doi:10.16104/j.issn.1673-1891.2021.02.015

基于 Hadoop 平台的主题概念股票挖掘系统应用研究

丁　俊

(安徽工业经济职业技术学院计算机与艺术学院,安徽 合肥 230051)

摘　要:针对目前资本市场上快速挖掘某种主题概念股票的需求,提出了一种新思路,该思路以上市公司的核心题材、主营收入和资本运作 3 项数据为基础,进行主题概念相关指数的分析和计算,最终以此指数作为标准推荐主题概念相关股票,并开发了一套数据抓取程序和 Web 应用程序。数据抓取程序利用定时组件 Quartz 从各大财经网站抓取全体上市公司已公开的各类基本信息,存入分布式文件系统 HDFS 中;Web 应用程序接收用户输入的查询关键字组合,系统利用抓取的数据集从公司收入、投资和核心概念 3 方面分析和计算出公司与用户需要查询的关键字组合的相关指数,最后汇总为总相关指数,总相关指数越高的公司,其相关度越高,相关度越高的公司越有可能就是用户想要查找的相关主题概念公司。通过这 3 方面的结合,在公司的过去和未来,在定性和定量等多个方面都进行了相关度的挖掘,从而计算出来的相关性将更加可靠、准确。

关键词:数据抓取;Hadoop;主题概念;股票挖掘;相关指数

中图分类号:TP311.13;F831.51　　**文献标志码**:A　　**文章编号**:1673-1891(2021)02-0082-07

Application of Thematic Concept Stock Detecting System Based on Hadoop Platform

DING Jun

(College of Computer and Art, Anhui Technical College of Industry and Economy, Hefei, Anhui 230051, China)

Abstract: In response to the demand of promptly detecting thematic concept stocksin the current capital market, this paper proposes a new approach which analyzes and calculates the correlated index of the theme concept based on the data of the core concept, main business income and capital operation of the listed companies. The outcome of the calculation provides a standard for selecting thematic concept stocks. This paper also develops a data capture program for catching various basic information from all listed companies and saving the data in the distributed file system HDFS with timing components Quartz, and a Web application program which receives the query keyword combination from users and figures out correlated index of the query keyword combination between the demand users of and that of companies in terms of the company's income, investment and core concept. At last, the program aggregatesall related index into the total correlation index. The higher the total correlation index is, the higher the correlation degree is, the more likely the company is to be the related thematic concept company that users want to search for.Through the combination of the three aspects, correlative degree is determined by the past and future of the company through qualitative and quantitative assessments, therefore the calculation is more accurate and reliable.

Keywords: data capture; Hadoop, thematic concept; detecting stocks; correlation index

0 引言

目前,中国的资本市场还不够成熟和完善,股票市场上热衷于题材和概念的炒作行为依然盛行。对于市场中诞生了一个新题材或者新概念,普通投资者想要快速挖掘出该概念或题材相关的股票是一件非常繁杂的事务——需要浏览大量的公司信息,而且随着资本市场的进一步扩大、上市公司数量的不断增加、技术的进步,新题材新概念更是层出不穷,这种挖掘难度将会与日俱增。

收稿日期:2020-08-09

基金项目:安徽省高校自然科学研究重点项目(KJ2019A1049);2020 年安徽省级精品线下开放课程《WEB 程序设计(JSP)》(2020kfkc130)。

作者简介:丁俊(1979—),男,安徽肥西人,副教授,硕士,研究方向:软件开发、大数据。

目前市场上已经出现了一些主题概念股票挖掘的应用和研究,不过大多是基于文本层面的挖掘应用。通过采集股票核心题材资讯、股吧评论等文本信息,再构建主题热度因子,综合考虑主题行业热度和主题概念热度 2 个方面来描述主题和个股之间的关系,构建多因子量化选股模型,并利用逻辑回归模型进行主题概念股票挖掘[1]。由于文本对信息的描述局限于定性层面,缺少量化数据的精确性,所以,这类仅仅基于文本数据挖掘出来的相关主题概念股票,其真实相关度存在很大的不确定性。

本文针对这一现实问题,提出了一种新思路,在原有的股票核心题材资讯、股吧评论等文本信息的基础上,引入了个股的主营业务收入和资本运作 2 项数据作为挖掘的基础数据集。这 2 项数据是公司在主题概念方面实实在在的收入和投资数据,收入是公司在主题概念方面已经实现的,代表了公司过去与主题概念的相关性;而投资是公司在主题概念方面即将投入的资本,代表着公司未来与主题概念的相关性。通过这 3 方面的结合,在公司的过去和未来,在定性和定量等多个方面都进行了相关度的挖掘,从而计算出来的相关性将更加可靠、准确[2],克服了之前仅依赖文本数据进行主题概念股票挖掘存在的弊端。

本文为了对这一新思路开展进一步的应用性研究,还开发一套数据抓取程序和一套 Web 应用程序,数据抓取程序利用定时组件 Quartz 从各大财经网站抓取全体上市公司已公开的各类基本信息,存入分布式文件系统 HDFS 中[3];Web 应用程序接收用户输入的查询关键字组合,系统利用抓取的数据集从公司收入、投资和题材 3 方面分析和计算出公司与用户需要查询的关键字组的相关指数,最后汇总为总相关指数。总相关指数越高,相关度越高,相关度越高的公司越有可能就是用户想要查找的相关主题概念公司,这也克服了目前大多股票交易软件只能通过固定的概念板块划分来寻找某个概念板块股票集合的限制。

1 系统的总体架构设计

系统主要完成从数据抓取、分布式存储、数据预处理到数据分析查询和数据展现等一整套数据业务处理流程,实现基于真实数据基础上的主题概念股票挖掘功能[4]。系统主要包括 3 大子系统:Hadoop 服务器集群存储系统、数据采集系统、数据分析查询和展示系统,系统整体架构如图 1 所示。

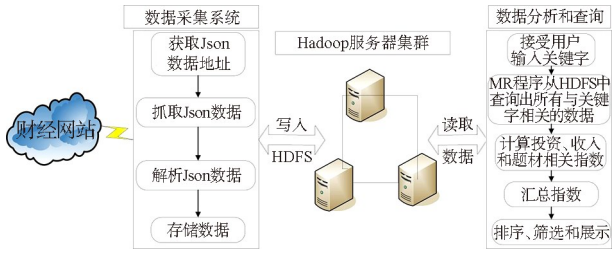


图 1　系统总体架构

2 系统的详细设计

系统的分布式存储平台是利用 3 台虚拟机构建的 Hadoop 平台,系统的应用程序是在 Java 平台上实现的,实现过程利用了开发框架 SpringMVC、定时任务组件 Quartz、前端组件 BootStrap 和可视化组件 Echarts 等技术,3 个子系统的基本情况如下。

2.1 Hadoop 服务器集群子系统设计

该子系统主要由 3 台 Linux 服务器构成,由于资源的限制,这里是利用 3 台虚拟机来搭建一套分布式 Hadoop 平台,用于分布式数据存储 HDFS 和执行分布式计算 MR 程序,并安装配置数据仓库 Hive,用于管理数据[5]。

2.2 数据采集子系统设计

该子系统主要完成数据采集功能,是一套基于 Hadoop 的 Java Appliation 程序。利用定时组件 Quartz,通过 Java 平台开发出一套系统,从各大财经网站抓取全体上市公司已公开的各类基本信息,存入到 HDFS 分布式存储上[6],并通过 Hive 和 MapReduce程序进行数据预处理,然后导入关系数据库 MySQL 进行后续的统计和分析。数据采集流程如图 2 所示。

2.3 数据分析、查询和展示子系统设计

该子系统是一套基于 Java Web 的 Web 应用程序,主要用于接收用户输入的查询关键字组合,系统利用抓取的数据集从公司收入、投资和核心概念 3 方面分析和计算出公司与用户需要查询的关键字组的相关指数,最后汇总为总相关指数,总相关指数越高,相关度越高,相关度越高的公司越有可能就是用户想要查找的相关主题概念公司。最后将统计分析的结果按照相关指数的大小进行排序并传到前端,在前端利用 BootStrap 和 Echarts 组件将结果直观地展示在 Web 页面上,便于用户对结果一目了然[7]。该子系统工作流程如图 3 所示。

系统会根据用户需要查询的主题概念关键词,分别对题材指数、投资指数和收入指数进行计算,由于相关指数计算是系统的一个重难点,在图 3 的

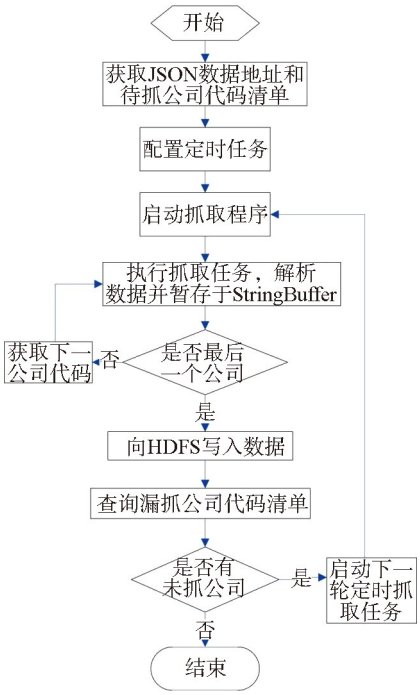


图 2 数据采集系统工作流程

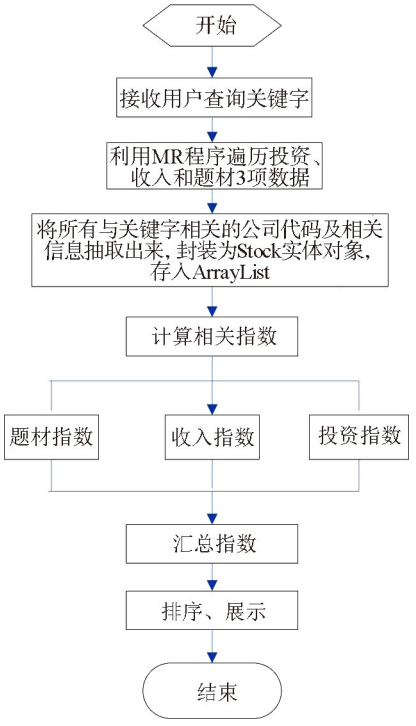


图 3 数据分析和指数计算流程

流程图中无法很好地展示出来,因此在后文将以题材相关指数的计算为例详细阐述指数计算的过程,计算后进行归一化,然后再汇总得出总相关指数。将每支标的个股的查询结果封装为 Stock 对象,其中包含个股名称、代码、题材详情、投资详情、收入详情、3 项指数以及汇总指数。然后再按汇总指数排序并存入 ArrayList,再传给前端,在前端读入该 ArrayList 后进行遍历,并利用可视化组件 Echart 进行展示。最终展示效果如图 4 所示。

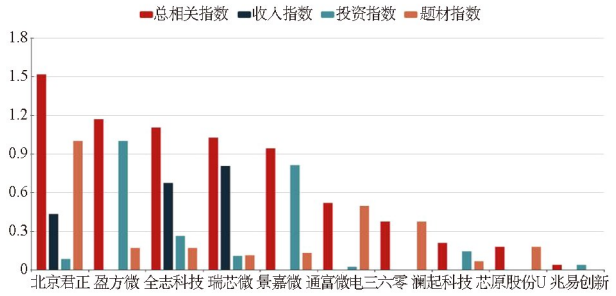


图 4 查询“CPU|处理器”组合关键字结果的可视化展示

3 系统实现的关键技术点——数据的抓取

为了实现定期抓取几千家公司的数据,并以指定的格式存入 HDFS 中,本文通过以下 4 个环节实现了数据的定期抓取和存储操作,下面以主营业务数据为例来详细讨论数据的抓取过程。

1)获取数据地址。由于上市公司大多的数据都是动态更新的,如交易数据在交易期间每时每刻都在变化,所以这些经常变化的数据基本都不会以静态的方式呈现在页面上。本文使用的数据主要包括上市公司核心题材、主营收入和投资项目 3 项数据,虽然变化没有交易数据那么频繁,但也是动态数据,基本每个季度都会更新一次。经过对网页的分析,利用 Fiddler 抓包工具,可以找到这些数据的传送方式,它们大多是以 JSON 格式发送到客户端的,利用 Fiddler 截取每一类数据的 JSON 地址,然后再利用代码对这些数据进行定时抓取。

2)安排定时任务。由于本文的应用需求,需要 A 股市场所有上市公司数据,同时还要定期进行抓取,以获得最新的数据,所以抓取过程中定时任务少不了,既要控制程序定期抓取,也要控制程序抓取频率,以防止服务器因访问过于频繁而阻止了客户端发出的 HTTP 抓取请求[8]。本文利用第三方作业调度框架 Quartz 和 Java 自带的 TimerTask 机制实现双层定时机制,通过 Quartz 实现定期启动抓取任务[9],如对主营收入数据,按照每季度启动一次抓取程序,每次启动后再对全市场所有上市公司数据进行抓取。每次定时任务启动后,再利用 TimerTask 来控制抓取频率,从而解决对服务器访问速度过快导致服务器拒绝连接以及数据来不及存储而导致的数据丢失等问题。

3)数据抓取。由于每次任务启动后需要抓取全市场近 4 000 家公司的数据,所以真正实现数据抓取,首先需要获取所有上市公司的代码列表,下

面代码中的 code_list 就是所有上市公司的代码列表,然后把每个公司代码与上面获得的 JSON 地址进行拼接以获得每个公司各项数据的真实地址,然后才能真正进行数据的抓取[10]。数据抓取的核心代码如下:

```
TimerTask task = new TimerTask() {
public void run() {
String result, urlString, line,data,code,sql,key;
JSONArray jArray,hy_arr,qy_arr,cp_arr;
JSONObject jObj,jObj2,jObj3;
Iterator<String> iterator;
if (currentID < code_list.size()) {
code=code_list.get(currentID);
data="";
urlString = "http://emweb.securities.eastmoney.com/PC_HSF10/BusinessAnalysis/BusinessAnalysisAjax?code="+code;
result = tools.getJsonString(urlString);
System.out.println(currentID+"\tresult="+result);
try {
jObj=new JSONObject(result);
jArray=jObj.getJSONArray("zygcfx");
//jArray 就是要抓取的主营收入 JSON 数据,接下来调用解析程序进行数据解析。
}
strBuffer.append(data);
} catch (JSONException e) {
// TODO Auto-generated catch block
e.printStackTrace();
}
currentID++;
return false;
}else{
table_name="main_business_income";
sava_Thread=new Sava_Thread();          //启动数据存储线程。
sava_Thread.start();
return true;
}
};
};
timer.schedule(task, 10, 5000);
```

4)解析和存储数据。上一个步骤将每家公司的主营业务收入数据已经从网络上抓取下来了,并以 JSONArray 的形式获取到程序中,再进一步分析每家公司的主营收入数据。其中又包括区域、产品和行业 3 种形式描述的主营收入数据[11],下面再调用解析程序对每家公司的主营收入数据进行解析,按照不同的类别进行数据重组才能存入到 HDFS 中,数据解析的核心代码如下:

```
jArray=jObj.getJSONArray("zygcfx");
for(int  i=0;i<jArray.length();i++){
jObj2=jArray.getJSONObject(i);
hy_arr=jObj2.getJSONArray("hy");
for(int j=0;j<hy_arr.length();j++){
jObj3=hy_arr.getJSONObject(j);
iterator=jObj3.keys();
data+=code+"\thy\t";
while(iterator.hasNext()){
key=iterator.next();
data+=jObj3.getString(key)+"\t";}
data+="\n";}
qy_arr=jObj2.getJSONArray("qy");
for(int j=0;j<qy_arr.length();j++){
jObj3=qy_arr.getJSONObject(j);
iterator=jObj3.keys();
data+=code+"\tqy\t";
while(iterator.hasNext()){
key=iterator.next();
data+=jObj3.getString(key)+"\t";}
data+="\n";}
cp_arr=jObj2.getJSONArray("cp");
for(int j=0;j<cp_arr.length();j++){
jObj3=cp_arr.getJSONObject(j);
iterator=jObj3.keys();
data+=code+"\tcp\t";
while(iterator.hasNext()){
key=iterator.next();
data+=jObj3.getString(key)+"\t";}
data+="\n";}
strBuffer.append(data);
```

数据解析后,将对数据进行存储,考虑到 HDFS 的访问特性,不适合少量数据的频繁写入操作,所以每家公司主营业务数据抓取后并不立即写入,而是存入到 StringBuffer 中,等所有公司的主营收入数据全部抓取完毕后,再一次性写入,以提高 HDFS 的访问效率以及空间利用率,同时由于数据存储的耗时性,开辟了新的线程来完成数据的存储操作[12]。

3.1 相关指数的计算

本文研究的主题概念股票挖掘主要以 4 项指数来做为主题概念股票的最终挖掘指标,这 4 项指数是题材概念指数、收入指数、投资指数和总相关指数,总相关指数是由前 3 项指数汇总而来的,这里以题材概念指数的计算为例,简要阐述一下题材概念指数的计算过程,其他 2 个指数的计算过程类似,不再赘述。题材概念指数的计算流程如图 5 所示。

题材概念指数的计算主要包括以下几个步骤:

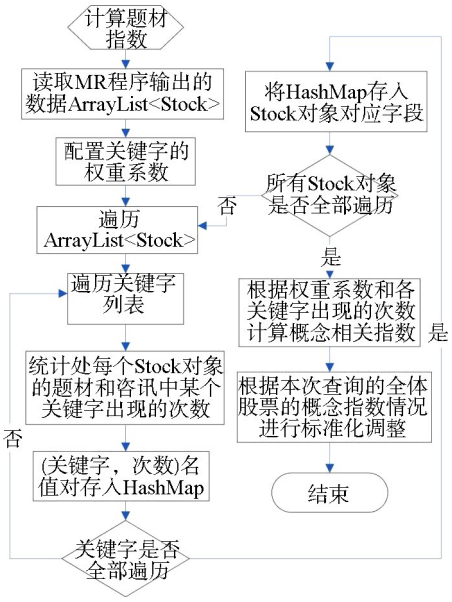


图 5 题材概念指数的计算流程

1)读取用户查询的关键字列表。系统利用表单接受用户需要挖掘的主题概念关键字,目前关键字限制在 1~3 个,用字符'|'分割,系统读取用户查询关键字后,对关键字进行分割并存入数组中,以备后用。

2)关键字权重系数配置。系统提供了每个关键字的权重系数配置功能,当用户输入多个关键字时,可以通过这个环节设置每个关键字在指数计算中所占的比重,从而挖掘出更符合客户需求的主题概念股票名单[13]。

3)获取含有用户关键字列表的公司列表。读取所有公司的题材数据,并利用字符串的 contains 方法先初步获取含有任何一个关键字的公司列表,并封装为 Stock 对象存入 ArrayList 中,这些公司都是潜在的挖掘对象[14],也是下一步需要进一步处理的股票对象列表,通过本环节的操作可以大大减少后期指数计算时的股票数量。

4)统计每个关键字在每支个股核心题材中出现的次数。通过遍历关键字列表和前一步产生的存入在 ArrayList 中的 Stock 对象列表,统计出每个可能相关的公司的核心题材中各个关键字的出现次数,并存入到名为 concept_map 的 HashMap 中。核心实现代码如下:

```
tempStr = "";
while (core_concept_rs.next()) {
if (stock_code.equalsIgnoreCase(core_concept_rs
.getString("cStock"))) {
tempStr += core_concept_rs.getString("ydnr")
+ core_concept_rs.getString("gjc");
} else {
core_concept_rs.previous();
break;}
}
int times;
for (int i = 0; i < keyword.length; i++) {
times = (tempStr.length() - tempStr.toUpperCase().re-
placeAll(
keyword[i].toUpperCase(), "").length())
/ keyword[i].length();
concept_map.put(keyword[i], times);
}
stock.setKeyword_statistics(concept_map);
```

5)计算题材概念指数。根据前面环节统计的每个关键字的出现次数以及每个关键字的权重系数,以次数乘以权重系数计算出每支潜在相关概念股票的题材相关指数,每支个股的题材概念指数计算后也存入到名为 concept_map_degree 的 HashMap 中,其核心实现代码如下:

```
concept_map = stock.getKeyword_statistics();
if (concept_map ! = null) {
float t = 0;
int sum = 0;
int data[] = new int[keyword.length];
for (int j = 0; j < keyword.length; j++) {
data[j] = concept_map.get(keyword[j]);
}
//计算并汇总每个关键字的相关指数值=次数 * 权
重系数
for (int j = 0; j < data.length; j++) {
sum += data[j];
t+= data[j] * rate[keyword.length - 1][j];
}
t += sum * rate[keyword.length - 1][data.length];
concept_map_degree.put(stock.getCode(), t);
}
```

6)指数归一化处理。归一化是将最终相关指数控制在 0~1。上一步计算的相关指数范围是无法确定的,可能会很大,也可能很小,这种大范围的指数在展示时可能会造成展示效果不够直观,所以要将指数归一化[15]。具体实现方法比较简单,可以将上一个环节暂存在 concept_map_degree 中的各个股相关指数取出,然后找出指数中的最大值,然后再让其他的每个指数除以该最大值,从而实现指数的归一化处理。

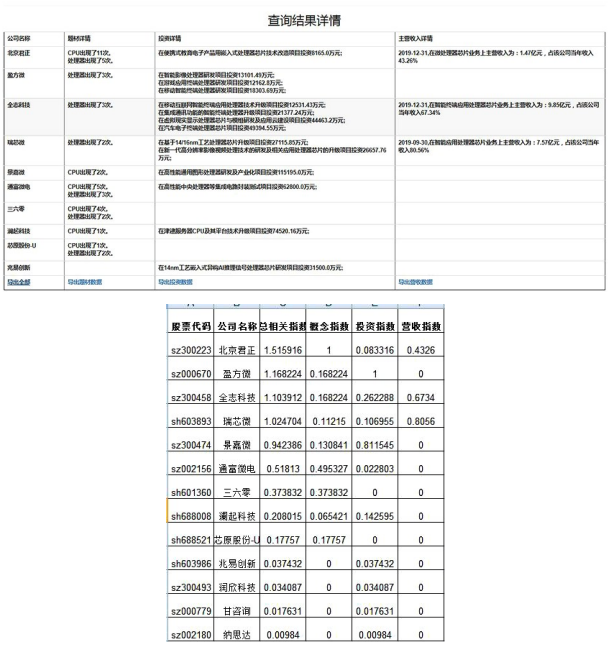
4 结果分析与讨论

以"CPU|处理器"组合关键字为例,通过模型

系统的分析和计算,挖掘出了 13 家符合条件的主题概念股票,这 13 家公司按照总相关指数依次排列如图 6 所示。从详细数据中可以看出,排名第 1 的“北京君正”在题材概念中,“CPU”概念出现了 11 次,“处理器”概念出现了 5 次,在挖掘的所有公司中,出现“CPU|处理器”组合关键字的总次数最多,从图 10 右侧的具体指数中可以看出其概念指数为 1,在所有公司中也是最高的;在处理器芯片上有 8 165 万元投资,投资指数为 0.08;最近一年度在微处理器芯片上有 1.47 亿元的营业收入,占公司总营收 43.26%,营收指数为 0.43,3 项指数总计 1.52,在本文所选的 3 项指标上,该公司都有相关性。经查阅公开资料,该公司主营业务为微处理器芯片、智能视频芯片及整体解决方案的研发和销售,公司自成立以来一直专注于国产 CPU 技术的研发,拥有全球领先的 32 位嵌入式 CPU 技术和低功耗技术,与我待查的“CPU|处理器”概念相关度很高。

在排名第 3、4 的“全志科技”和“瑞芯微”也都与这 3 项指标相关,但是它们在题材概念中出现“CPU|处理器”的次数相对较少,且只出现了“处理器”,没有出现“CPU”关键字,从而在概念指数方面较低;但“全志科技”和“瑞芯微”在投资和主营收入方面与待查询关键字都有较高的相关度,如“全志科技”的投资指数为 0.26,营收指数为 0.67;“瑞芯微”的投资指数为 0.11,营收指数为 0.81,累加后其总相关指数也较高。通过查阅公开资料,这 2 家公司确实是以各类处理器为主营业务的公司,与本文待查的“CPU|处理器”概念相关度很高,可能正是本文所要关注的标的之一。

排名第 2 的“盈方微”虽然总相关指数较高,但该公司由于经营不善业绩连续 3 年亏损,目前已暂停上市。后期系统将采取措施对 ST 或者退市类股票进行过滤,排除在挖掘范围之外。该公司相关指数较高的主要原因是由于其在“处理器”方面投资项目总额较大,累计约 4.3 亿元,而其最近一个年度营业收入只有 413 万元,从而导致投资额占营收比例过大,从而使得该公司的投资指数在所有公司中最大,即为 1,从而也提高了该公司的总相关指数,使其排到第 2 的位置。系统设计的初衷是按照在指定概念上的项目投资总额占公司营收的比例来确定投资指数的,占比越高,投资指数越高。但在这个案例上,由于公司营收的大幅波动,从而导致该项指数有些失真,后期将不断改善算法设计,采用若干年营收均值来进行计算,从而可以平滑营收大幅波动带来的扰动。



股票代码	公司名称	总相关指数	概念指数	投资指数	营收指数
sz300223	北京君正	1.515916	1	0.083316	0.4326
sz000670	盈方微	1.168224	0.168224	1	0
sz300458	全志科技	1.103912	0.168224	0.262288	0.6734
sh603893	瑞芯微	1.024704	0.11215	0.106955	0.8056
sz300474	景嘉微	0.942386	0.130841	0.811545	0
sz002156	通富微电	0.51813	0.495327	0.022803	0
sh601360	三六零	0.373832	0.373832	0	0
sh688008	澜起科技	0.208015	0.065421	0.142595	0
sh688521	芯原股份-U	0.17757	0.17757	0	0
sh603986	兆易创新	0.037432	0	0.037432	0
sz300493	润欣科技	0.034087	0	0.034087	0
sz000779	甘咨询	0.017631	0	0.017631	0
sz002180	纳思达	0.00984	0	0.00984	0

图 6　主题概念股票详细数据

从实验结果的详细数据分析中可以看出,公司在题材概念中出现待查主题关键词次数越多,其概念指数越高;公司在投资方面与待查主题关键词相关的投资额占公司营收比例越高,其投资指数越高;公司在营收方面与待查主题关键词相关的营收额占公司总收入比例越高,其营收指数越高。最后汇总这 3 项指数得出总相关指数,公司总相关指数越高,则与待查主题概念相关度就越高。这个指数不但考虑了传统基于文本信息的挖掘,同时还增加了公司在待查主题概念方面的实际投资额和营收额两项量化指标,更具有量化特性。营收额代表了公司的过去,投资额代表公司的未来,通过这种综合分析可以提高相关度的可靠性和准确性。

5 结语

本文基于个股的核心题材概念、主营收入和投资 3 项数据进行统计分析,可以快速挖掘某个相关概念的股票集合。这种挖掘可以克服目前市场上固有板块划分的限制,但这种挖掘也局限于在核心概念、主营收入和投资 3 项数据中出现相关概念,才能够被挖掘出来,否则将无法被挖掘出来,这也是本系统当前存在的局限性。后期将扩大用于挖掘的数据源以及实现挖掘的算法,将公司的日常报告、股吧信息、研究报告等纳入挖掘范畴,并利用机器学习算法实现更加智能的数据挖掘,以不断提高主题概念股票挖掘的准确度和可靠性。

参考文献:

[1] 张惠玲.基于股吧文本的主题挖掘及其股票投资应用[D].广州:华南理工大学,2018.
[2] 庄郭冕.基于主题模型和实体识别的股市热点概念挖掘[D].杭州:浙江大学,2018.
[3] 鲁志芳.基于 Hadoop 技术的大数据分析应用系统的研究与设计[J].电子设计工程,2019(16):11-14.
[4] 唐黎.面向金融大数据的高效数据处理机制的研究与设计[D].北京:北京邮电大学,2015.
[5] 汪保友,钱晶,袁时金.基于 Hadoop 的电信大数据采集方案研究与实现[J].电信科学,2017,33(1):135-142.
[6] 袁昌权,胡益群,许光,等. 基于 Hadoop 的高可用数据采集与存储方案[J].电子技术与软件工程,2019(18):169-170.
[7] 王悦.基于数据挖掘算法的金融数据采集与分析研究[D].天津:天津大学,2016.
[8] 焦继笑.基于 Web Services 和 Quartz 的数据整合系统的设计与实现[D].北京:北京交通大学,2016.
[9] 丁俊.基于 HBase 的证券交易数据实时采集系统的应用研究[J].黑龙江工业学院学报(综合版),2019(12):42-49.
[10] 叶刚.基于 Quartz 的可视化定时任务管理方案[J].电子技术与软件工程,2018,1(17):139-140.
[11] 李艳斌.基于数据挖掘技术的股票选择分析研究[D].大连:东北财经大学,2018.
[12] 郭鹏程,李迎春,付春燕,等.海量日志数据采集系统的设计与优化[J].电子测量技术,2018,41(1):12-17.
[13] 陈民.基于多层次数据交换的区域智慧城市公共信息平台[J].计算机应用与软件,2016,33(12):67-70.
[14] 李希尧.基于数据挖掘技术的股票数据分析研究[D].成都:电子科技大学,2020.
[15] 王凯.基于集成学习的量化选股策略研究[D].广州:华南理工大学,2017.

(上接第 77 页)

参考文献:

[1] 中华人民共和国水利部.2019 年中国水资源公报[EB/OL].(2020-08-03)[2021-05-02].http://www.mwr.gov.cn/sj/tjgb/szygb/202008/t20200803_1430726.html.
[2] 张兵,袁寿其,成立.节水灌溉自动化技术的发展及趋势[J].排灌机械,2003(2):37-41.
[3] 袁寿其,李红,王新坤.中国节水灌溉装备发展现状、问题、趋势与建议[J].排灌机械工程学报,2015,33(1):78-92.
[4] 岳学军,刘永鑫,洪添胜,等.基于土壤墒情的自动灌溉控制系统设计与试验[J].农业机械学报,2013,44(S2):241-246+250.
[5] 刘晓艳.基于土壤墒情监测的农田灌溉系统研究[J].自动化应用,2020(9):1-3.
[6] 杨波,魏文政,陈盟,等.基于神经网络的智能化节水灌溉系统设计研究[J].水利技术监督,2020(5):44-48.
[7] 王应海.大数据及人工智能技术在灌溉领域的应用初探[J].节水灌溉,2017(3):100-102.
[8] 高玉芹.基于 ZigBee 和模糊控制决策的自动灌溉系统的设计[J].节水灌溉,2010(8):52-55.
[9] 张伟,何勇,裘正军,等.基于无线传感网络与模糊控制的精细灌溉系统设计[J].农业工程学报,2009,25(S2):7-12.
[10] 谢守勇,李锡文,杨叔子,等.基于 PLC 的模糊控制灌溉系统的研制[J].农业工程学报,2007(6):208-210.
[11] 杨伟志,孙道宗,刘建梅,等.基于物联网和人工智能的柑橘灌溉专家系统[J].节水灌溉,2019(9):116-120+124.
[12] STMicroelectronics. DS5792_STM32F103xC, STM32F103xD, STM32F103xE 单片机数据手册[DB/OL].(2018-08-15)[2021-05-04].https://www.stmcu.com.cn/Product/pro_detail/cat_code/STM32F103/family/81/sub_family/124/sub_child_family/142/layout/product.
[13] 郑海雷,黄子琛.春小麦单叶气孔行为及蒸腾作用的模拟[J].高原气象,1992(4):423-430.
[14] 郭松年,丁林,王福霞.作物调亏灌溉理论与技术研究进展及发展趋势[J].中国农村水利水电,2009(8):12-16.
[15] 郑永丹.中国主要粮食作物生育期时空格局及其变化[D].武汉:华中师范大学,2015.
[16] 王一罡,徐践,郭大鹏,等.北京冬小麦各生育期灌水量辅助决策模型研究[J].华北农学报,2015,30(S1):241-244.
[17] 宓文海,江荣风,刘全清,等.不同灌溉方式对华北冬小麦生长的影响[J].华北农学报,2013,28(2):175-179.