

重庆大学本科学生毕业设计（论文）

基于神经网络的时事投资分析系统



学 生：陈 点

学 号：20125209

指导教师：涂风华

校外指导教师：罗 平

专 业：物联网工程

重庆大学计算机学院

二〇一六年六月

Graduation Design(Thesis) of Chongqing University

**Design of Real-time Investment Analysis
System Based on Neural Network**



Undergraduate: CHEN Dian

Supervisor: TU Fenghua

Off-campus Supervisor : LUO Ping

Major: Internet of Things

**Computer Science
Chongqing University**

June 2016

摘 要

信息爆炸的时代，资讯的数量日益增多，在面对大量需要阅读的专业情报时，专业人士的知识储备和人员数量便成为了限制或门槛。但毕竟人各有专精，并且技术人员地域分散，在资讯发表速度逐渐迫近人们处理速度的情况下，该如何高效获取文本内容中的关键情感与信息则是不得不面临的问题。

本项目实现了一个自动获取实时更新的资讯，并借助机器学习算法提取出内容中的情感倾向与相关情报，通过网页的形式展现给用户的时事分析系统。特别地，我们探讨的是金融领域上的资讯，即金融领域的研报投资分析系统。

在抓取方面，该系统通过实现一个自动获取代理来提供不同 IP 地址的模块，来完善实时自动检测新资讯出现的功能。使用 urllib2 库的方法获取数据之后，通过 BeautifulSoup4 或者 Goose 进行解析，从而获得结构化的有效数据。

在分析方面，该系统通过有监管机器学习预先训练模型，利用模型加载完毕后处理输入语料的高效性来大幅提升分析性能，该系统中内置了 LR、LibLinear、CNNTText 模型，和一个开源的知识型卷积神经网络模型。

当抓取的资讯被分析完毕后，由数据库存储管理，由 PHP 前端页面负责展示。即该系统是一个从信息获取输入直到前端信息页输出的完整流程实现。此外，系统中也考虑了中断重开后的快速恢复以及面向其他领域与算法的可扩展性。

关键词：爬虫，数据挖掘，机器学习，文本分析，稳健性与可扩展性

ABSTRACT

In the century of information explosion, there is a increasing number of information, and when we have to be in face of reading a lot of professional intelligence, knowledge reserves and the number of personnel professionals has become the limit or threshold. Because people all have their own expertise and the technicians disperse geographically, in case that the rate of information publication has gradually approached people's processing speed, we have to consider how to efficiently get the key emotion and the main information in the text contents.

This paper implements a current affairs analysis system, which can automatically get the real-time updated information contents, and extract the emotional tendencies and relevant information in the contents, and finally present the analysis result to users via web form. In particular, we discuss the information on the financial sphere, which is what we called Financial Sphere Research Report Investment Analysis System.

In the aspect of crawling, this system implements an automatically acquiring proxy module to provide different IP addresses, which improves the function of real-time automatic detection of new information. After using urllib2 library to get the data, the data is parsed by BeautifulSoup4 or Goose to obtain the valid structured data.

In the aspect of analysis, the system considerably improves analysis performance through pre-trained supervised machine learning model and efficiently processing the input corpus after loading model. The system is built into LR, LibLinear, CNNText model, and an open source “knowledgeable convolutional neural network” model.

When a crawled information is analyzed, it is stored and managed by the database when PHP is responsible for displaying the front page. That is, the system is a complete process from taking input to the front page showing output. In addition, the system also covers the rapid recovery after the interruption and re-opened, and scalability for other areas and algorithms.

Key words: Crawlers, Data Mining, Machine Learning, Text Analysis, Robust and Scalability

目 录

中文摘要	I
ABSTRACT	II
1 绪论	1
1.1 研究工作的范围	1
1.1.1 大数据方向	1
1.1.2 金融领域	2
1.1.3 研报资讯	2
1.2 现状及需求分析	4
1.2.1 现状需求	4
1.2.2 研究目标	5
2 项目实现概述	6
2.1 整体框架	6
2.2 问题评估	7
2.2.1 研究中的基本假设	8
2.2.2 实现的难点与挑战	9
3 主模块的组成及实现	11
3.1 代理池模块/Retainer	11
3.2 实时轮询抓取模块/Crawler	12
3.2.1 实时轮询模块/Timer	14
3.2.2 代理模块/PrxMod	14
3.2.3 抓取模块/CrlMod	15
3.2.4 数据存储/MysqlMod	16
3.3 模型处理模块/Scikit	18
3.3.1 分词处理/WordSeg	19
3.3.2 训练词向量/Word2Vec	19
3.3.3 知识性卷积神经网络/KnowledgeableCNN	22
3.3.4 其他机器学习算法	24
3.4 前端展示模块/CDPage	25
3.4.1 通过 connector 连接数据库	25
3.4.2 前端 php 展示模块	26
3.5 日志模块/LogMod	27

4 模型算法及辅助模块	29
4.1 通过 Pyspider 平台获取训练集数据	29
4.2 通过 Theano 进行 GPU 训练加速	30
4.3 模型训练的设置	30
5 实测评估	33
5.1 训练过程中的参数自学习.....	33
5.2 实测运行评估及回馈评价.....	34
6 可扩展性	37
6.1 面向其他算法的扩展	37
6.2 面向其他数据集或领域的扩展.....	37
7 结论	38
致谢	39
参考文献	40
附录 A：系统相关信息及基本概念	附 1

1 绪论

当今社会早已进入了信息化大数据时代，产生的信息量以极快的速率逐日增长。个人、团体或小型规模的专业人士，倘若企图通过如此庞大的信息流获得即时情报，耗费的人力物力会越来越大——受到专家人数及个人在多领域中阅历的限制，信息更新的速度终究会超过人力所能企及的阈值，当信息达到人力所无法驾驭的数量级，产生的滞后性会造成无法及时掌控信息，甚至会因此而承担不可估量的损失。

其中，特别是在对信息依赖性极高的金融行业，数目繁多、信息庞杂的金融信息鳞次栉比，当前主流的金融门户网站^{1,2}，几乎每天都有各个行业的研报（研报的具体说明见本节 1.1.3）输出。同时，对其解读所需的专业知识和能力的要求越来越高，没有达到专家水平的绝大多数相关人员，难以迅速而准确的从中获取所需要的信息。比如，金融行业中负责投资的部门面对大量涌入的最新金融情报，从阅读中得到信息，从而得出结论投入资金。在投资良机来临时，早一秒掌握便是多一份收益。

正是针对这种情况，我们想到可以通过机器学习，来实现一种专家系统（或更进一步的知识图谱）来为人们简化阅读、降低阅读门槛。特化到金融方面，则是想要实现一种可以自动获取新发研报，提取研报中的有效信息，并附上推荐信息、标记重点实时展示给用户。

在本篇论文中，我们将以信息处理的逻辑顺序，即从信息来源开始，直到展示给用户为止的一条完整信息处理链，来介绍这个系统的设计与实现。

1.1 研究工作的范围

本次的研究可以定位为：在“大数据方向”下，针对“金融领域”的情报主体“研报资讯”，关于如何帮助用户快速处理此类信息的研究。

1.1.1 大数据方向

关于大数据方面，由于需要尽可能的提升预测与推荐的准确率，必须以大数量级的数据作为训练集。这在大数据方向上的数据挖掘与预处理的技术要求，对研究提出了较大的挑战。

在国外^[1]，研究大数据不仅仅是研究概念，还研究了大数据技术，并将技术研究作为重点。以美国为例，他们的大数据研究计划，绝大多数都是将数据工程视

¹ <http://data.eastmoney.com/report/>

² http://vip.stock.finance.sina.com.cn/q/go.php/vReport_List/kind/search/index.phtml

为重点，并从分析算法和系统效率两方面进行考虑作出设计。

在我国，当前国家将大数据研究放在了战略性的位置，加快了对大数据相关技术攻关的进程，工信部发布的《物联网十二五规划》里，把信息处理技术作为关键技术创新工程之一而提出来，其中包括了海量数据存储、数据挖掘、图像视频智能分析等大数据技术的重要组成部分。在近期推行的中华人民共和国《“十三五”规划》³中也指出了将“实施国家大数据战略，推进数据资源开放共享”作为战略之一，可见对于国内而言大数据研究的重要程度。

当前的机器学习算法往往难以具有可解释性^[2]。通过数据挖掘出的信息训练出目标模型，同时在机器学习的过程中我们也要尝试着寻找可解释的意义，这也是本次研究的目标之一。在“海量”级别的数据中进行挖掘有效信息，通过机器学习寻找隐含的规律与逻辑，这是大数据的魅力，也是我们在数据中创造价值的动力。

1.1.2 金融领域

本研究所实现的系统旨在提取各类文本中的重点与关键部分，在众多类型的文本中，金融领域是一个特殊性很强的领域。在金融领域中，由于其时效性与收益的即时性，导致与金融相关的信息与资讯往往非常具有规范性与专业性。金融领域稳定输出且高质量的情报流，毫无疑问是作为研究对象的极佳选择。

于是，期望通过基于卷积神经网络的深度学习，实现用于文本分析的专家系统。目标模型适用于金融、医疗、公共服务、政府决策等多种不同的场景，本次课题具体定位到金融方向的研报分析，目标为实现一个面向“时事投资”的分析系统，本课题旨在对不定期不定量出现的大量数据进行即时获取，抓取挖掘获得有效信息后，一方面进行即时分析，一方面备份存储完整信息于数据库，并实时地通过较为清晰的形式展现于用户面前，以达到同步高效获取信息流的效果。

1.1.3 研报资讯

在金融领域中，信息有很多表现形式，在这里我们注重于选择信息量大、更新速率稳定且在金融行业被广泛认可的“研报”资讯作为主要研究的信息对象。

一般来说，“研报”指的是⁴——

“券商或者投行的专职财务研究人员编写的，就某些上市公司的经营状态和盈亏情况做出的分析，提供给投资者作为参考的研究报告。有时会作出‘推荐’，‘观望’，‘卖出’等评价，对市场有一定的导向作用。”

研报资讯的获取也相对稳定，在我国，多家门户网站的金融板块都会定期（如东方财富网）或实时（如新浪财经）发放最新的研报信息，如图 1.1 中，新浪财经实时更新研报信息，以时间顺序展现在一个固定的域名地址上，这对于及时获取

³ <http://www.ocn.com.cn/us/shujuzhongguo.html>

⁴ <http://zhidao.baidu.com/question/476868286.html>

最新的金融信息是非常便捷的。

The screenshot shows the Sina Finance homepage with a search bar for research reports. Below the search bar is a table listing 10 research reports from May 23, 2016. The columns include序号 (Number), 标题 (Title), 报告类型 (Report Type), 发布日期 (Release Date), 机构 (Institution), and 研究员 (Researcher). The reports cover various industries like TMT, mechanical industry, pharmaceuticals, and energy.

序号	标题	报告类型	发布日期	机构	研究员
1	华融证券美股TMT行业周报	行业研究	2016-05-23	华融证券股份有限公司	安静
2	China Watch P232:Takeaways from...	行业研究	2016-05-23	德意志银行	Jack Hu
3	机械军工行业：轨交核心零部件、进口替代、国际化助推行业景气...	行业研究	2016-05-23	中国银河证券股份有限公司	王华君
4	神剑股份:主业聚酯树脂盈利进一步提升,嘉业航空前景广阔	公司研究	2016-05-23	中信建投证券股份有限公司	罗婷
5	华融证券机械行业周报	行业研究	2016-05-23	华融证券股份有限公司	张迪
6	证券行业:博取确定性“券商”机会	行业研究	2016-05-23	中信建投证券股份有限公司	杨荣
7	财政部政策对市场的影响:供给侧改革加码,钢铁煤炭成切入点	投资策略	2016-05-23	上海证券有限责任公司	王伟力
8	中国医药:业绩增长稳健,优质资产即将注入	公司研究	2016-05-23	西南证券股份有限公司	朱国广
9	新纶科技:携手T&T及日本东山,战略布局动力电池及显示用高端...	公司研究	2016-05-23	中国银河证券股份有限公司	王莉
10	常山股份深度研究报告:IT业务加速成长,公司发展进入新阶段	公司研究	2016-05-23	中国银河证券股份有限公司	沈海兵

图 1.1 新浪财经门户网站的研报信息流

对于每一份金融研报资讯而言，如图 1.2(A)，我们可以从一则研报⁵中获得有关来源、研报类别、具体分析文本等内容，而这些文本情报中往往包含了很多与投资决策相关的因素。研究的主体便是这些文本。如图 1.2(B)，金融研报具有着较为规范的格式，如末尾往往会以简明扼要的文字表述出推荐与否，但是这是面向大众用户的，较为专业的投资者会更加专注于其主题文本中所包含的信息与情感倾向。但是文本是复杂的，研报的数量与更新速度是人力所不能及的，这也是本项研究所致力解决的问题。

This screenshot shows a detailed view of a research report. The title is "中国医药:业绩增长稳健,优质资产即将注入". Below the title, it says "类别: 公司研究 机构: 西南证券股份有限公司 研究员: 朱国广 日期: 2016-05-23". The main content discusses the company's performance and asset integration. It mentions the company's strong growth and its strategic acquisition of a pharmaceutical company. The text is dense and provides specific financial data and market analysis.

图 1.2(A) 研报内容概览

图 1.2(B) 研报内容概览 (尾部)

⁵ http://vip.stock.finance.sina.com.cn/q/go.php/vReport_Show/kind/search/rptid/3271140/index.phtml

1.2 现状及需求分析

就现状而言，可以用“日益增长的信息量”与“无法普及的专业性”之间的矛盾来表述：特别地，在金融领域无论是对于情报导向的投资部门、关注经济走势的研究机构，还是有些闲钱尝试理财的散户，甚至是一时兴起观望大市的人们，都面临着信息摄取能力不足或是效率不高的问题。若是有一种可以协助阅读与理解相关情报的系统，想必在提升效率和效益上有所帮助。

1.2.1 现状需求

一个世纪前，著名作家高尔基就曾说过“书籍是人类进步的阶梯”。而在信息时代技术飞速发展的今天，可以演变为“资讯是人类进步的电梯”了。但是，和以前不同的是，现在的时代被人们称作“信息爆炸”的时代：

“据英国学者詹姆斯·马丁统计，近年来，全世界每年登记的新专利多达 70 万项，每年出版的图书数量高达 50 多万种。人类知识的倍增周期，即便是在 80 年代末就已经几乎到达了每 3 年翻一番的程度……新的理论、材料、工艺、方法的不断出现，使知识老化的速度日益加快。”

当前面临的是该如何和信息产出速度赛跑的问题，那么该如何提升效率变成了本质问题。我的想法是，和学习中的划重点类似，我们可以通过某种方法来让机器学习出如何判断一篇文本中的情感倾向，从而筛选挑出重要的部分展现于用户眼前，更进一步，我们甚至可以将支撑这次判断的依据展现出来。这样一来不但提升了阅读效率，也同时降低了技术门槛。



图 1.3 国家“十三五”计划中提出的大数据领域核心技术

1.2.2 研究目标

本次设计的标题，若更加贴切一些可以取作：《在线文档的实时抓取与分析：以投资研报为例》，在重视设计与实现的本科设计需求下更正为现题，但原考虑的标题中可以醒目的表现出本次研究的设计目标：“实时”，“抓取”与“分析”。

本次研究的目标为实现一个完整的协助阅读系统“CDReader”⁶，可以普遍地应用于各类纯文本的阅读辅助中，在本次设计中选择了金融研报的分析处理作为目标。系统通过维护一个代理池（见 3.1 节）来获得常新的虚拟 ip 地址，以此来周期性地监视信息流，然后对新出的文本进行抓取与分析（见 3.2 节），所获取的结果连同机器学习算法所计算出来（见 3.2.4 节）的结论一起，按照统一的格式写入数据库（见 3.3 节），用户通过一个 PHP 前端页面（见 3.4 节）可以随时获取到最新的信息，以一个类似朋友圈⁷时间轴的形式展现出来。

本课题的主要任务是实现一个针对金融类研报的实时分析系统，包括增量轮询的实时抓取、对获取文本高效处理，以及数据的存储与展示。实现的以金融为目标的大数据系统，如图 1.3，参照大数据所涉及的核心技术，依次介绍本次研究中各个方向的主体目标：

- 数据采集与预处理方面，需要实现针对固定信息流的实时轮询抓取。
- 数据存储与管理方面，需要设计 MySQL 优化存储，并逐步实现分布式。
- 数据分析与挖掘方面，对关键词句的提取分析，预处理成结构化数据并设计多样化的投资分析，其中每一种分析结果都需要对应的机器学习算法。
- 数据展现与应用方面，实现用户友好、可视化强的情报输出流，自行设计方案（后决定使用 PHP 时间轴形式）以展示优化后的信息。

此外，尝试设计代理池⁸以优化抓取、设计前端页面以优化展示。此外，时间与能力允许的情况下，增加对相关行业、股票等的投资建议算法 DEMO。

⁶ <https://github.com/okcd00/CDReader>

⁷ <http://baike.baidu.com/item/微信朋友圈>

⁸ <https://github.com/okcd00/CDRetainer/blob/master/readme.md>

2 项目实现概述

2.1 整体框架

本次实现的系统涉及了多个领域的不同角度，所以模块之间的关联复杂度较高。关联性既是框架解释的难点也是系统实现的重点（详见第3节），本节不多作赘述，仅采用如图4思维导图的形式，来简要概述系统的主要部分，横向介绍该系统的整体框架。

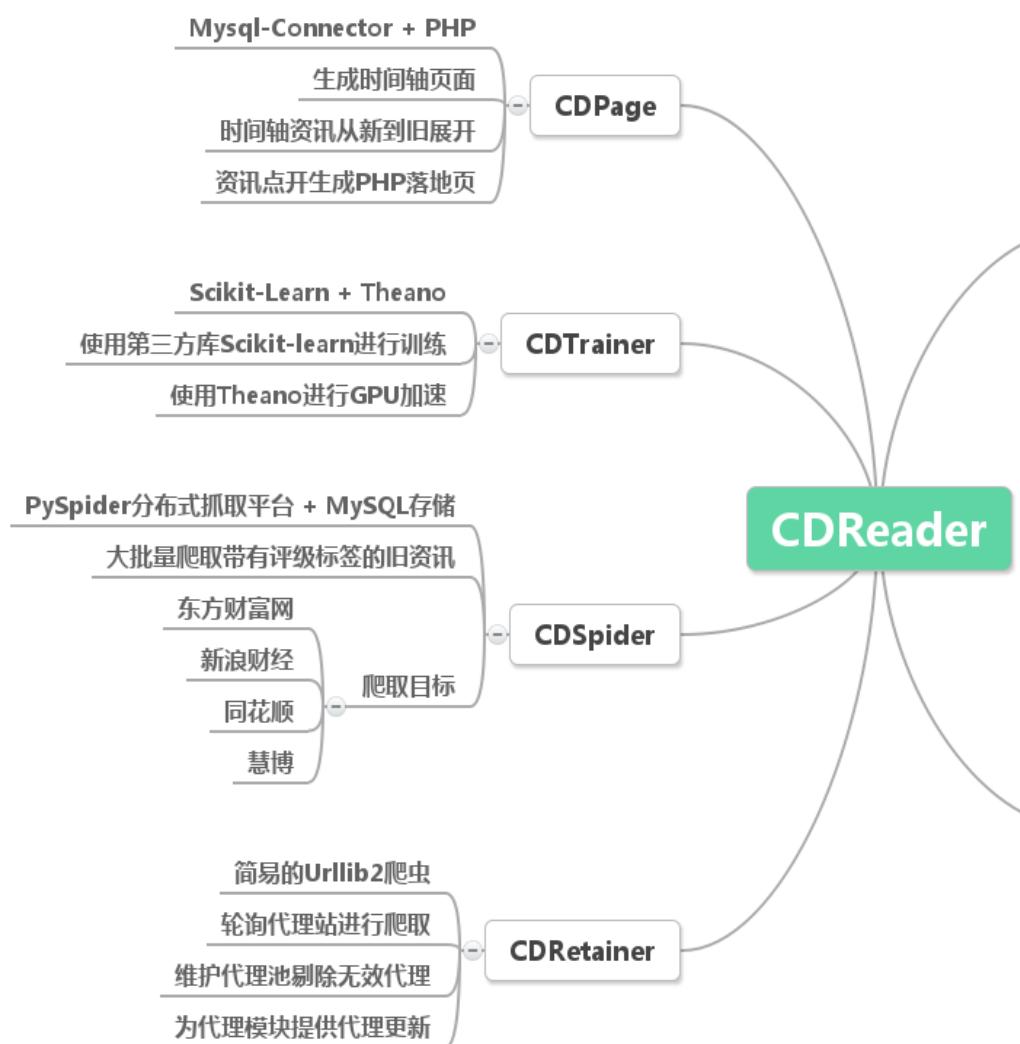


图1.4(A) CDReader系统模块一览图(左)

如图1.4(A)中，展现出系统具有将情报展现给用户的前端信息页CDPage、异

步训练以提升准确度的训练模块 Trainer、为训练模块批量抓取训练文本的爬虫平台 Spider，以及源源不断为系统提供代理的代理池模块 Retainer⁹。

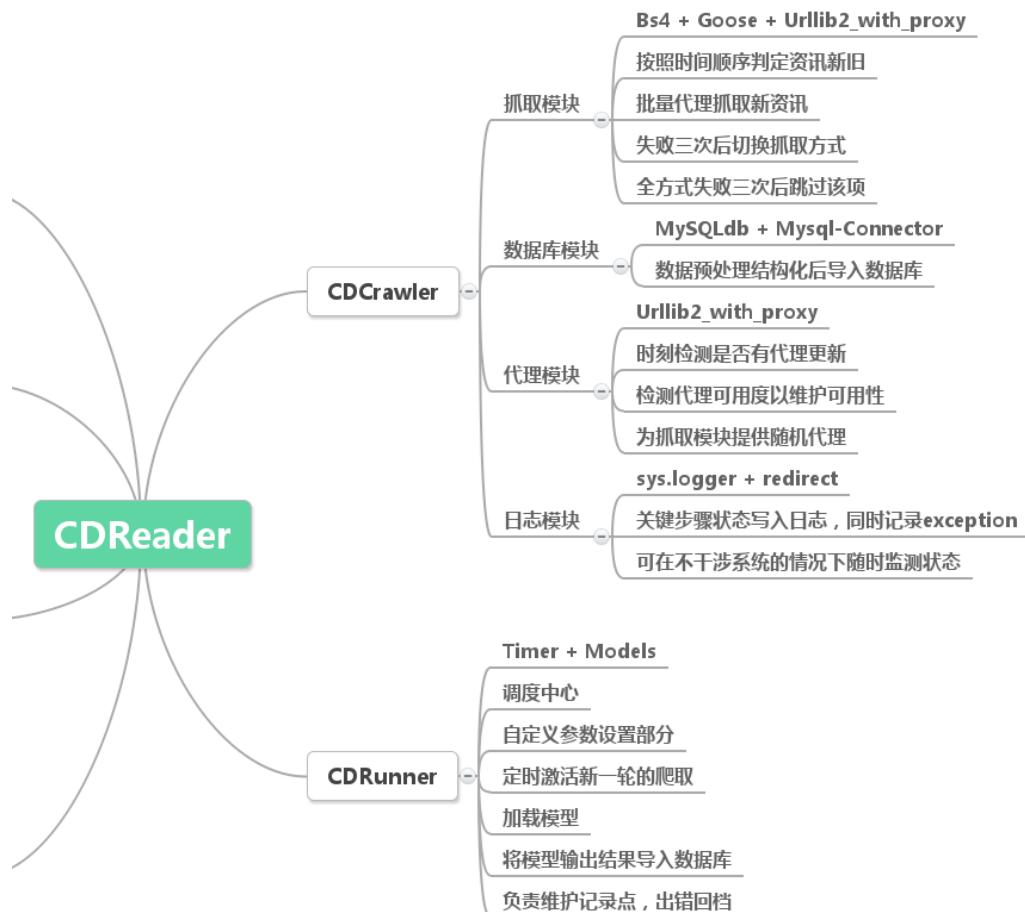


图 1.4(B) CDReader 系统模块一览图 (右)

如图 1.4(B)中所示，系统的挖掘部分（Crawler）包含集成了多种抓取方式的爬虫、精心设计的关系型数据库、与 Retainer 对接为抓取模块负责伪装的代理，以及过程中时刻记录状态报错监视的日志模块。此外，有一个中心模块（Runner）负责用户自定义的循环周期等各类变量，负责各个模块的唤醒和数据引导工作，并且在系统出错时，负责回档至上一个正常的记录点。

2.2 问题评估

我们的系统是为文本类的各类别信息服务的，但本次研究中我们将系统设计为针对金融研报相关的文本。在目标确定的情况下，我们需要对金融领域的研究中可能出现具有代表性的问题作出评估。

⁹ <https://github.com/okcd00/CDRetainer>

2.2.1 研究中的基本假设

面向金融研报的研究，需要考虑到研报编者具有多种情感。为了便于进行研报情感的信息处理，我们需要将它们数字化，为此需要作出一定程度的假设。

首先是模型训练中必需的情感分析，我们抓取了大量带有情感（此处的情感，专业名词称作“研报评级”）的金融研报数据，我们最终需要的是“正向/负向”的二元情感，于是我们以“是否含有会获取收益”为界限进行 0/1 分类的划分假设，如类似表 2.1 中的“减持”我们标记为负情感，记作 0，同时类似“确信买入”的评级词汇，我们标记其为正情感，记作 1。特别地，对于英文研报的情感分析则需要训练英文的模型，由于英文研报在总体数据中占比不大，故直接排除，仅考虑中文研报的分析处理。

训练集中人工设定的研报评级 0/1 分类¹⁰

表 2.1

标记(Sign)	研报评级(Rank)
0	中性、落后大市、观望、减持、卖出、回避、持有-落后同业
1	买入、买进、优于大市、增持、大市同步、审慎推荐、强力买入、强推、强烈买入、强烈推荐、持有、累积、持有-超越同业、推荐、确信买入、收集、谨慎买入、谨慎增持、谨慎推荐、超强大市、跑赢大市、长线买入

此外，同上述分类准则，通过机器学习得出某篇文章的情感倾向值，我们将超过阈值的视为“看涨”（如图 2.1(A)，得分为 $Labelmood = 0.999167$ ），低于阈值的视为“看跌”（如图 2.1(B)，得分为 $Labelmood = 0.252388$ ）。在这里，我们假设阈值 $Labelmood_Threshold$ 为 0.5。

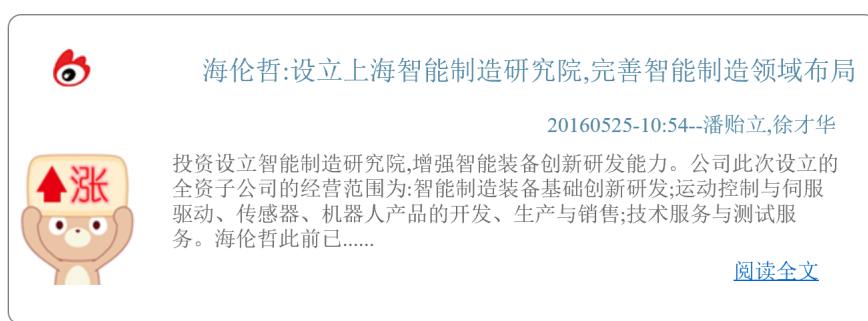


图 2.1(A) 前端页面“看涨”资讯示例

¹⁰ <https://github.com/okcd00/CDReader/blob/master/Theano/data/cdr/test/labels>

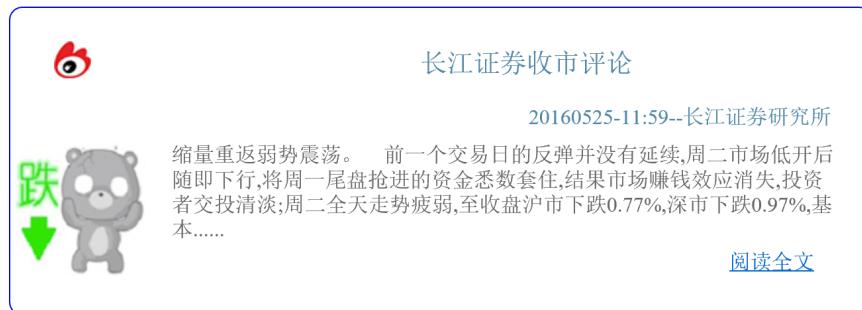


图 2.1(B) 前端页面“看跌”资讯示例

在测试阶段的抓取过程中，我们发现目标站点的东方财富网¹¹在每天的 00:00-04:00 进行更新，大多数情况下每天更新 1 次，偶尔为 2 次；目标站点的新浪财经¹²会在每天的整点进行更新，类似地，考虑到为了尽可能的减少轮询扑空的现象，轮询时间可根据需求进行简单设置。

2.2.2 实现的难点与挑战

在系统实现的过程中，发现许多实现上的难点，不仅如此，系统所本身要求的高效性和稳定性也会对实现的过程带来相当大的挑战——

- 获取数据的稳定性。在获取数据方面，由于每个网站信息展现的方式大相径庭，如表 2.2，静态文本自然是最容易的，但也有 jQuery 渲染、GET/POST json 串甚至分段加载等，如何完整获取信息（见 3.2.3 节）成为了一大难题；有了信息获取方式还不够，现在的网站为了防止流量攻击，大都会设定同一个 IP 地址不得短时间发送过多请求，通过 Error403¹³拒绝的方式来保护自己。所以为了抓取的稳定，必须使得每次访问都使用不同 IP，通过设置代理池对访问抓取进行保障（见 3.1 节）。

几种不同的门户网站信息展现方式

表 2.2

站点	信息展现方式
新浪财经	分步加载+站内跳转（很容易出现抓取不完整的现象）
东方财富	接收用户的搜索请求后，返回 Json 串格式数据，需解析
同花顺	静态 HTML 网页，可直接通过 html 结构树针对正文抓取
慧博	时间轴方式显示最新资讯，可通过一直模拟点击“下一步”遍历

¹¹ <http://data.eastmoney.com/report/>

¹² http://vip.stock.finance.sina.com.cn/q/go.php/vReport_List/kind/search/index.phtml

¹³ <http://baike.baidu.com/item/403%E9%94%99%E8%AF%AF>

• 获取结论的高效性。使用算法和模型为人们服务，不仅仅把人们从繁琐而不具有创造性的任务中解放出来，看重的还有一个决定性的因素——速率。如图 2.2，每个点所对应的是我们所抓取的训练集中，资讯数量按月分布的情况（如 Aug-11 即 2011 年八月的数据）。可以看出，每年的 4 月与 8 月是研报发布集中期，这段时间内对研究人员的压力非常之大。本次研究的系统是为了辅助人们更快捷地处理信息，并为今后可能到来的人力不及的状态未雨绸缪。于是如何能够更快的处理信息无疑是面临的一项挑战。

• 训练模型的准确性。机器学习算法在训练模型部分至关重要，无论是对训练集的选择与处理（见 4.1 节），还是训练过程中对维度与迭代次数的把握（见 3.2.4 节）上。模型准确度越高，才越能帮助人们完成任务。难点主要集中在互联网上的海量数据的筛选与整合，除此之外，如何训练模型以提升准确度则是另一个需要重点关注的问题。



图 2.2 训练集中的金融研报按发布月的分布堆积图

• 资讯获得的时效性。一般来说，情报往往越早获得越有利，尤其是金融行业，情报直接决定了决策的拍定。我们的“实时轮询”功能便是为此而存在的：利用代理伪装（见 3.1 节）IP 地址，定时检测是否具有新资讯（见 3.2 节），当存在时立马获取并进行分析。这其中具有较大的困难，一个是很难把握轮询速度，过快会导致代理池迅速亏空，过慢又担心会延误太长时间；此外，构建一个可以低消耗休眠且能够快速激活的整体框架，是一项很大的挑战。

3 主模块的组成与实现

3.1 代理池模块/Retainer

需要说明的是，此处代理池模块 CDRetainer^[4]是个人初学爬虫的时候一时兴起做的一个开源辅助程序，期间断断续续地有更新过，后实习期间用于百度某项目中作为辅助模块使用，以 Github 发表时间、作者身份及百度大数据实验室该模块负责人身份，证明其原创性。

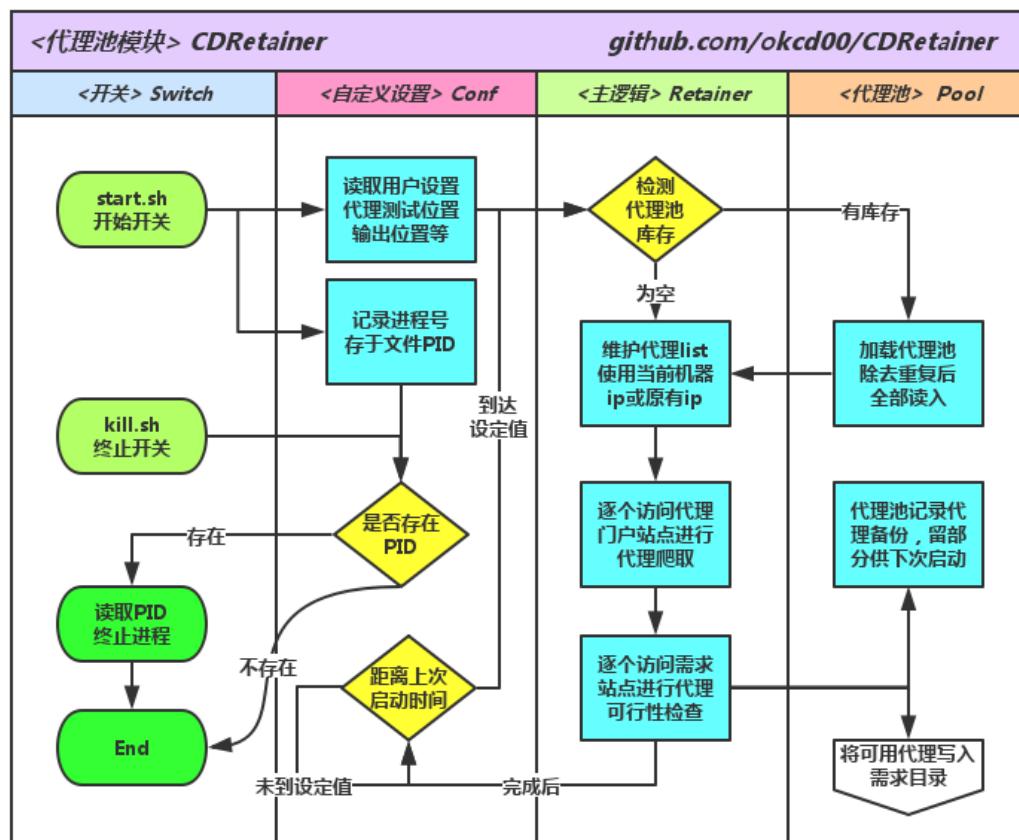


图 3.1 代理池模块 CDRetainer 的基本逻辑泳道图

代理池模块的作用主要是维护一个字符串数组（称之为代理池），其中每一个字符串是一个代理的地址和端口号。代理池模块仅仅依赖最简单的 urllib2¹⁴方式，如图 3.1，主要通过以下几个步骤维护代理池的可用性与新鲜度——

- 定时访问代理站点，抓取当前轮询周期（用户自定义，默认为 86400 秒）

¹⁴ <https://docs.python.org/2/library/urllib2.html?highlight=urllib2#module-urllib2>

中目标代理站点的最新代理项，判断代理具有可用性后加入池内，Retainer 中的用户自定义变量见表 3.1；

- 对当前代理池内的代理项进行去重，然后依次访问目标站点，将连接正常的代理字符串写入目标站点所对应的目录下，供负责目标站点的模块使用；
- 维持一个“New50（最新 50 条）”的列表，在每次 Retainer 启动的时候用作初始抓取种子代理来获取其他的代理，以及在其他模块申请但一时没有新的代理的时候可用作应急代理使用。

代理池模块用户自定义参数说明

表 3.1

变量	默认值	描述
path_home	“./..”	主系统的 HOME 目录位置
path_log	“./log/Log.txt”	Retainer 的日志记录位置
path_list	“./sourcelist.txt”	目标网站相关参数文件位置
USE_PROXY	0	是否使用代理来抓取代理
RUNTIME_WAITTIME	20	检测代理可用性时判断超时的阈值
EXEC_CYCLETIME	86400	Retainer 自启动周期
TESTALL	“http://www.baidu.com”	最初获得代理时用以判断可用性的站点

此外，目标站点等信息是以明文的形式直接存储在 sourcelist.txt 文本文档中的，形如“EastMod/data/new.txt@http://data.eastmoney.com/report/”的字符串，以@为分割，右边为目标站点，左边为目标目录。例如这条字符串，表达的意思则是：对于代理池内的每条代理，尝试访问“http://data.eastmoney.com/report/”，若访问成功，则在主目录下寻找“EastMod/data/new.txt”文件，在其末尾添加写入当前代理。

3.2 实时轮询抓取模块/Crawler

实时轮询抓取模块，顾名思义该模块的作用在于通过轮询检测以达到实时抓取最新数据的效果。抓取模块是逻辑复杂度最高、且重要性仅次于机器学习模型处理的模块。该模块比较繁琐，实现起来包含较多子模块，其中不免会涉及到一些依赖项，为助于理解，较为重要且常被提及的会简要的进行列举说明（见表 3.2）。

实时轮询抓取模块中所使用到的依赖项

表 3.2

依赖项	描述
bs4	全称 BeautifulSoup4 ¹⁵ , 用以将 HTML 文本解析为剖析树(parse tree)
Urllib2	Urllib2 是 Python 自带的用以获取 URLs 的组件
Goose.text	Goose ¹⁶ 是一个由 python 重写的页面提取开源库, 功能性较强
ConfigParser	用以读取 conf 文件, 实现用户便捷修改自定义参数的功能
cPickle	cPickle ¹⁷ 可以将变量完整的保存并能够完全可逆的恢复, 在本系统中用于生成记录点和出错时的回滚操作

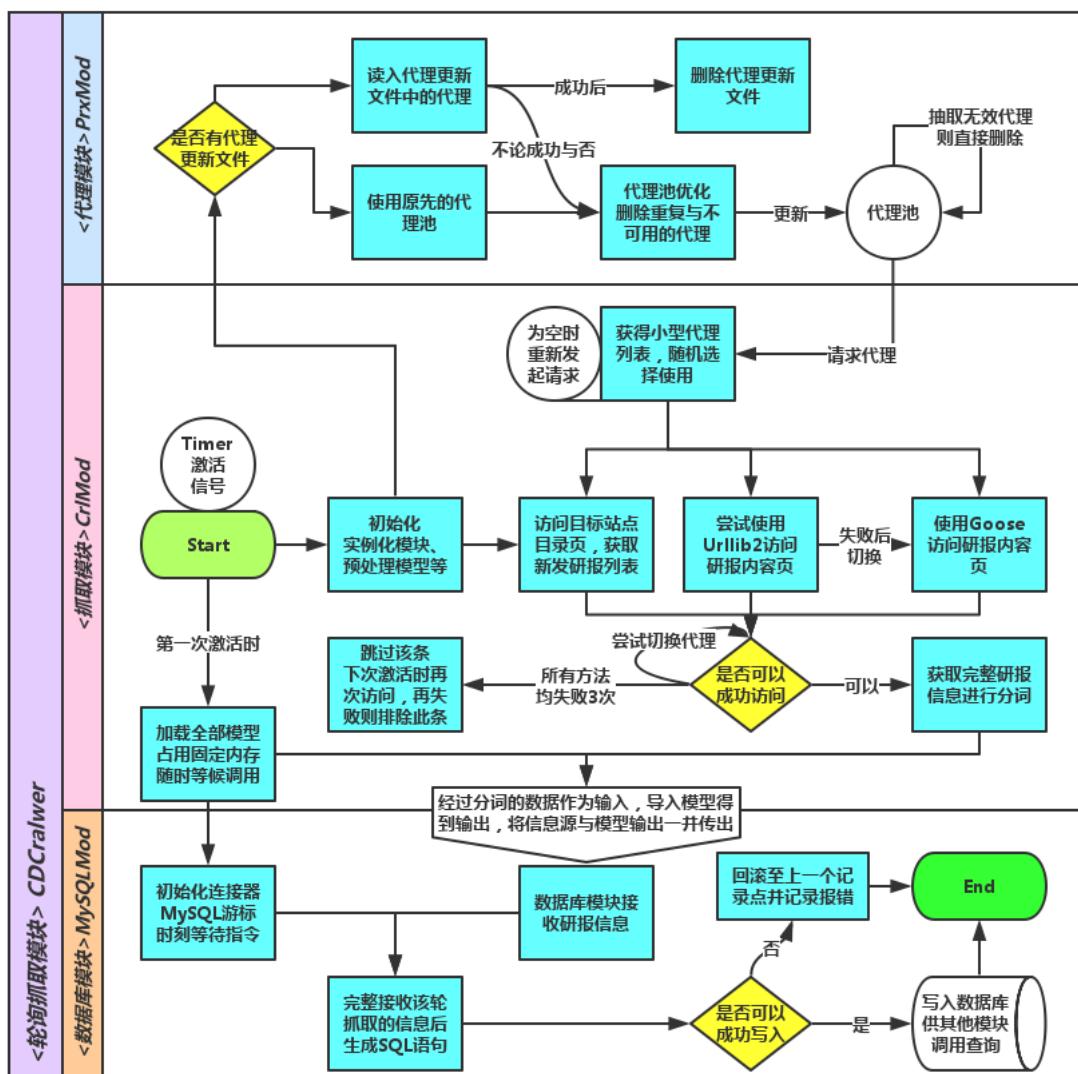


图 3.2 实时轮询抓取模块 CDCrawler 的基本逻辑泳道图

¹⁵ <https://www.crummy.com/software/BeautifulSoup/bs4/doc/>¹⁶ <https://pypi.python.org/pypi/goose-extractor/>¹⁷ <https://docs.python.org/2/library/pickle.html>

3.2.1 实时轮询模块/Timer

轮询操作由一个 Timer 负责控制激活信号。如图 3.2 中 Start 上方的激活信号生成器，除第一次启动外（第一次激活时需要较长时间初始化并实例化多个实体），每当计时器的读秒超过用户设定的轮询时间，则会发送激活信号开始新一轮。需要注意的是，第一次是指首次部署，当中断或因为严重报错而退出，再次开始时也会加载先前的进度继续，并不属于“第一次”的范畴。

实时轮询模块同时也是启动模块（Runner）。主要的自定义参量的读取也是在这个模块中完成的，调用 ConfigParser 将制定目录中的 conf 参量文件读入并解析，获得代码中各位置需要调用的关键参量。如表 3.1 中代理池模块的 conf 参量，其他模块中的参量在此不一一列举，但参量名的可读性还是相对比较强，感兴趣的读者可以在相关开源社区¹⁸自行阅读源码。

3.2.2 代理模块/PrxMod

抓取模块中的代理子模块主要负责在 Retainer 的代理池与 CrlMod 的抓取之间充当代理资源的输送桥梁，并维护子模块本身带有的小型代理池的可用性。如图 3.2.1 中，这是一个以激活信号为开始，持续维护代理池的模块。

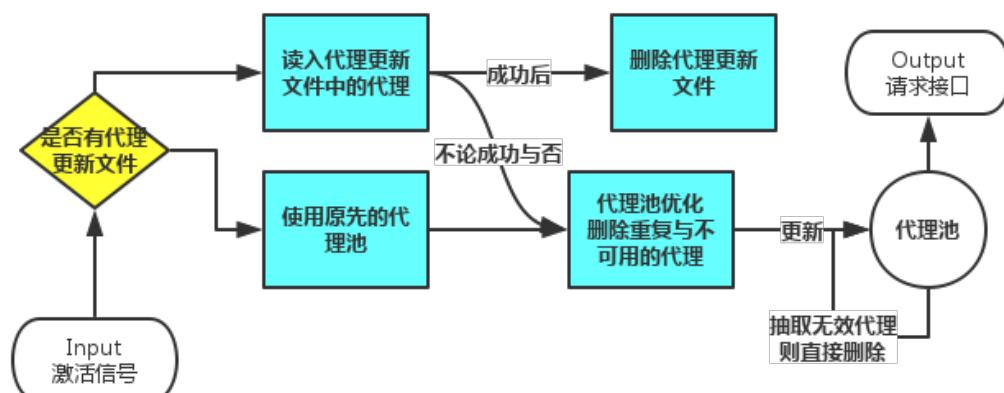


图 3.2.1 代理模块逻辑流程图

代理模块包含一个小型的爬虫功能，仅用于对当前模块的目标站点进行代理连通测试，出错的将被剔除出代理池；该模块同时具有回滚功能，在每次筛选与更新完之后，都会将代理通过 cPickle 在本地 Dump 生成一个记录文件，以备在断电断网等突发情况下终止进程之后，可以快速重启：读入记录点时的代理容器，以恢复功能继续进程。

¹⁸ <https://github.com/okcd00/CDReader>

3.2.3 抓取模块/CrlMod

抓取模块是所有模块中不确定因素最多的一个。所以抓取逻辑相对比较复杂，接下来如图 3.2.2，将针对各种实验过程中出现的问题，描述提升稳健性的方法与逻辑。

首先是从网页源码中进行文本的获取。默认抓取到网页源码之后使用 BeautifulSoup4 进行解析，成功的情况下可以解析为剖析树（Parse Tree），即将 HTML 代码按照由根到叶的从属关系生成一个树状结构，可以通过标签、Tag 名等来选择进入某一个子节点，所以我们可以通过 Selector 来事先选择好抓取正文的逻辑，如——

- 对于新浪财经，由于是跳转的方式进入落地页，有多种不同的用以寻找正文的选择逻辑，但在同一个站的逻辑也不会相差太远，大同小异。在此以其中一种进行举例：Selector_SinaF = “body > div > div.main.clearfix > div.ml > div > div.content > div.blk_container”；

- 而对于东方财富网，在 BeautifulSoup4 可用的情况下，经试验可以使用的是：Selector_EastF = “#ContentBody > div” 来作为正文逻辑的选择。

事先决定好的逻辑，会在每次获取到页面完整的 HTML 源码并生成剖析树后，根据上述的顺序依次进入对应名称的子节点中，逻辑走完之后通过预设的函数即可获得正文的文本。当 bs4 抓取失败时，我们会转而使用 Goose 进行抓取。

其次，获得文本之后需要对其进行预处理。获得文本之后，并不能直接使用。自然而然能够想到的是我们需要去除其中的样式（\t, \r, \b 等），此外由于我们之后需要把字符串转化成语料、MySQL 语句等其他形式，我们需要特别的将所有双引号去掉或者替换为单引号。

有了基础的获取文本途径，接下来需要在其上考虑更多优化的部分。首要考虑到的是轮询过程中杜绝重复的问题。在这里原先采取的是在 Output 时检测是否在库内来决定是否入库的方式，后来为了优化速率和减少数据库操作，采取了如下流程实现单条信息唯一入库：

1. 维护一个“已抓取信息戳”的列表，信息戳由日期、标题和作者唯一确定
2. 每次将已抓取部分的最新的不超过 *Thres* 个信息戳记录下来存于列表中
3. 抓取时仅判断当前条目是否匹配列表中的信息戳，不满足则抓取
4. 将入库的所有资讯和列表中原有的一起，列表更新不超过 *Thres* 个最新的

经过实验中的多次尝试与调研，我们发现，每日的更新量大多不超过 50，峰值时期不超过 200，且门户网站的每一页一般为 40 或 50 条，于是我们设定 Thres 的值取为 50。

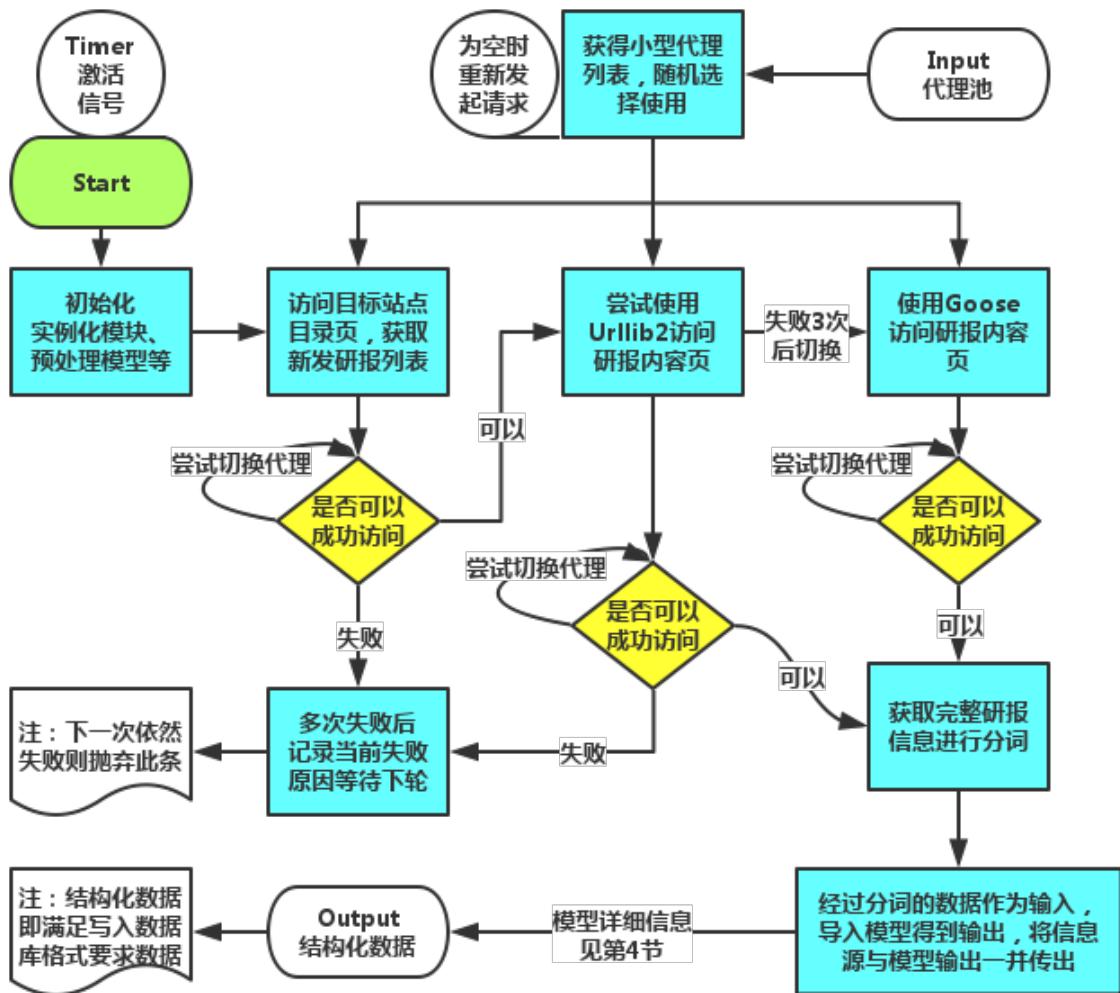


图 3.2.2 抓取模块逻辑流程图

稳健性不仅仅是体现在持续运行的耐久力，也要体现在崩溃之后的恢复力上。抓取模块中的几个关键的部分：已抓取信息戳列表、当前激活的代理池以及当前轮询的目标状态，它们在每次完成任务时都会通过 cPickle 生成 dump 文件，在出错重启时的初始化中，会 load 所有 dump 的记录点，最快速度回滚恢复到先前的状态。cPickle 是一种通过序列化储存和恢复变量的方式，也可以读写为二进制序列化文件，其好处是占用空间小、读入速度快（理论上还原参量最快的方法之一）。在系统中，多处恢复机制都是用 cPickle 来实现的。

3.2.4 数据存储/MysqlMod

数据存储模块是使用 MySQLdb 模块的 connect 方法连接 Python 程序与数据库（使用 MySQL），将结构化数据转化并编辑成 SQL 语句从而操作数据库的模块。

如表 3.3，数据库根据表中字段进行设置，根据考察研究目标的属性，经过多次的尝试和调整，最终决定了生成的 SQL 语句中选择这些普适性较强的字段：

CDReader 系统的 MySQL 数据库设计

表 3.3

字段/Field	数据类型/Type	能否为空	描述
Id	bigint(20) unsigned	NO	【主键】资讯越新则该值越大
Source	varchar(20)	NO	来源, 如 EAST、SINA 等
StockCode	varchar(10)	YES	该资讯所涉及的股票代码
StockName	char(30)	YES	该资讯所涉及的股票名称
CompanyCode	varchar(10)	YES	资讯来源的公司代码
CompanyName	char(30)	YES	资讯来源的公司名称
RateA	char(30)	YES	资讯研究对象原先的评级
RateB	char(30)	YES	资讯中对推荐更改的评级
RateC	char(30)	YES	资讯研究对象此后的评级
Author	char(100)	YES	资讯编写作者或机构的名称
Url	varchar(512)	YES	资讯落地页的网址
Title	varchar(512)	NO	该条资讯的标题
Text	longtext	NO	该条资讯预处理后的正文
Date	varchar(8)	NO	资讯发布日期/YYMMDD
Time	varchar(10)	NO	资讯发布时间/HH:MM:SS
LabelMood	varchar(30)	NO	情感倾向标签/0~1 之间浮点数
LabelRelate	varchar(1024)	NO	相关股票标签/股票名-相关系数
Trade	varchar(1024)	YES	相关行业标签/行业名-相关系数
Positive	varchar(256)	YES	情感正向词汇
Negative	varchar(256)	YES	情感正向词汇
Other	varchar(1024)	YES	其他 (备用)

值得一提的是, 由于该项研究在大数据环境下, 所以数据库模块的逻辑实现是允许分布式存储的, 前端也可以选择不同机器的不同数据库进行情报展现。

数据库模块中主要实现的三个功能, 连接数据库, 使游标一直在需要的位置待命¹⁹; 生成 SQL 语句后调用游标令数据库执行; 发生错误时 Rollback 恢复到提交该 SQL 指令之前。对于数据库模块而言, 这是一个规范、简洁而稳健的基本框架。

¹⁹ <http://blog.csdn.net/okcd00/article/details/50250263>

3.3 模型处理/Scikit

模型，在此处可以直接指代需要使用机器学习的算法学习出来的目标模型。模型本质上是一个结构复杂的计算框架（可以理解为一个复杂的方程或者函数），具有较高的纬度，给定符合该模型规范的输入之后将会给出输出。

接下来讨论的是针对研报文本的情感分析与分类模型的处理。

本节中，若无特殊指明，“学习”一词主要讨论“深度学习”^[5]相关领域，“模型”一词主要讨论“有监督机器学习”模型(Supervised Machine Learning)。

见图 3.3，以 WordSeg 的分词与 Word2Vec 的词向量数据为基础，我们使用了逻辑回归 (LR)、CNNTText 等模型或算法（详见表 3.4），我们将在本节进行说明。

涉及的机器学习相关模型（算法）描述

表 3.4

模型（算法名称）	描述
LR	逻辑回归模型（Logistic Regression）
CNNTText	卷积神经网络（CNN）文本分类模型
Liblinear	针对大数据的 LibSVM 优化版，用以解决线性模型的分类问题
KnowledgeableCNN	参加的开源项目 ²⁰ ，顾名思义是知识性卷积神经网络的构想

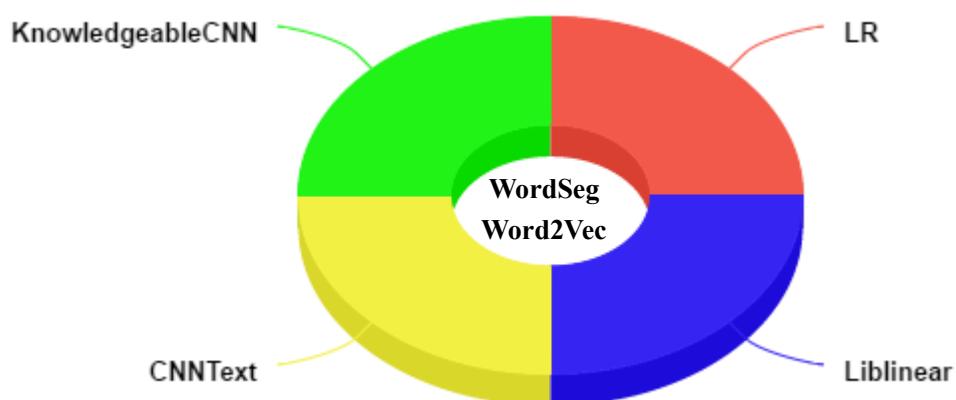


图 3.3 CDReader 系统中涉及到的机器学习相关模型（算法）

本系统中的模型选择与训练，主要依赖于当前适用性最广的 Python 机器学习组件 Scikit-Learn^{[6],[7]}，该组件是一个基于 NumPy, SciPy, 和 matplotlib 开发的开源

²⁰ <https://github.com/shockline/KnowlegeableCNN>

项目，可以通过预设的接口和函数大幅简化实现机器学习算法的操作。并且，开发者也实现了一些常用算法（如谱聚类²¹）的实例²²，在已有模型框架的基础上进行特化与修改，可以大大缩短开发周期和降低开发成本。

本节主要展现从输入到输出的模型训练研究过程中的开发要点。

3.3.1 分词处理/WordSeg

我们所获得的文本，即便是在预处理之后也并不能作为机器学习算法的输入，因为文本字符串是一个长度过长的完整元素，不但不易数字化表示（词向量见3.3.2节），而且当前计算机的处理效率和内存容量都不足以对如此大的元素进行处理。

所以我们需要进行分词处理——

分词，是指针对目标字符串进行的根据用户的关键词，使用各种匹配方法，将句子断成多个短词组（通常不超过5字）的一种技术，通常和Word2Vec配合使用，可以用高维度的浮点数组表示词组，是研究文本的机器学习算法中较为普遍的输入格式。

关于分词技术的选择，尝试过中科院的ICT-CLAS，盘古分词等，后考虑到系统轻量化、Python语言亲和度高和配置简单的优势选择了开源的JIEBA²³分词。在配置过程中，也研究了JIEBA分词的API，如表3.5，在分词处理的角度上也因支持自定义词库的导入从而²⁴大大提升了系统的可扩展性。

JIEBA分词效果示例

表3.5

原句	分词效果	备注
"徐天源是创新办主任 也是云计算方面的专家"	徐天源/是/创新/办/主任/ 也/是/云/计算/方面/的/专家/	JIEBA分词 直接分词效果
"我爱北京天安门"	我/爱/北京/天安门	同上
"徐天源是创新办主任 也是云计算方面的专家"	徐天源/是/创新办/主任/ 也/是/云计算/方面/的/专家	添加自定义词典 后的分词效果

3.3.2 训练词向量/Word2Vec

词向量²⁵，普遍被认为是将深度学习带入自然语言处理领域的核心与关键技术，其在模型训练中扮演的重要角色无可替代。T. Mikolov等人在其论文^[3]中提出，

²¹ http://scikit-learn.org/stable/auto_examples/bicluster/plot_spectral_coclustering.html

²² http://scikit-learn.org/stable/auto_examples/index.html

²³ <https://github.com/fxsjy/jieba>

²⁴ `jieba.load_userdict(filename)` # filename is the custom dictionary.

²⁵ <http://licstar.net/archives/328>

训练得到的高维浮点数组向量，可以代表词汇的隐含特性，令词汇之间可以通过计算词向量的距离来获得相关性的多寡，如图 3.4 中，可以看出训练好的模型中，词向量距离相近可以一定程度上代表词义相近。除了可以快速输出特定词的相关词外，模型也支持查询特定两个词的词向量距离，这对于自然语言的处理无疑是领路人一般的贡献——当自然语言的词汇具有向量的特性，就可以将文本数据看作其他类型的变量一样，沿用深度学习的多种算法来寻找各种有趣的特性。

Enter word or sentence (EXIT to break): 宝马		
Word	Cosine distance	
奔驰	0.719989	
奥迪	0.673603	
轿车	0.654061	
别克	0.652714	
丰田	0.614717	
本田	0.613870	
新车	0.611007	
旅行车	0.610864	
华晨	0.608774	
豪华轿车	0.603684	
斯柯达	0.601588	
雅阁	0.600212	
夏利	0.599699	

图 3.4 Word2Vec 效果展示²⁶

谷歌公司的 Mikolov 等人使用 Skip-Gram 算法实现^[8]的 Word2Vec（此处我们禁用了 Negative-Sample 开关，故本文的讨论将忽略此部分），是用来将语料库训练输出词向量模型的便捷工具。同时，他们认为，连续词汇之间的组合隐含着语义规律^[9]，研究出由输入层到隐层学习权重的向量空间词表示。

词向量的起源应该是 Hinton 在 1986 年的论文中^[10]这个概念首次提出，2000 年百度 IDL 的徐伟将“词向量”展现于国人，后在 Bengio 的 FFNNLM 论文^[11]中，被发扬起来，而确实在 Word2vec 的开源之后，它才真正被广大学者所熟知。

使用词向量概念构建自然语言模型方面，Bengio 等人^[11]构建的三层神经网络训练语言模型（见图 3.5）常被称之为经典，这个语言模型用作文本预测（已知前 n-1 个词预测第 n 个词的概率），较之普通 n-gram 算法的效果要好 10%~20%（使用 APNews 数据集测试）。详见表 3.6，该模型的三层分别为简单拼接的输入层、偏置后由 tanh 激活的隐藏层和 softmax 激活的输出层，这份模型对于 Word2Vec 生成的词向量是一种非常成功的应用，同时对于我们的模型具有重要的启发作用。由于本次设计中的系统是受此启发基于词向量来进行自然语言的文本分析，第四节中

²⁶ <http://blog.csdn.net/memray/article/details/12562027>

会详细说明本次设计中的模型处理，此处不再赘述。

Bengio 等人论文中所实现的语言模型 Layer 描述

表 3.6

层名 (Layer)	操作	备注
输入层	将输入的(前 n-1 个)词向量(m 维)依次首尾拼接起来	(n-1)*m 维
隐藏层	偏置向量后使用 tanh 激活函数进行激活 ²⁷	tanh 即双曲正切函数
输出层	Softmax 函数激活，输出 V 维，对应出现概率	出现概率未做归一化

我们使用 Word2Vec，对同花顺财经网站²⁸中 2011 年 8 月至 2015 年 8 月间全部包含评价与评级的研报²⁹，即总共 338,580 条沪指资讯，322,929 条深指资讯，共 1653.44M 字节的数据制作成语料库训练词向量模型。Word2Vec 训练主要参数及简要说明见表 3.7。

CDReader 中 Word2Vec 训练使用的主要参数

表 3.7

参数名称	取值	描述
cbow	0	CBOW 算法开关，0 为关闭，即使用 Skip-Gram 算法训练
size	200	词向量纬度设置，经测试此处设置为 200 结果更加贴切真实
window	5	滑动窗口大小，每次以窗口进行滑动取样，该值决定单条样本的容量
negative	0	负采样开关，0 为关闭，同时 hs 参数为 1，即转而使用 HS 方法
sample	1e-3	采样阈值，即出现得越频繁的词汇则越容易被采样
threads	12	线程数，需要注意的是，线程数会一定程度上影响模型结果
binary	0	输出的模型文件是否以二进制进行存储，此处关闭，以备后续读入浮点数

²⁷ tanh 激活可以使得函数收敛最快²⁸ <http://10jqka.com.cn/>²⁹ http://search.10jqka.com.cn/snapshot/report_pdf/17c5311a01530f61.html 【研报示例】

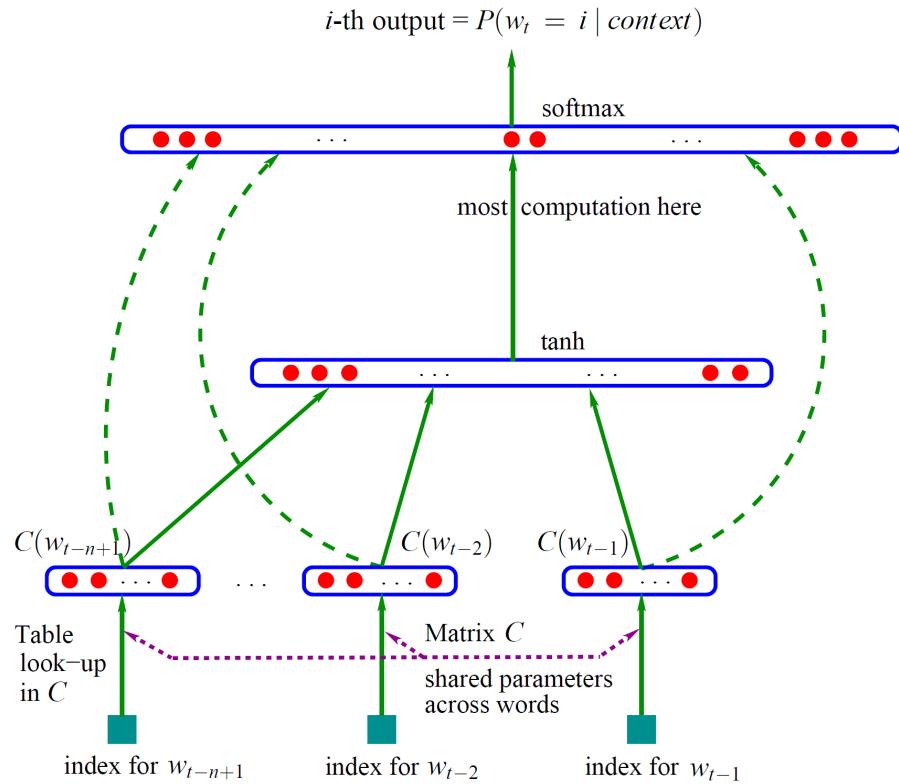


图 3.5 Bengio 提出的用以语言模型构建的神经网络

3.3.3 知识性卷积神经网络/KnowleageableCNN

知识性卷积神经网络 (Knowledgeable Convolutional Neural Networks)，是中科院一名研究生³⁰在其论文中提出的构想，并开源其实现思路。KnowledgeableCNN 的目标是：在具有相对准确词向量的基础上，使用卷积神经网络，对变长的文本进行知识性的判断（知识性与非知识性文本）与评分。

了解到该模型时，本设计的研究进展到仿效 Johnson R 等人的方法，使用卷积神经网络处理文本数据^[12]，则联系作者加入其开源项目，在之后的开发中提供想法、优化修改了部分代码³¹。理解掌握并获得原作者同意之后，将项目中所需要运用在 CDReader 系统的机器学习模型部分提取并进行了重构³²与特化，类比转化成用于情感分析并标记情感倾向最高的文本。

在最开始，需要对卷积神经网络 (Convolutional Neural Network, CNN) 进行说明，这是一种通常用于处理图像信息的神经网络算法，如下图(图 3.6 为对于 CNN 的介绍中，使用最广泛的教科书式论文及应用实验)中，用以处理手写图像的像素信息，来识别图像所对应的字符。CNN 主要通过其 local receptive fields (感受野)，

³⁰ Ganbin Zhou ; Mailto: zhouganbin@ics.ict.ac.cn

³¹ <https://github.com/shockline/KnowleageableCNN/graphs/contributors>

³² <https://github.com/okcd00/CDReader/tree/master/Theano>

shared weights (共享权值), sub-sampling (下采样) 的三个特性来解决问题, 篇幅所限不再赘述基本概念。唯一需要说明的是, CNN 对于图像的处理同样适用于变长文本的训练与信息提取^[12], 这是模型建立的基础。

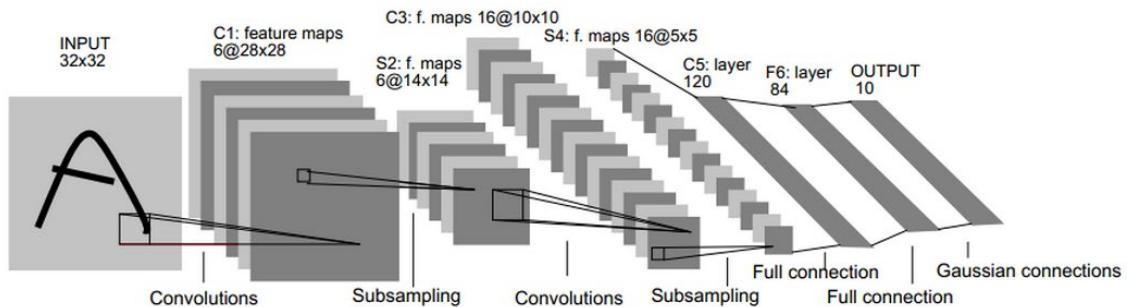


图 3.6 卷积神经网络的在 LeNet-5^[13]中的应用（手写体识别）

我们借用 Shockline 的 KnowledgeableCNN 模型初始架构 (见图 3.7) 来简要描述一下特化为本系统使用的 KnowledgeableCNN 的模型结构, 对本系统的 Layers (分层) 进行简要概括。

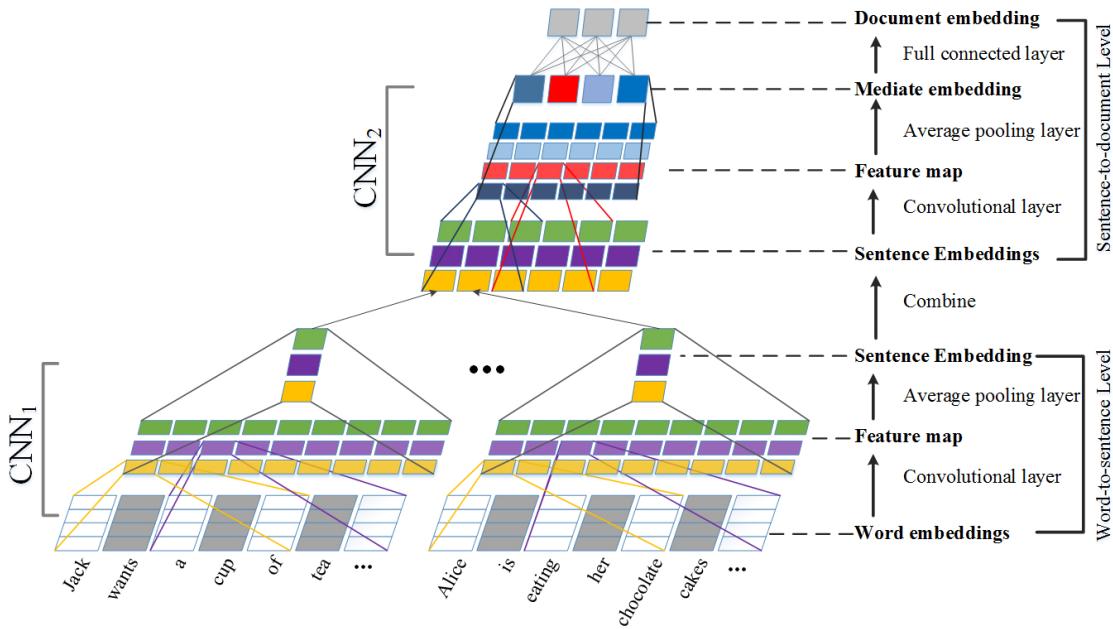
本项设计中的模型架构可称之为“Word-Sentence-Paragraph”(词-句-段) 模式。但与现在 NLP 领域的“Sentence2Vec”³³和“Paragraph2Vec”^{34,35}有所不同, 这两种模型是试图将句段也构造成高维向量, 使句段也可以具有词向量一样的特性与应用。考虑到这两种模型还在完善中不够成熟, 无法达到 Word2Vec 可以达到的级别。在 CDReader 的系统中仅需要对一个情感倾向的值进行估计 (输出纬度和范围都相对较小), 并不需要类似获取文档间相似度的高要求。此处的“词-句-段”模式的操作, 是在输入层的 Word2Vec 基础上作卷积(Convolution)然后池化(Pooling, 图像处理算法中的常见方法³⁶)成下一层的输入, 隐含层在基础的偏置激活操作以外依然是进行一次卷积与池化, 输出层获得该项文本的词向量 (常称作 Embedding), 从而得到所需的数据 (模型算法与数据类目的说明将于第 4 节作详细说明)。

³³ <https://github.com/klb3713/sentence2vec>

³⁴ <https://github.com/cemoody/Document2Vec>

³⁵ <http://www.jianshu.com/p/d34d61188ab5>

³⁶ <http://blog.csdn.net/zhoubl668/article/details/24801103>, 卷积与池化为图像处理算法常用方法, 作为基础概念如要进行解释, 文本较长, 限于篇幅贴一个易于理解的博客, 不作过多赘述。

图 3.7 KnowledgeableCNN 的模型结构³⁷

3.3.4 其他机器学习算法

在本系统中，KnowledgeableCNN 处于尚未证明其可解释性的测试阶段。考虑到增强系统的完整性以及传输更多高密度信息的需求，除此以外，也内置了目标为图 3.8(A)效果的用以预测情感倾向的线性回归（Logistic Regression, LR）模型、目标为图 3.8(B)效果的用以分析资讯相关股票的 Liblinear 模型，以及目标为图 3.8(C)效果的用以分析与资讯相关行业的 CNNText 模型。



图 3.8(A) 情感倾向

图 3.8 (B) 关联股票部分

图 3.8 (C) 关联行业部分

³⁷ 架构设计图来源于模型作者，依照卷积神经网络架构设计绘制

3.4 前端展示模块/CDPage

为了更好的展示抓取模块获得的文本及模型输出等成果，一个可视化效果强的展示界面是必需的。考虑到跨平台和开发难度，决定了用 PHP 生成网页的形式来做页面前端。由于大多终端（PC、移动端甚至嵌入式终端）都配备了可以加载网页的浏览器，所以前端采用页面展示，一定程度上也实现了前端的跨平台访问。



图 3.9 前端 PHP 主页面

3.4.1 通过 connector 连接数据库

在调用数据库数据之前，准备工作需要加载 PHP 中链接数据库的部分，定义目标机器地址、用户名密码、端口号与时区，配置好连接之后模块化为 **connect.php** 供其他前端页面作依赖项调用。

然后需要实现的是 json 请求模块，当每次请求时，我们需要知道当前请求的是第几组数据，GET 方式获得组数之后，在数据库里 Select 出组数所对应的（每组 10 条资讯），将这十条数据按照 Key-Value 的形式封装成 JSON 字符串传出，功能实现后模块化并命名为 **result.php**。

初始访问到的主页面，如图 3.9，默认初始会显示最新 1 组的数据，当每次响应用户需求（页面滚动到底部 20%）时，会再次发送请求，以获取下一批（组）

数据，当数据库被遍历完（ID1 被返回）后会反馈告知用户“没有更多数据”。

此外，在点击“阅读全文”后，会再次调用数据库查询点入资讯的 ID，获得完整数据后会自动生成 PHP 落地页，供用户查看资讯全文及系统给出的各项阅读辅助。

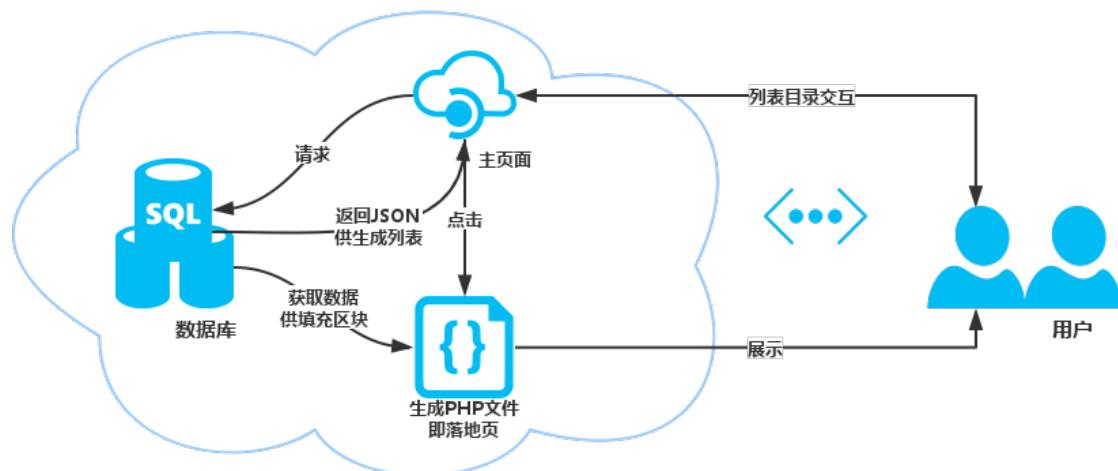


图 3.10 “前端—数据库”获取数据逻辑

3.4.2 前端 php 展示模块

展示模块主要分为两个部分，目录列表与落地详情页。

关于目录列表，上节中说到的滚动至页面底部时获取新信息，我们定义当前滚动比例为滑动条滑动到主页面的 body 部分末尾 20%时激活数据库查询并调用。此外，数据在目录列表（即主页）上是以越新的越靠前的方式排列，可以近似的看成一个资讯的时间轴。见图 3.9，每一个资讯的实体中包括来源图标（如新浪）、涨跌情况（涨跌小熊图片，见图 3.11）、资讯标题、来源（见图 3.9 及图 3.11）及部分正文预览。当用户遇到感兴趣的资讯，可以点击阅读全文，将跳转至详情页。



图 3.11 涨跌状态图 及 资讯来源为“东方财富网”的实体对象

关于落地详情页，见图 3.12，是一个将资讯相关的全部数据与模型结果一并展现给用户的可视化界面。

页面最上方，是资讯的标题及作者名称（有时是资讯来源公司的名称），其下

则是完整的资讯正文内容，此处的文本样式较为单一，由于最初对数据进行预处理时，为了防止字符串传递过程中发生错误，去掉了所有样式相关的字符，对此其实可以采用替换为特殊字符序列而非删除的方法，显示时再替换回来即可，该思路留作今后系统更新改进的考虑点之一。页面的右方则是阅读辅助栏，系统中的机器学习部分得出的结果均在此展现在用户面前：咨询中包含的关键词（正向情感与与负向情感均具有关键词）；资讯中涉及关联的股票名称（同时也标注出了相关度，方便区分关联性的强弱）；资讯中涉及到关联的行业；以及，一个 0 到 1 之间的浮点数，用以表示该条资讯的综合情感倾向，我们将这个值大于 0.5 的视作看涨，反之视作看跌（不看涨的中立情感也归于此类）。

分析及建议

隧道股份-600820

正向词VS负向词
利好 增长 优势

关联股票-关联度
隧道股份-0.0015
中国中铁-0.0014
上海建工-0.0014
中国铁建-0.0013
北京城建-0.0009

关联行业-关联度
医药生物-0.2179
纺织服装-0.139
商业贸易-0.1069
机械设备-0.0701
电气设备-0.0467

综合建议: 0.999172

图 3.12 落地详情页示例图（关键词版本）

3.5 日志模块/LogMod

日志模块的实质，其实是实现了一个在程序运行过程中，将一部分输出重定向到预设的文件中的效果。此处的 LogMod 特指本系统中对 sys 库中的 logging 进行重载包装而做成的新模块，普遍用于本系统中一切需要长时间稳定运转的部分。

对于一个对稳健性要求很强的系统，日志模块是必不可少的，由于进程会持续运转极长时间（如本系统的实测评估则是持续了三个月的无人值守运行测试），我们无法时刻监测其运行动态。特别是类似模型训练算法的长时间运算，开发者在调试过程中甚至无法判断当前是正常计算中还是卡在死循环等程序错误中，如图 3.13(A)，在日志模块的帮助下，即便是控制台不作任何输出的运行阶段，也可以时刻监测到当前动向：如 19:24:05 时 Theano 正在为模型创建层，至 19:24:09 时

完成创建并开始读取第一组训练集数据。

```

D:\GitHub\CDReader\Theano\log\Log.txt - Notepad++
文件(F) 编辑(E) 搜索(S) 视图(V) 格式(M) 语言(L) 设置(T) 宏(O) 运行(R) 插件(P) 窗口(W) ?
MysqlMod.py run.py Log.txt
342 05-11 19:20:55 [Notice] getCorpus at getDataMatrix()
343 05-11 19:24:05 [Notice] Using Model[train], Data[cdr], PoolingModel[average_exc_pad]
344 05-11 19:24:05 [Notice] Function Work Starts.
345 05-11 19:24:05 [Notice] Start Constructing Layers.
346 05-11 19:24:09 [Notice] Layers Constructed.
347 05-11 19:24:09 [Notice] Now Starts Training with data/cdr/train/text_sh000 and data/cdr/train/
348 05-11 19:24:09 [Notice] Start Loading Data[h000]
349 05-11 19:24:10 [Notice] 6338 Labels Loaded
350 05-11 19:24:14 [Notice] 6338 Documents Loaded
351 05-11 19:24:14 [Notice] 6 Stop-Words Loaded
352 05-11 19:24:28 [Notice] w2v model Loaded, contains: 544206 elements

```

图 3.13(A) 模型训练模块日志节选

```

MysqlMod.py run.py Log.txt structureTest_Training.py MultiTraining.py Train.log
91 05-11 22:20:09 [Notice] Now Saving parameters.
92 05-11 22:20:09 [Notice] Parameters Saved.
93 05-11 22:20:09 [Notice] Now batches at 8-th epoch.
94 05-11 22:43:14 [Notice] Current validate Model[8]
95 ROC: 0.78273299028
96 TPR: 0.993710691824
97 FPR: 0.969696969697
98 AR: 0.512006861063
99 threshold: 1
100 05-11 22:43:14 [Notice] Now Saving parameters.
101 05-11 22:43:14 [Notice] Parameters Saved.
102 05-11 22:43:14 [Notice] End Training Data[h000]
103 05-11 22:43:14 [Notice] Now Starts Training with data/cdr/train/text_sh000
104 05-11 22:43:15 [Notice] Start Loading Data[h001]
105 05-11 22:43:15 [Notice] 6338 Labels Loaded
106 05-11 22:43:19 [Notice] 6338 Documents Loaded
107 05-11 22:43:23 [Notice] getCorpus at getDataMatrix()
108 05-11 22:43:29 [Notice] Data has been loaded already
109 05-11 22:44:07 [Notice] Valid current model: Cost,
110 ROC: 0.78273299028
111 TPR: 0.993710691824
112 FPR: 0.969696969697

```

图 3.13(B) 模型训练模块日志节选

4 模型算法及辅助模块

4.1 通过 Pyspider 平台获取训练集数据

Pyspider，是一个开源项目，异常强大的网络爬虫系统。其带有强大的 WebUI，采用 Python 语言模式的 PyQuery 代码，分布式架构，支持链接多种数据库后端。在此，我们使用 Pyspider 来为系统的模型训练获取训练集的语料。

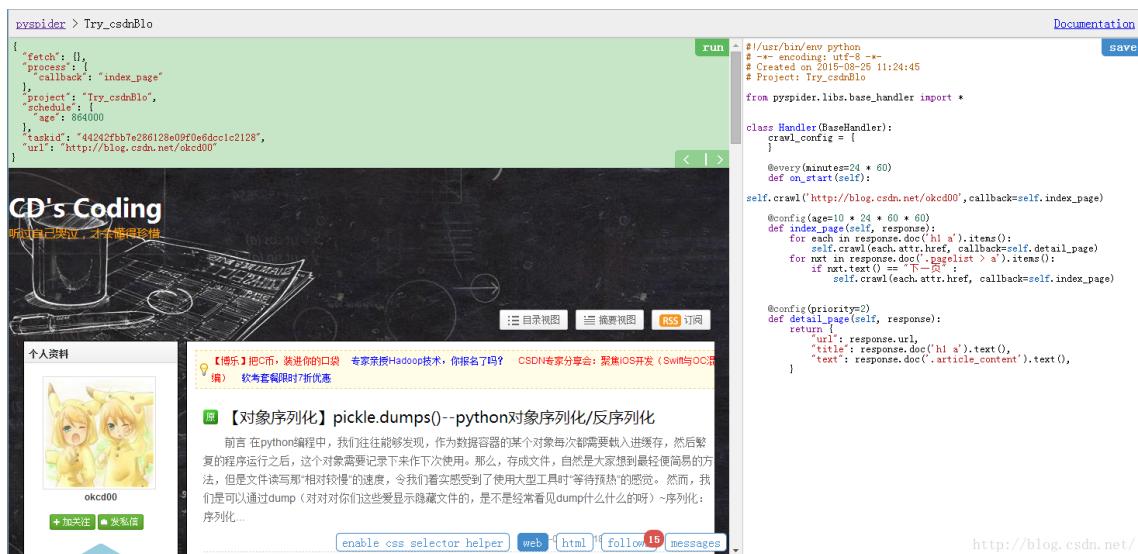


图 4.1(A) 使用 Pyspider 的 css selector 辅助抓取



图 4.1(B) 使用 Pyspider 的 css selector 辅助抓取

关于 Pyspider 抓取，略微有些心得，也写过相关的技术博客³⁸。在制作训练集数据时，采用图 4.1 中的方式筛选出目标资讯站点中需要抓取的标签（即图 4.1(B) 中的红框），就可以使用选择的 selector 在如图 4.1(A) 右侧编写抓取代码，同花顺财经站点中抓取逻辑并不复杂，但是种类较多逐个获取 selector 编写代码比较繁琐，此处抓取逻辑不一一列举，若参考，代码均已开源有兴趣可自主前往查看³⁹。

4.2 通过 Theano 进行 GPU 训练加速

Theano 是一个 Python 的第三方库，因为可以使用它有效地定义，优化和评估涉及多维数组的数学表达式，所以通常 Theano 可用于训练模型。此外，Theano 另一个被广泛运用的特性则是 GPU⁴⁰ 加速，Theano 可以将数据全都转化成浮点数，然后存于显存之中，利用显卡对矩阵运算的高效性，如表 4.1，能够有效地加速运算过程。

同任务下 CPU/GPU 运算效率的对比

表 4.1

目标	CPU 耗时 (sec)	GPU 耗时 (sec)
求 1000 次 e 的随机次幂	2.6071999073	【GTX 275】 2.28562092781
预训练纬度为 1050 的单层神经网络	3330	【GTX 970】 228
构建五层神经网络的全局调优过程 ⁴¹	10926	【GTX 970】 378

4.3 模型训练的设置

模型的训练包括模型的搭建、训练集的准备、人工定义参数、调参、训练和验证。对于一个机器学习算法的实现效果，其中每个步骤都可能对结果造成影响，所以耐心仔细的面对每一个步骤，是降低模型误差率的关键。

本系统的模型设置为三层的深度学习神经网络，在丁效等人提出^[14]的关于构建“事件驱动的股票预测”神经网络，以及“结构化数据预测股票走势”^[15]的理论基础⁴²上，我们尝试使用海量研报语料训练出的词向量关联性，来维护一个以“资讯情感”为参考目标的参数集合——意思是，在当前系统中，参数相似的实体代表的并非“语义”相近，而是“情感”相近。我们的 目标是分析当前文章的情感（在金融研报中，情感特指对当前研究目标是否看涨）。系统的三层模型在 Theano 下训练时，

³⁸ <http://blog.csdn.net/okcd00/article/details/47975375>

³⁹ https://github.com/okcd00/CDReader/blob/master/Pyspider/ReportList_sh10jqka.py

⁴⁰ http://deeplearning.net/software/theano/tutorial/using_gpu.html

⁴¹ <http://blog.csdn.net/m624197265/article/details/45700619> 五层的维度分别为 [784, 1050, 4901, 500, 10]

⁴² 此处我们并非确定可以通过此方法预测股市，理论依据指的是我们由此知道了应该从什么角度辅助阅读

基本参数设置如表 4.2:

训练过程的重要预设参数（终稿）

表 4.2

参数名	设定值	描述
Pooling_Mode	average_exc_pad	共包含 ‘max’, ‘average_inc_pad’, ‘average_exc_pad’ 三种池化模型，经逐一测试选择效果最好的模型
activation	T.tanh	最常用的激活函数之一，具有最快收敛的性质
wordEmbeddingDim	200	词向量维度，测试[50,100,200,500]后选择效果最佳值
n_epochs	8	单训练集循环训练次数，通常在第 8 次不再明显变化

训练过程分多段进行，但是每段数据容量大、持续时间久、训练进度慢，在使用日志模块监测的同时，需要一个可以无人值守、自动训练、同时还能提升训练效率的逻辑来帮助执行，于是有了重构后的 **MultiTraining** 脚本。如图 4.2，将训练集的 Text 与 Label 一一对应后分割成 19 组沪指训练数据与 20 组深指训练数据（每 8,888 条资讯分为一组），两者均抽出两组数据作为验证集。模型文件单次加载持续更新，每次更新后都 dump 出序列化文件，以备随时回档。

训练过程由 Log 日志记录⁴³，单次激活完整稳定训练完全部输入的训练集(取 18 个沪指与 19 个深指数据)，训练过程如表 4.3，参数学习部分详见 5.1 节。

使用 Theano 进行模型训练的耗时情况

表 4.3

训练集	样本数量 ⁴⁴	样本容量	开始时间	结束时间	耗时
h000	7370	1771610	19:35:27	22:43:14	3:07:47
h001	7379	1760451	22:43:15	01:53:02	3:09:47
h002	7332	1757620	01:53:02	05:04:49	3:11:47
.....					
h017	7351	1746129	00:56:09	04:07:25	3:11:16
z000	7553	1806839	04:07:26	07:29:45	3:22:19
z001	7583	1837432	07:29:45	10:52:03	3:22:18
.....					
z018	7597	1824088	12:46:11	15:22:07	2:35:56

⁴³ <https://github.com/okcd00/CDReader/blob/master/Theano/Train.log>⁴⁴ 样本数量为输入的 8888 行中满足训练集要求（如文章不得少于 5 句等）的文本数

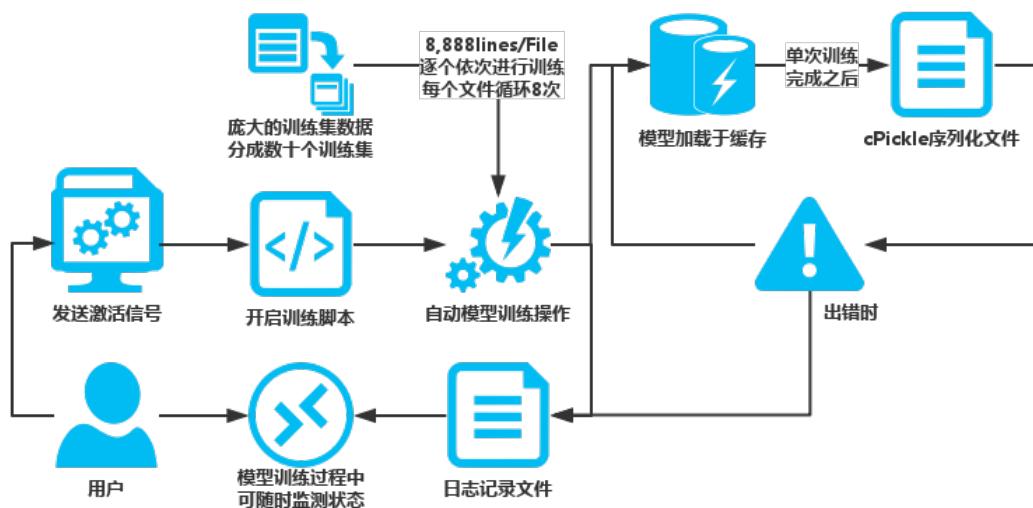


图 4.2 重构后的模型训练逻辑

5 实测评估

5.1 训练过程中的参数自学习

模型训练过程中，通常需要通过一些评判指标来评价模型的表现。本系统的训练中选择了如下四个较为常用的评判指标：ROC、TPR、FPR 和 AR。

	True matches	True non-match.
Pred. matches	TP = 18	FP = 4
Pred. non-match.	FN = 2	TN = 76
P = 20	N = 80	
TPR = 0.90		FPR = 0.05

图 5.1 TPR、FPR 的定义

受试者工作特征 (Receiver Operating Characteristic, ROC) 通常用来评价一个二元分类器；二元分类问题中，我们通常有正类(Positive)与负类(Negative)，如图 5.1，当正确地预测正类时，我们称其为真正类 (True positive)，如果实例是负类被预测成正类，称之为误报率 (False positive)。其中，两列 True matches 和 True non-match 分别代表应该匹配上和不应该匹配上的，两行 Pred matches 和 Pred non-match 分别代表预测匹配上和预测不匹配上的。

由上述两条引出一个定义，准确率(Accuracy Rate, AR)，用以评价综合准确率，其计算公式为：

$$AR = [TPR + (1 - FPR)] / 2$$

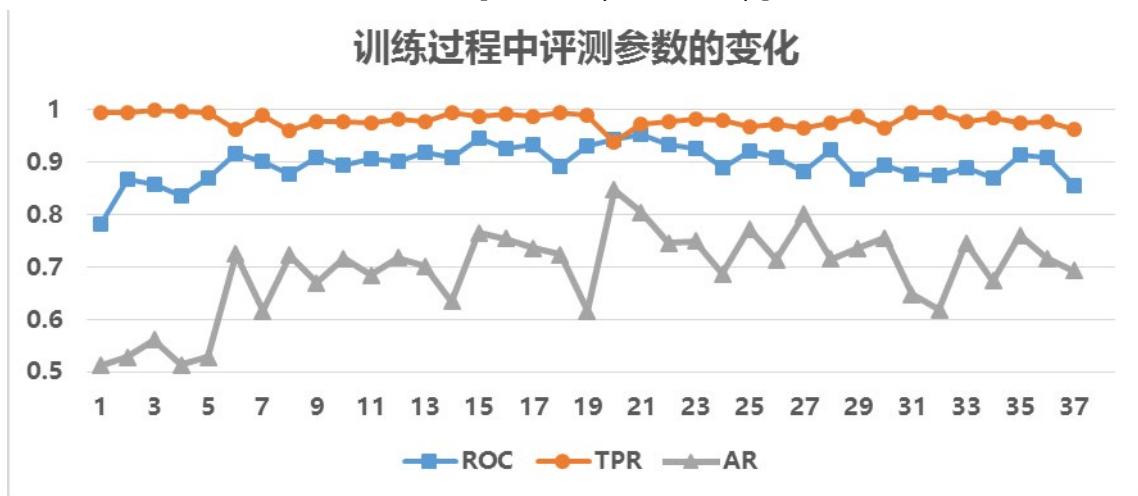


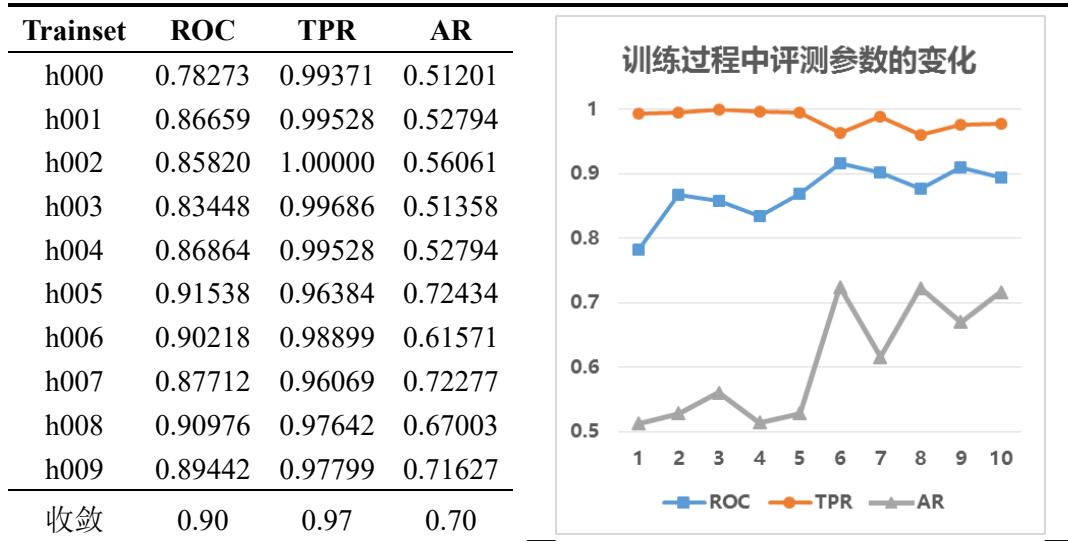
图 5.2 完整训练过程中主要评测参数的变化情况

我们将准备的庞大数据集（使用分布式平台 Pyspider 批量抓取，详见 4.1 节）分为数据条数 10:1 的两份，多的那份作为训练集，少的那份作为验证集。其中，由于采用的是随机梯度下降(Stochastic Gradient Descent, SGD⁴⁵)的迭代求解思路，可以采取(Mini-Batch⁴⁶ SGD)的方式将训练集分批处理。

我们按照每 8,888 个资讯样本为一组，分作 37 个小型的训练集，每次读入其中的一个训练集后，便通过验证集来获取计算 ROC、TPR、AR 所需要的各项指标参数，通过日志在每次训练后计算各项评判指标，并记录下来。如图 5.2 中完整的训练过程中参数变化情况。通过观察不难发现，在使用前 10 个训练集训练模型的时候各项参数开始达到基本收敛了，我们将前 10 次训练的数据拿出来（见表 5.1）进行简单分析，可以更加明显地看出训练成效。

使用前十个训练集训练过程中的参数状况⁴⁷

表 5.1



5.2 实测运行评估及回馈评价

自系统雏形完成之后，为了检测评估系统各方面的性能，进行了几次实测运行。一次是为了评估系统稳健性，在初期进行的无人值守持续抓取测试，如图 5.3，一直成功稳定抓取无重复的中文研报数据（单位：行）。我们有趣的发现，每周的周六和周日、以及法定节假日期间大多是没有获取新研报的，这也与实际情况也相符（其中 12 月 12 日有一条 4 行的研报，经核查也是因为前一天研报访问错误，

⁴⁵ <http://baike.baidu.com/item/随机梯度下降>

⁴⁶ batch GD 的每一轮迭代需要所有样本参与，对于大规模的机器学习应用，经常有 billion 级别的训练集，计算复杂度非常高。而训练集是数据分布的一个采样集合，我们可以在每次迭代只利用部分训练集样本

⁴⁷真正类率(true positive rate ,TPR)，计算公式为 $TPR = TP / (TP + FN)$

假正类率(false positive rate, FPR)，计算公式为 $FPR = FP / (FP + TN)$

所以在第二天方才获取得到，以及图 5.3(B)中春节期间也没有研报数据获取），其中每一条数据如图 5.3(C)所示（截图未完全，还包括 ID 等 3.2 节中提及的字段）。



图 5.3(A) 系统实现初期（2015/11/05 至 2015/12/31）爬取数据稳定性实测

12 月 31 日的中断，同时是对断点恢复的测试。该日更改了抓取逻辑并更新了模型文件，重启系统，自动加载终止进程时的代理池、记录戳等，继续了先前的任务，继续进行了三个月测试（图 5.3B）。展现了模块化的优势——机器工厂模式，易于组合、分解、修改与更换，每个模块只需要负责自己的功能与对外接口即可。

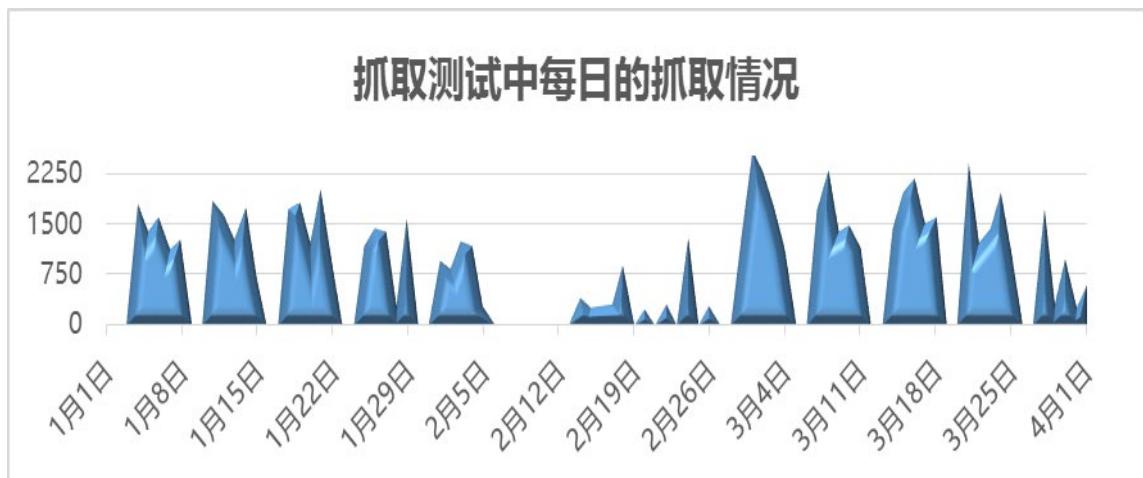


图 5.3(B) 系统断点重开阶段（2016/01/01 至 2016/04/01）爬取数据稳定性实测

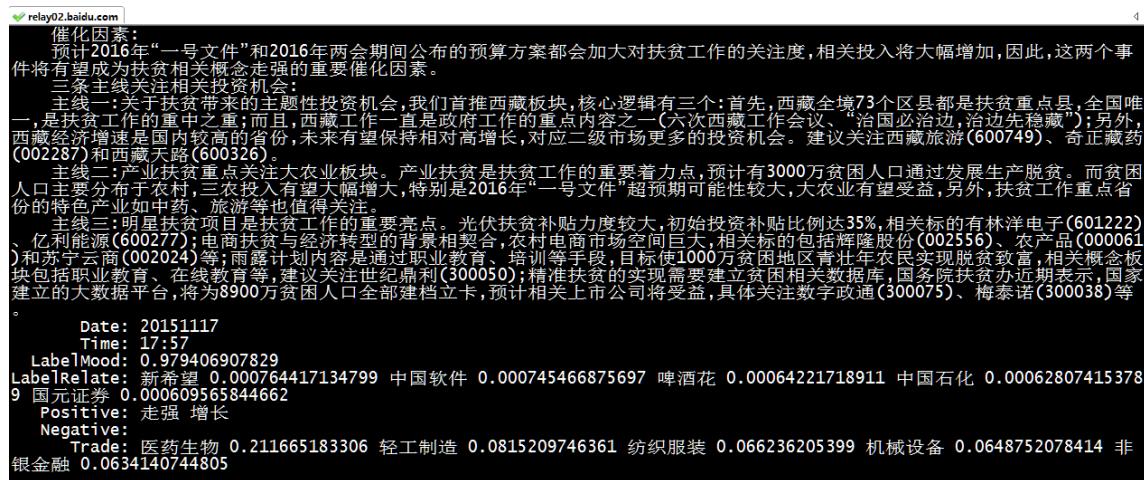


图 5.3(C) 获取数据示例

```

mysql> select count(*) from StockDataAL where Source like 'East';
+-----+
| count(*) |
+-----+
| 2680 |
+-----+
1 row in set (0.10 sec)

mysql> select count(*) from StockDataAL where Source like 'Sina';
+-----+
| count(*) |
+-----+
| 20166 |
+-----+
1 row in set (0.00 sec)

```

图 5.3(D) 不同来源获取的数据数量比例概览

6 可扩展性

6.1 面向其他算法的扩展

算法扩展在模块化的系统中非常容易，在启动模块(Runner)中的模型加载部分，当分词完毕获得语料 Corpus 之后，逐个对各种算法类传入，并将算法输出结果记录下来。当需要添加算法时，只需要自行将新算法模块化后，在启动模块的初始化部分创建实例，在模型加载部分添加一行当前算法的调用接口即可。

6.2 面向其他数据集或领域的扩展

现在是数据为王的时代，在机器学习领域自然也不例外——在一个算法一直优化到无法再提升性能的瓶颈期，一份更好的语料库或训练集往往能发挥奇效。如果需要增加其他数据集，在系统中也充分做好了考虑：在训练文件目录中，Train 文件夹放入文本数据及标签，分别以 Text 和 Label 为文件名前缀且除前缀以外对应的文本与标签名必须一致，系统训练时会自动读取使用；Test 文件夹放入同等要求的少量数据，训练过程中会作为验证集使用。

CDReader 系统最初的设想便是一个阅读辅助，由于对可用性的展示与说明必定需要一个目标，于是选择了实用性与难度并存的金融领域。实际上，如图 6.1 的应用中可以看出，同样的模型在其他领域的应用也是有实用性与可行性的。



图 6.1 KnowledgeableCNN 在探究科普类文章知识性语句应用中的表现

7 结论

本文针对当前资讯产生速度过快、专业性较强的问题，实现一个可以实时获取最新数据，分析关键情感并提取重要信息，通过数据库存储与管理，最终采用 PHP 页面展现给用户的完整资讯采集分析系统。

经过大量的优化与增改，系统基本完成了稳健的实时轮询抓取，稳定的数据库传输读写，高效的模型加载计算，以及跨平台的展示功能的实现。其中，无人值守的代理池模块，与在其基础上的自动轮询抓取是一个比较亮眼的创新点，引入了一个还在发展过程中的开源深度学习算法也是一次不错的尝试。

在完成毕业设计的过程中，首次接触到如此多的开源项目，结识到许多也努力维护和开发自己的灵感的开源项目开发者们。在提升抓取模块的性能与增添新功能上，开源项目的 Goose、Jieba、Pyspider 等都扮演着不可或缺的角色，学习的同时也开始为自己曾经的项目添砖加瓦，在这期间获得了相当的锻炼，并认识到开源社区对于开发与技术发展的重要性。本次设计是第一次涉足前端开发和模型训练，学习新鲜事物的过程虽然艰辛但非常具有成就感。

由于想要将功能实现得尽可能的强大，也自行在设计中增加了许多尝试，即便是每个模块都尽可能详尽地考虑了，但毕竟没有足够的时间进行压力测试和恶意样本测试等，难免会出现没有考虑周全的部分。此外，在前端与建模的实现上也具有一些瑕疵：前端部分在展现信息的模块逻辑略微有些混乱；模型加载部分中，在从神经元中获取尽量完整信息的逻辑还没有完善妥当……这些可以作为今后改进与扩展的着手点，让系统的功能变得更加完善与可依赖。

致 谢

本次毕业设计是大学以来所做的规模最庞大的一次设计。涉及多个领域，其中过半都是从未涉足过的崭新事物，之所以能够最终完成，是因为这其中有许多无私地为我提供帮助的人。在此，对他们的帮助表达真挚的感谢。

最初的选题与方向选取方面，必须要感谢我的校外导师罗平⁴⁸老师。罗平老师为我的毕设选题提供了好几种设想，并且在我选择了需要学习大量新技术才能实现的这个课题时，也给予了极大的支持与肯定。在遇到不知如何解决的算法难题和实现时，罗平老师都会给出机器学习算法或者模型的推荐——参考文献中，几乎一半的论文都是罗平老师筛选出来推荐我去阅读学习的，此外，重要的论文也会督促我看懂吃透，如附录 A 中的演示文档就是学习过程中所做的笔记。

课题进展过程中，必须要感谢我的校内导师涂凤华老师⁴⁹。在我校外毕设期间，对毕设的各项要求都不明确，涂凤华老师耐心地帮我指出毕设过程中不正确和不规范的地方，而且在完成设计的过程中，遇到难题时也对我进行了鼓励与支持。

此外，非常感谢校外实习期间，为我提供大数据开发环境的实习公司——百度时代网络技术有限公司。校外毕设期间，我刚好由于拿到 OFFER 在**百度研究院的大数据实验室**作为数据实习生学习工作，于是可以在工作之余为自己的毕设努力：百度的 ODP 环境为简易部署前端网页提供便捷；工作环境的开发机的 RedhatOS 不仅省去了安装虚拟机的繁琐，其性能（主要是缓存、内存与运算速度）也是其他机器所不能比拟的；用以部署 pyspider 的对外机器具有极大的带宽，分布式平台的抓取能力在大带宽的配合下极大缩短了获取语料库的时间周期。在这期间，大数据实验室的各位前辈与老师⁵⁰也为我解答了疑惑，提供了数不清的指点。非常感谢百度，让我可以在如此的高度，为自己的大学交上一份满意的答卷。

当然，除此之外，由于实习期间无法返校，帮助我为各项繁琐的文件与材料来回奔波的舍友杨烨宇；开源项目 KnowledgeableCNN 作者，为我亲切的讲解并允许我加入一同深入开发的 Shockline（原名周干斌）；帮助我进行机器学习入门的中科院的学长闫肃；时时嘘寒问暖的亲人们……还有很多很多，在这期间我确实得到了很多人的帮助，在此对你们表示无比由衷的感谢。

谢谢你们！

⁴⁸ http://sourcedb.ict.cas.cn/cn/jssrck/201405/t20140504_4108331.html

⁴⁹ <http://www.cs.cqu.edu.cn/public/szdw/index>

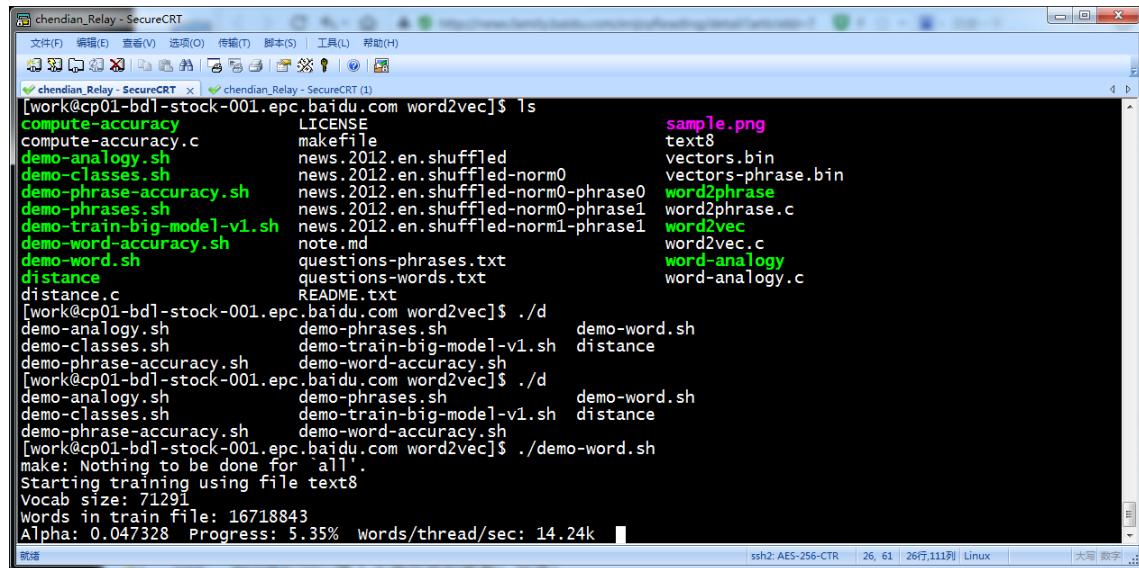
⁵⁰ Wang Kai, Shen Zhiyong, Liu Zhiqiang, Zhan Zhizheng, Wang Qing, Hu Wei etc.

陈 点

参 考 文 献

- [1] 王军,刘金辉; .大数据的国内外研究现状及发展动态分析[J]. 电子技术与软件工程,2015,23:200.
- [2] P. Luo, S. Yan, Z. Liu, Z. Shen, S. Yang and Q. He. From online behaviors to offline retailing. In KDD, 2016.
- [3] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In NIPS, 2013.
- [4] Dian Chen. CDRetainer In Github 2015. <https://github.com/okcd00/CDRetainer>
- [5] Andrew Ng.. 深度学习将为人工智能带来新机会[N]. 中国信息化周报,2015-04-06007.
- [6] Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011.
- [7] API design for machine learning software: experiences from the scikit-learn project, Buitinck et al., 2013.
- [8] Mikolov T, Chen K, Corrado G, et al. Efficient Estimation of Word Representations in Vector Space[J]. Computer Science, 2013.
- [9] Mikolov* T, Yih W T, Zweig G. Linguistic regularities in continuous space word representations[J]. In HLT-NAACL, 2013.
- [10] Hinton G E. Learning distributed representations of concepts[C]// In Proceedings of CogSci. 1986.
- [11] Bengio Y, Schwenk H, Senécal J S, et al. A neural probabilistic language model[J]. Journal of Machine Learning Research, 2003, 3(6):1137-1155.
- [12] Johnson R, Zhang T. Semi-supervised Convolutional Neural Networks for Text Categorization via Region Embedding.[J]. Computer Science, 2015, 28:919-927.
- [13] Lécun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11):2278-2324.
- [14] Xiao Ding, Yue Zhang, Ting Liu, Junwen Duan. Deep Learning for Event-Driven Stock Prediction. In Proc. IJCAI 2015
- [15] Xiao Ding, Yue Zhang, Ting Liu, Junwen Duan. Using Structured Events to Predict Stock Price Movement: An Empirical Investigation. In Proc. EMNLP 2014
- [16] Ji, Shihao, Satish, Nadathur, Li, Sheng, et al. Parallelizing Word2Vec in Shared and Distributed Memory[J]. 2016.

附录 A：系统相关信息及基本概念



```
[work@cp01-bd1-stock-001.epc.baidu.com word2vec]$ ls
compute-accuracy LICENSE
compute-accuracy.c makefile
demo-analogy.sh news.2012.en.shuffled
demo-classes.sh news.2012.en.shuffled-norm0
demo-phrase-accuracy.sh news.2012.en.shuffled-norm0-phrase0
demo-phrases.sh news.2012.en.shuffled-norm0-phrase1
demo-train-big-model-v1.sh news.2012.en.shuffled-norm1-phrase1
demo-word-accuracy.sh note.md
demo-word.sh questions-phrases.txt
distance questions-words.txt
distance.c README.txt
[work@cp01-bd1-stock-001.epc.baidu.com word2vec]$ ./d
demo-analogy.sh demo-phrases.sh demo-word.sh
demo-classes.sh demo-train-big-model-v1.sh distance
demo-phrase-accuracy.sh demo-word-accuracy.sh
[work@cp01-bd1-stock-001.epc.baidu.com word2vec]$ ./d
demo-analogy.sh demo-phrases.sh demo-word.sh
demo-classes.sh demo-train-big-model-v1.sh distance
demo-phrase-accuracy.sh demo-word-accuracy.sh
[work@cp01-bd1-stock-001.epc.baidu.com word2vec]$ ./demo-word.sh
make: Nothing to be done for 'all'.
Starting training using file text8
Vocab size: 71291
words in train file: 16718843
Alpha: 0.047328 Progress: 5.35% Words/thread/sec: 14.24k
```

图 F1 Word2Vec 训练词向量模型时的效率（每秒每线程 14.24k 词）

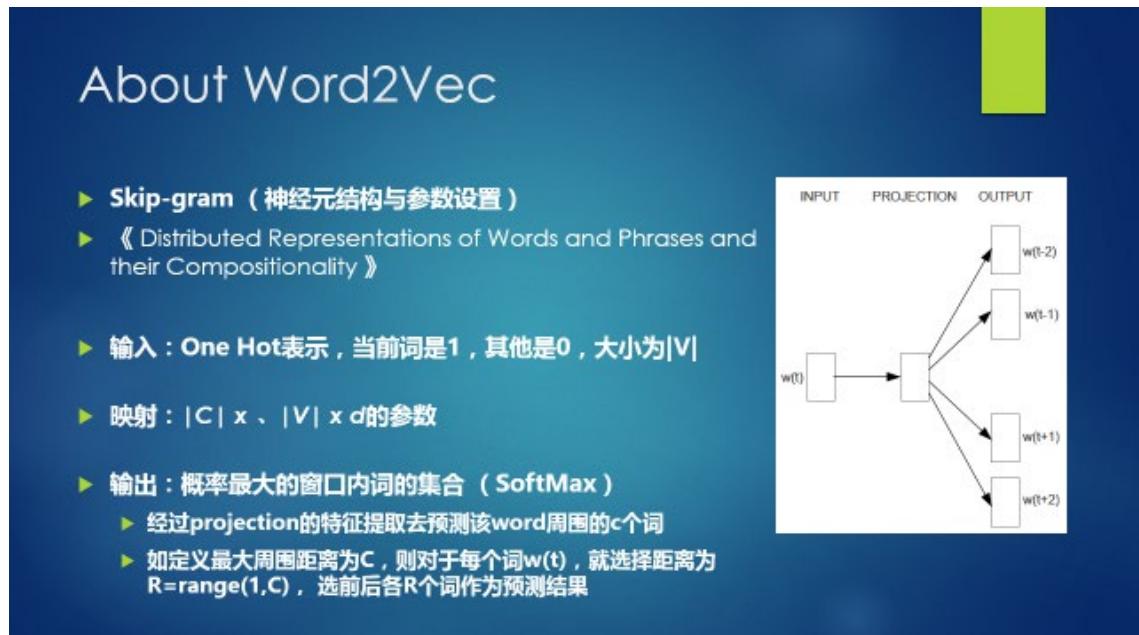


图 F2(A) Word2Vec 的 Skip-gram 模型⁵¹（自制演示文档）

⁵¹本质上是一个动态的逻辑回归

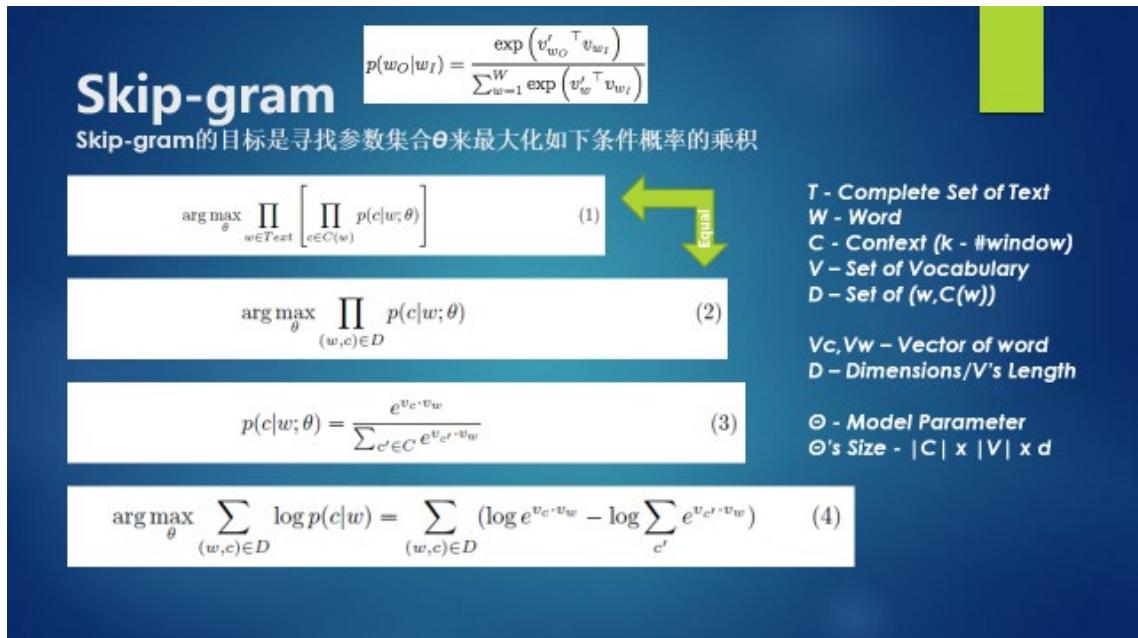
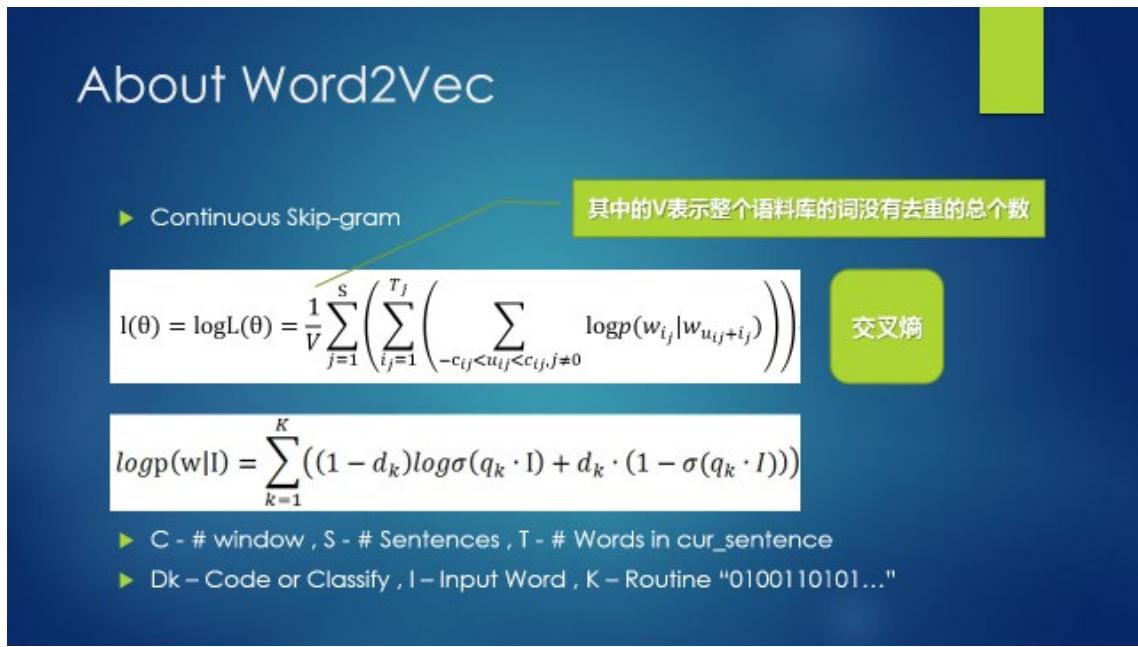


图 F2(B) Word2Vec 的 Skip-gram 模型表示（自制演示文档）

图 F2(C) Word2Vec 的 Skip-gram 模型目标⁵²（自制演示文档）

⁵² http://blog.csdn.net/mytestmy/article/details/26969149?utm_source=tuicool&utm_medium=referral
解法：随机梯度下降算法更新

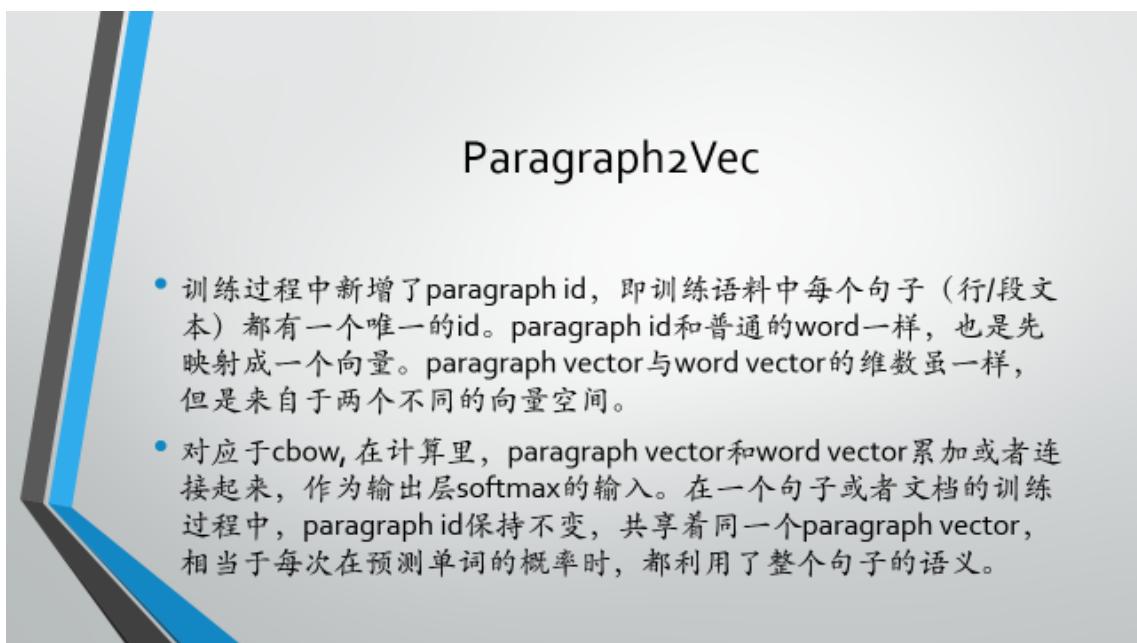
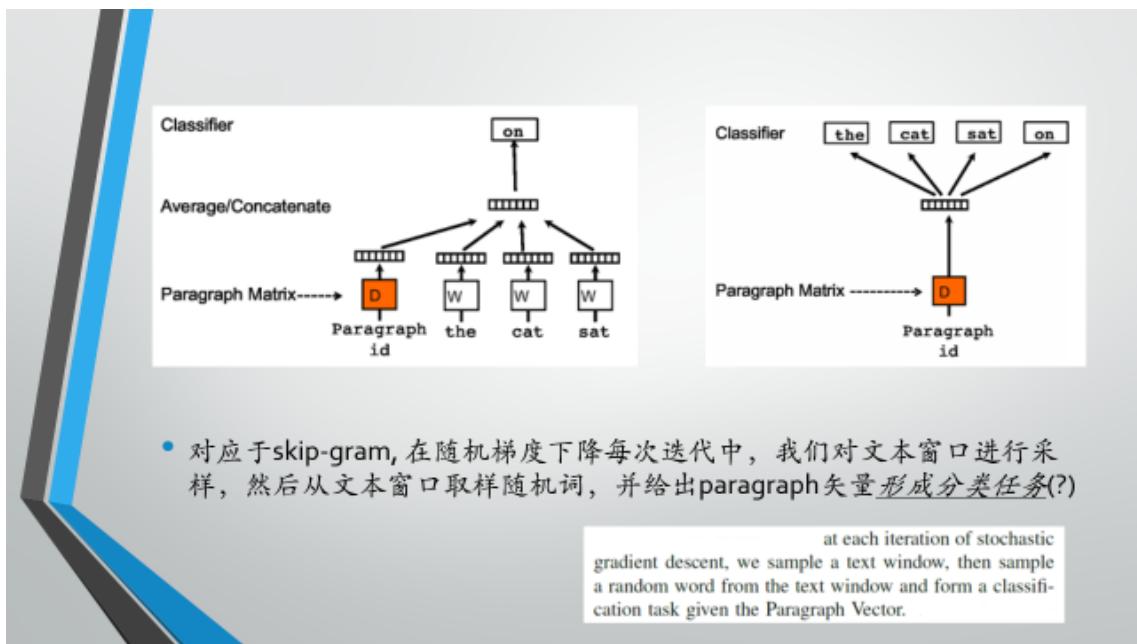
图 F3(A) 提及的 Paragraph2Vec⁵³ (自制演示文档)

图 F3(A) 提及的 Paragraph2Vec (自制演示文档)

⁵³ <http://www.jianshu.com/p/d34d61188ab5>

Paragraph2vec 是一种非监督学习方式，输入为文本，输出则是文本对应的向量表示。
http://cs.stanford.edu/~quocle/paragraph_vector.pdf

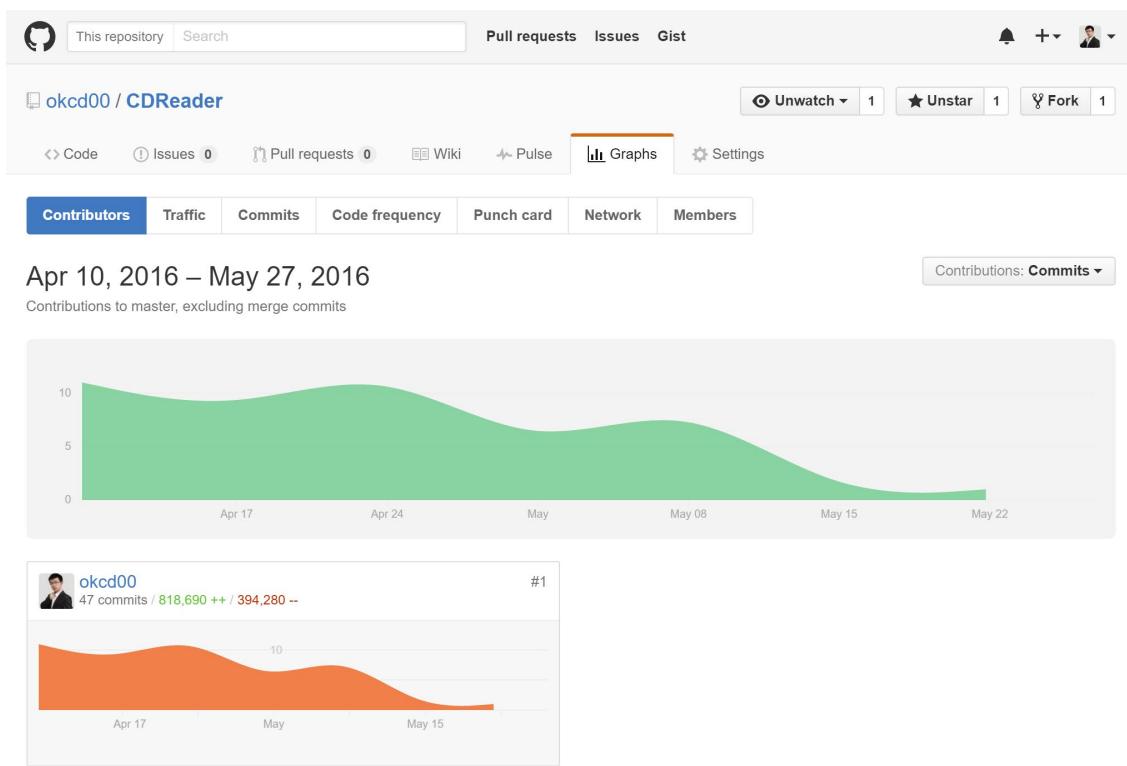
Algorithm 1: Event Embedding Training Process

Input: $\mathcal{E} = (E_1, E_2, \dots, E_n)$ a set of event tuples; the model $EELM$

Output: updated model $EELM'$

- 1 random replace the event argument and got the corrupted event tuple
- 2 $\mathcal{E}^r \leftarrow (E_1^r, E_2^r, \dots, E_n^r)$
- 3 **while** $\mathcal{E} \neq []$ **do**
- 4 $loss \leftarrow max(0, 1 - f(E_i) + |f(E_i^r)| + \lambda \|\Phi\|_2^2)$
- 5 **if** $loss > 0$ **then**
- 6 Update(Φ)
- 7 **else**
- 8 $\mathcal{E} \leftarrow \mathcal{E} / \{E_i\}$
- 9 **return** $EELM$

图 F4 参考文献[14]中提及的事件嵌入训练过程伪代码

图 F5 自 CDReader 系统的 Prototype⁵⁴开源以来进行的增改情况

⁵⁴ Prototype 即原型，首次上传时即已经达成全部开题报告所涉及的基本要求。