2): 37–48.

[19] 钱学森, 于景元, 戴汝为. 一个科学新领域——开放的复杂巨系统及其方法论 [C] // 中国系统工程学会第六次年会. 1990: 526–532.

[20] ERDOS P, RENYI A. On random graphs[J]. Publicationes Mathematicae, 1959, 6(4): 290–297.

[21] BARABáSI A, ALBERT R. Emergence of Scaling in Random Networks[J], 1999.

[22] WATTS D J, STROGATZ S H. Collective dynamics of |[lsquo]|small-world|[rsquo]| networks[J]. Nature, 1998, 393: 440–442.

[23] NEWMAN M E, MOORE C, WATTS D J. Mean-field solution of the small-world network model.[J]. Physical Review Letters, 2000, 84(14): 3201–3204.

[24] KONG X X, HOU Z T, SHI D H, et al. Markov Chain-based Degree Distributions of Evolving Networks[J]. 数学学报 (英文版), 2012, 28(10): 1981–1994.

[25] HOU Z, KONG X, SHI D, et al. Degree-Distribution Stability of Growing Networks[C] // International Conference on Complex Sciences. 2009: 1827–1837.

[26] KAISER M. Mean clustering coefficients: the role of isolated nodes and leafs on clustering measures for small-world networks[J]. New Journal of Physics, 2008, 10(8): 083042.

[27] SPIZZIRRI L. Justification and application of eigenvector centrality[J]. Algebra in Geography: Eigenvectors of Network, 2011.

[28] GILBERT E N. Random Graphs[J]. Annals of Mathematical Statistics, 1959, 30(4): 1141–1144.

[29] ERDőS P, RéNYI A. On the evolution of random graphs[J]. Transactions of the American Mathematical Society, 2012, 286(1): 257–274.

[30] DJ W, SH S. Collectivedynamics of ’small-world’networks[C] // Nature. 1998: 440–442.

[31] ALBERT R, BARABASI A L. Statistical mechanics of Complex Network[J]. Review of Modern Physics, 2002, 74(1): xii.

[32] 张建萍, 刘希玉. 基于聚类分析的 K-means 算法研究及应用 [J]. 计算机应用研究, 2007, 24(5): 166–168.

致　谢

首先感谢我的毕设导师方伟老师，感谢他在毕设的各个环节对我的提醒、检查和督促. 至始至终一步步指导我完成，耐心的解答了我的所有疑惑，并明确为我制定了详细的毕设执行时间表，帮助我明确那个时间阶段干什么，具体怎么做，以及如何完善给出了详细的规划，将原来无从下手的毕设，分解成一个一个的小阶段，指导我去完成. 并且他在工作中认真负责，在生活中积极开朗的性格也深深的感染了我.

感谢所有任课老师孜孜不倦地教导，让我在大学四年学到了方方面面的知识. 再次感谢我的班主任宋春霖老师，他热情开朗的性格以及对同学负责的态度对我产生了深远的影响. 还要感谢现在的班主任宋威老师，感谢最后大学期间对我们的学习和生活的照顾，还要感谢我的同窗室友，感谢他们在我最需要帮助的时刻，给我指导性的意见和想法，帮助我完成毕业设计.

感谢我的家人，感谢他们对我的无私奉献，是我一直以来的坚强后盾，帮助我一起面对困难，战胜困难.

最后，真诚的感谢所有给我提供帮助和鼓励的人.