**1.1课题背景：**

随着以互联网为代表的各种网络技术的飞速发展，网络上存储的各种信息已经可以用海量来形容, 并呈现出指数增长的趋势，成为21世纪最为重要的经济资源之一。如何在海量的信息中快速检索到所需要的信息，成为了网络技术研究领域的一大热点。最初始的搜索引擎采用人工方式收集信息，由编辑员查看信息之后，人工形成信息摘要，并将信息置于事先确定的分类框架中。这种方法虽然准确可靠，但随着网络的飞速发展，在信息收集的数量和更新速度上已经远远无法满足要求，因此目前绝大多数搜索引擎都采用自动化的信息收集方式，由一个称为网络爬虫的机器人程序以某种策略自动地在互联网中搜集和发现信息，由索引器为搜集到的信息建立索引。该类搜索引擎的优点是信息量大、更新及时、毋需人工干预。可以看到，网络爬虫程序的质量直接关系到搜索引擎的质量，因此是目前网络大数据技术研究领域的一个热点。  
 网络爬虫，又称网页蜘蛛、网络机器人。随着计算机技术的高速发展，互联网中的信息量越来越大，搜索引擎应运而生。传统的搜索引擎会有返回结果不精确等局限性。为了解决传统搜索引擎的局限性，专用型网络爬虫在互联网中越来越常见。同时，专用型网络爬虫具有专用性，可以根据制定的规则和特征，最后只体现和筛选出有用的信息，并且可以对爬取得数据进行有效的数据分析实现快速的获取数据获取加快大数据在现实世界的应用。

**1.2发展历史：**

一开始世界上没有人写爬虫，后来网络上的数据多了，人看不过来了就有了最初的爬虫程序，进而出现早起的搜索引擎。

之后爬虫最大的应用还是在搜索引擎，一般的企业都忙着做网站，做系统。后来一些咨询公司需要分析网上的数据，出现了一些爬虫岗位，后来网络上的舆情事件层出不穷，针对网络舆情有了大量的需求，于是很多出现了很多爬虫岗位，这个阶段，爬虫的岗位技术门槛不高，采集的对象都是些新闻资讯站点。

再后来 大数据和数据挖掘火了，很多人都意识到数据中的财富，一时间各大小公司都组件大数据部门，结果发现没数据，于是想到了爬虫，这时候爬虫的采集对象多了，电商，论坛，机票，本地服务其中某两家本地信息平台互相采集对方数据，形成最早的爬虫与反爬虫。

逐渐的社区出现一大批开发爬虫，采集技术门槛越来越低，用心研究一下，就能实现大规模的数据采集很多公司往往设置一个爬虫岗位，领导没有长远规划，只是下达采集任务。

由于爬虫的岗位稀少，需求量一时很大，导致薪资比一般开发偏高，大批人员涌向爬虫岗位。

目前招聘爬虫的以大公司为主，一般划分为大数据岗位，另一些稳定岗位是互联网金融公司，长期需要采集数据目前看不到爬虫岗位增加的趋势，一般采集需求，现有人员利用开源爬虫即可完成互联网金融公司的爬虫一般服务于业务，但基本上各个维度的信息都有团队攻克，业界都有商业解决方案，大公司基于数据安全考虑会自建采集团队，一般小公司基于成本考虑会购买成熟服务，打了一定体量才会构建采集能力。

随着竞争的激烈，爬虫的技术会逐渐公开化，各家的竞争点将是稳定性和其他数据附加。

**1.3发展现状：**

随着大数据时代的来临，互联网对人类的生活影响越来越深入，已经成为人类获取信息的主要来源之一。互联网为用户带来海量数据的同时也带来了困扰，如何及时获得有效信息成为研究重点。搜索引擎根据预定的策略从互联网上发现和抓取数据，存入本地；对数据进行去噪、抽取和生成索引等处理，最终为用户提供信息检索服务，将相关信息展示给用户的系统。

爬虫（Crawler）是搜索引擎架构中的最底层模块，以一定的策略从互联网上抓取数据，并对数据进行预处理，然后将处理后的数据提交给搜索引擎其他模块，数据的质量和数量直接影响用户的体验。但是，随着大数据时代互联网数据爆炸式增长，爬虫抓取数据的速度越来越不能满足实际应用的需要。解决这个问题主要从软硬件考虑：一是升级爬虫的硬件，使用性能更好的硬件设备，但性价比不高，且不易扩展；二是利用分布式方式提高爬虫的并行处理能力，但这种方法会增加爬虫系统设计的复杂度。

目前，大多数大型爬虫系统采用的是分布式方式，但仍然不能满足用户的实际需要。其次，爬虫系统还需解决网页动态变化导致本地副本过期的问题。网页随时都可能变化，有的几小时更新一次，爬虫系统必须及时的发现并更新本地的网页，但互联网海洋中网页数量多、分布广，爬虫系统更新一遍需要数周甚至更长的时间，使得本地库中网页副本时新性较低。因此，一个采集速度快，网页更新及时的高可靠爬虫系统，不仅仅为搜索引擎提供基础数据，也可以为数据分析、挖掘提供基础数据，从而获得信息、知识。

**1.4研究意义**

1.深入学习巧Python和Soapy-Redis开源框架，自己动手实现有良好拓展性的网络爬

虫原型，将对我们学习新技术和拓宽眼界有着积极的作用。

2.虽然实现的只是一个原型程序，但是探索简单易模改易拓展的思路是非常正

确的。针对不同用户的特定需求提供特定的功能，正是如今程序开发领域流行的思

想。简单易模改的爬虫将是很好的实践。

**1.5研究目标**

本课题的研究目的在于使用基于Python的开源技术，结合其他网络编程的知识等，实现一个自定义的拓展能力强的网络爬虫原型程序。

1.基于Soapy-Redis框架，自定义的爬取规则。

2.选择合适的数据库进行数据存取。

3.采用多线程并发结构，提高运行效率。

4.由于是本科阶段不管是学习时间上的不足或者难度比较大工程量较大，所以本课题主要是研究高效强鲁棒性的爬虫系统，并且爬取到的数据进行一个非常简单的数据分析。

**1.6研究中遇到的关键问题**

1.突破目标网站对爬虫的限制，如拒绝访问、强制登出、封锁IP 、验证码、用户Cookie封锁等。破解网页的限制将是爬虫是否能高效运作的关键。

2.解决URL重复问题。在高速运行的爬虫工作过程中，如何鉴别即将爬取的网页是否己经爬取过极大影响了运行效率和资源的利用程度。

3.爬虫爬取有记忆可以中断，再启动时候可以继续中断时的爬取。这样做可以防止出现意外中断爬虫程序就前工尽废，增强爬虫程序的鲁棒性。

4.多线程并发实现。根据爬取目标网页的设计，如何设计更加高效利用CPU能力，怎样设计多线程的并发代码也需要学习解决。

5.编码问题。为了爬虫的数据表示有效使用Base64编码传输，为了应对动态交互AJAX网站表单等部分数据不可见使用JSON格式，数据库也是用MongoDB也是JSON格式。

**1.7 论文主要组织结构：**

本论文分八章节进行阐述:

第一章,绪论。首先介绍本文的研究背景网络爬虫的产生与发展以及在数据分析等邻域中数据获取的手段，接着是本课题的研究目的和意义。最后给出了论文的研究内容及论文结构安排。

第二章,分别介绍了网络爬虫系统的基本知识和相关技术。在这一章介绍爬虫的分类及作用，随后介绍网络爬虫涉及到的网络协议，最后介绍爬虫搜索策略。

第三章,爬虫总体架构设计。本章主要是爬虫系统的需求与架构设计进行了 详述的描述。首先介绍爬虫系统的业务需求，性能需求和其他需求，然后完成爬虫系统使用的Scrapy框架，再来是分别介绍两个NOSQL数据库，一个MongoDB是放在磁盘层作为获取的数据的持久化操作，另一个Redis内存数据库放在内存中一方面进行查重，另一方面进行爬虫的调度。最后介绍Scrapy+redis网络爬虫的架构以及实现。

第四章,爬虫关键问题实现。介绍爬虫系统的具体实现的时候会遇到的一些难题比如User-agent伪装，用正则表达式编写信息的过滤规则，怎么样对抗微博的反爬技术，如何打码，如何查重，并建立Cookie池,讨论并得到解决方案。

第五章,系统测试。介绍了系统实现的软件与硬件环境，爬虫爬取数据等进行简单的测试。

第六章,数据库设计与分析。对爬虫系统设计数据库，在MongoDB数据库上设计微博用户信息等三个表，E-R图，具体结构设计，逻辑设计等。然后对爬虫爬取到的微博的用户信息进行简单的数据分析得出一些简单的统计等结论。

第七章,总结与展望。总结本次论文研究课题的完成情况。对系统的整体设计与实现做回顾，分析自己的设计存在不足，展望进一步改进的主要方向。对本课题下一步发展方向做一些分析。

最后，向所有的在本人本科阶段研究哦期间给我支持帮助，指导点拨的家人，前辈，同学，朋友和老师致谢。