

基于认知计算的就业咨询智慧服务系统

唐新晨

(南京邮电大学 通信与信息工程学院,江苏 南京 210000)

摘要:随着智慧服务系统的发展和大数据时代的到来,如何实现类似人脑的认知与判决为应届生求职方向做出正确的决策,显得尤为重要。智慧服务系统由四部分组成,数据采集单元使用 Scrapy 爬虫框架获取信息,能够实时从各大招聘网站采集招聘信息;数据计算平台使用随机森林、SVM 和朴素贝叶斯等基于认知计算的相关算法进行文本识别、特征提取以及文本分类等工作,能够正确实现特征采样和数据分类;数据存储单元搭建 MongoDB 数据库集群完成数据存储工作,具备海量数据储量能力和高容错性;用户服务平台由 Web 应用框架构建,具备多用户业务服务能力。因此其能够有效采集和分类招聘信息,准确定位学生能力,从而高效地为院校学生的就业岗位选择提供咨询与帮助。

关键词:认知计算;Scrapy 爬虫;机器学习;Web 应用;服务系统

中图分类号:TP302

文献标识码:A

文章编号:1673-629X(2017)11-0166-05

doi:10.3969/j.issn.1673-629X.2017.11.036

Employment Consultation Intelligent Service System Based on Cognitive Computation

TANG Xin-chen

(School of Telecommunications and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210000, China)

Abstract:Currently, with the development of the intelligence service system and the arrival of the big data era, how to use the computer to help graduates make right decisions of job hunting like human is particularly important. Employment consultation intelligent service system with cognitive computation consists of four parts. Data collection unit uses the Scrapy framework for massive employee information from the various employee network in real-time. Data computing platform carries out the text recognition, feature extraction and text classification by several algorithms based on cognitive computing like random forest, SVM and Naive Bayes, which can correctly realize the feature sampling and data classification. Data storage unit builds the MongoDB cluster to complete the data storage with large memory capacity and high fault tolerance. User service platform integrates the Web framework and has multiple user services. Therefore, it can collect and classify effectively the employee information and evaluate students' ability accurately, which can provide students for effective help on choosing the right and good job.

Key words: cognitive computing; Scrapy; machine learning; Web application; service system

0 引言

IBM 在 2013 年宣布成立“认知计算研究联合会”。国内于 2013 年 10 月 11 日在北京举办了以“从大数据到认知计算”为主题的认知计算研讨会,达成“我们已经进入了认知计算的新时代”的共识。经过长期调研发现应届生就业面临如下问题:就业信息挖掘不足、应届生对自身实力定位不当而造成就业困难等。因此应当构建基于认知计算的就业咨询智慧服务系统,有效为院校学生就业岗位的选择提供咨询与帮

助。该系统能够实现招聘信息的采集和分类、学生实力的准确定位、信息的定向推送等功能。其主要由数据采集单元、数据计算平台、用户服务平台和数据存储单元四个部分组成。下面将从系统设计、技术选择、系统实现以及结果展示这四个角度重点阐述其工作原理^[1]。

1 系统设计

该项目的技术方案设计包括四部分:

收稿日期:2016-05-19

修回日期:2016-08-17

网络出版时间:2017-08-01

基金项目:全国 3S 杯大学生物联网技术与应用“三创大赛”组委会项目支持(16B049)

作者简介:唐新晨(1992-),男,硕士研究生,研究方向为网络技术应用。

网络出版地址: <http://www.cnki.net/kcms/detail/61.1450.TP.20170801.1548.002.html>

(1) 设计并搭建数据采集单元。

通过问卷调查、联合社团与院校合作等方式选取近年来南京邮电大学高质量的研究生简历以及最终就业单位、岗位信息。通过 Scrapy 爬虫框架,爬取各大就业信息网(南京邮电大学招生就业创业网、南大百合 BBS 等)的就业信息,并进行数据预处理。

(2) 设计并搭建数据计算平台。

使用多类别支持向量机、朴素贝叶斯算法,构造“就业岗位智慧分类模型”,对提取的就业信息进行数据分类;采用随机森林算法对用户简历信息进行数据分析,构造“就业智慧决策树模型”,洞察简历信息与就业岗位的内在联系,完成用户岗位信息的预测判决。

(3) 设计并搭建用户服务平台。

使用 SSH 框架完成人机交互服务与业务逻辑设计、数据展示等。

(4) 搭建数据存储单元。

采用 MongoDB 数据库完成数据存储,并配置用户登录、副本集等功能,保障数据安全和冗余备份。

具体业务流程如图 1 所示。

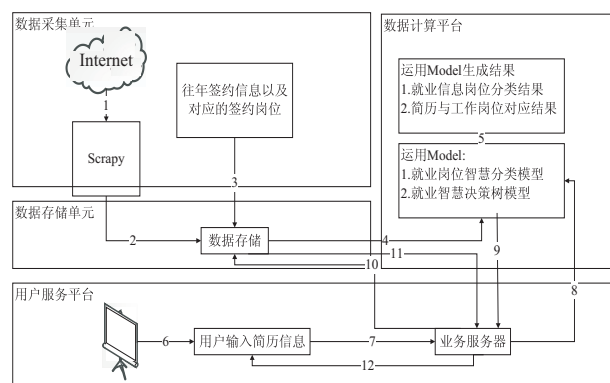


图 1 系统框架及业务流程

2 Scrapy 框架结构

Scrapy 是一个快速、高层次的屏幕抓取和 Web 抓取框架,用于抓取 Web 站点并从页面中提取结构化的数据。Scrapy 可用于数据挖掘、监测和自动化测试,并且是开源框架最新版本,提供了 Web2.0 爬虫的支持。

Scrapy 框架的主要构件是引擎,调度器,下载器,蜘蛛,管道项目,下载器中间件,蜘蛛中间件以及调度中间件^[2]。

3 数据计算平台算法设计

3.1 朴素贝叶斯算法的应用

朴素贝叶斯是贝叶斯分类器的一个扩展,是用于文档分类的常用算法。它在数据较少的情况下仍然有效,并且可以处理多类别问题^[3]。

根据贝叶斯定理,对于一个分类问题给定样本特

征 X , 样本属于类别 Y 的概率为:

$$P(y|x) = P(x|y)P(y)/P(x) \quad (1)$$

其中, x 为一个特征向量。假设 x 的维度为 M 。因为朴素的假设,即特征条件独立,根据全概率公式展开,式(1)可以表达为:

$$P(y = C_k | x) = \frac{\prod_{i=1}^M P(x^i | y = c_k) P(y = c_k)}{\sum_k P(y = c_k) \prod_{i=1}^M P(x^i | y = c_k)} \quad (2)$$

这里只要分别估计出特征 x^i 在每一类的条件概率即可。类别 y 的先验概率可以通过训练集计算出,同样通过训练集上的统计,可以得出对应每一类上条件独立的特征对应的条件概率向量^[4]。

从获得的数据中,通过学习得到朴素贝叶斯分类模型。具体做法如下:

TrainingSet = $\{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$ (训练集)包含 N 条训练数据,其中 $x_i = (x_i^{(1)}, x_i^{(2)}, \dots, x_i^{(M)})$ 是 M 维向量, $y_i \in \{c_1, c_2, \dots, c_k\}$ 属于 K 类中的一类。首先,计算式(2)中的 $p(y = c_k)$ 。

$$p(y = c_k) = \frac{\sum_{i=1}^N I(y_i = c_k)}{N} \quad (3)$$

其中, $I(x)$ 为指示函数,若括号内成立,则计 1,否则计 0。

接下来计算分子中的条件概率。设 M 维特征的第 j 维有 L 个取值,则某维特征的某个取值 a_{jl} , 在给定某分类 C_k 下的条件概率为:

$$P(x^j = a_{jl} | y = c_k) = \frac{\sum_{i=1}^M I(x_i^j = a_{jl}, y_i = c_k)}{\sum_{i=1}^N I(y_i = c_k)} \quad (4)$$

经过上述步骤,就得到了模型的基本概率,也就完成模型构建的任务。

之后当给定未分类新实例 x 时,就可通过上述概率进行计算,得到该实例属于各类的后验概率 $P(y = c_k | x)$ 。因为对所有的类别来说,式(2)中分母的值都相同,所以只计算分子部分即可,具体步骤如下:

计算该实例属于 $y = c_k$ 类的概率:

$$P(y = c_k | X) = P(y = c_k) \prod_{j=1}^n P(X^{(j)} = x^{(j)} | y = c_k) \quad (5)$$

得到该实例所属的分类 y :

$$y = \underset{c_k}{\operatorname{argmax}} P(y = c_k | X) \quad (6)$$

3.2 支持向量机算法的应用

支持向量机(Support Vector Machine, SVM)是一种通过寻求结构化风险最小来提高学习机泛化能力的分

类算法,实现经验风险和置信范围的最小化,从而达到在统计样本量较少的情况下,亦能获得良好统计规律的目的^[5]。

一般而言,一个点距离超平面的远近可以表示为分类预测的确信或准确程度。SVM 就是要最大化这个间隔值。分割超平面的形式可以写为 $\mathbf{w}^T \mathbf{x} + b$,若计算任意一点到达超平面的距离就是 $\frac{|\mathbf{w}^T \mathbf{x} + b|}{\|\mathbf{w}\|}$ 。其中常数 b 指截距。在实际的分类过程中,使用类似海维赛德阶跃函数对 $\mathbf{w}^T \mathbf{x} + b$ 作用得到 $f(\mathbf{w}^T \mathbf{x} + b)$ 。其中当 $u < 0$ 时, $f(u)$ 输出 -1 ,反之则输出 $+1$ 。因此 SVM 的核心就是寻找具有最小间隔的数据点,并利用这些点对间隔进行最大化,从而确定 \mathbf{w} 和 b 的值。具体的表达式可写为: $\arg \max_{\mathbf{w}, b} \{ \min(\text{label} \cdot (\mathbf{w}^T \mathbf{x} + b)) \cdot \frac{1}{\|\mathbf{w}\|} \}$ 。

通过引入拉格朗日乘子,使用基于约束条件表达式对目标函数进行进一步的解析,最终可以写为:

$$\max_{\alpha} \left[\sum_{i=1}^m \alpha - \frac{1}{2} \sum_{i,j=1}^m \text{label}^{(i)} \cdot \text{label}^{(j)} \cdot a_i \cdot a_j \langle x^{(i)}, x^{(j)} \rangle \right].$$

约束条件为: $\alpha \geq 0, \sum_{i=0}^m \alpha_i \cdot \text{label}^{(i)} = 0$ ^[6]。

求解 SVM 就是求解该表达式的最优解问题。

3.3 随机森林算法的应用

随机森林算法在机器学习、计算机视觉等领域内应用极为广泛,可以用来做分类和回归。随机森林由多个决策树构成,相比于单个决策树算法,分类、预测的效果更好,不容易出现过拟合的情况^[7]。

随机森林是由多个决策树构成的森林,算法分类结果由这些决策树投票得到。当基于某些属性对一个新的对象进行分类判别时,随机森林中的每一棵树都会给出自己的分类选择,并由此进行“投票”,森林整体的输出结果将会是票数最多的分类选项;而在回归问题中,随机森林的输出将会是所有决策树输出的平均值。决策树在生成过程中分别在行方向和列方向上添加随机过程。行方向上构建决策树时采用放回抽样得到训练数据,列方向上采用无放回随机抽样得到特征子集,并据此得到其最优切分点。

3.4 特征向量提取

特征向量提取的最终目标是使得选出的特征向量在多个类别之间具有一定的类别区分度。由于分词后得到大量的词语,通过选择降维技术能很好地减少计算量,并维持分类的精度。这里介绍卡方统计量和 TD-IDF 两种特征向量提取算法。

计算卡方统计的公式如下:

$$\chi^2(t, \text{词}) = \frac{(n_{t, \text{词}} - \frac{n_{t, \cdot} \cdot n_{\cdot, \text{词}}}{n})^2}{\frac{n_{t, \cdot} \cdot n_{\cdot, \text{词}}}{n}}$$

$$\frac{N \times (AD - CB)^2}{(A + C) \times (B + D) \times (A + B) \times (C + D)} \quad (7)$$

其中, N 为训练数据集文档总数; A 为在一个类别中包含某个词的文档数量; B 为在一个类别中排除该类别后,其他类别包含某个词的文档数量; C 为在一个类别中不包含某个词的文档数量; D 为在一个类别中不包含某个词,也不在该类别中的文档数量

TF-IDF (Term Frequency - Inverse Document Frequency) 是一种用于资讯检索与资讯探勘的常用加权技术。主要思想是:如果某个词或短语在一篇文章中出现的频率(TF)高并且在其他文章中很少出现,则认为此词或者短语具有很好的类别区分能力,适合用来分类。计算 TD-IDF 的公式如下:

$$\text{tfidf}_{i,j} = \text{tf}_{i,j} \cdot \text{idf}_i = \frac{n_{i,j}}{\sum_k n_{k,j}} \cdot \log \frac{|D|}{|\{j: t_i \in d_j\}| + 1} \quad (8)$$

其中, $n_{i,j}$ 为该词在文件 d_j 中出现的次数; $|D|$ 为语料库中的文件总数; $\sum_k n_{k,j}$ 为在文件 d_j 中所有字词的的出现次数之和; $|\{j: t_i \in d_j\}|$ 为在一个类别中,不包含某个词的文档数量包含词语 t_i 的文件数(即 $n_{i,j} \neq 0$ 的文件数目)。

4 系统结构设计

Struts2 和 SpringMVC 是目前比较流行的 MVC Web 后台框架,都规范封装了 Servlet 的开发,大大提升了 Web 后台的开发效率^[8]。Hibernate 是一个开放源代码的对象关系映射框架,对 JDBC 进行了轻量级的封装,使得 Java 程序员可以随心所欲地使用对象编程的思维来操纵数据库,并且它提供了对常用数据库的基本操作^[9]。Spring 是一个轻量级 Java 开发框架,是轻量级的 IoC 和 AOP 的容器框架。主要功能是提供了对象之间的解耦,简化开发,以及 AOP 编程,声明式事务的支持等功能^[10]。

MongoDB 是一种非关系型数据库,与关系型数据库相比,具有弱一致性、基于内存存储方式、支持大容量存储、更快速获取数据、内置 Sharding 提供数据分段存储等特点^[11]。

5 系统关键模块设计与结果展示

5.1 数据采集单元

采用 Scrapy 完成招聘数据的大量采集,采集的目标网站为南京邮电大学就业创业网等四家高校的招生就业信息专栏。图 2 展示了 Scrapy 爬虫获取的信息经过处理后得到的文本文件截图,可见数据采集单元具备就业数据采集能力。

2016-02-23 活动经理 <http://njupt.91job.gov.cn/view/id/472164> 岗位介绍: 1.负责编写活动策划方案和活动策划
2016-02-20 通信硬件工程师 <http://njupt.91job.gov.cn/view/id/472165> (一) 公司介绍 南京日晖信息技术有限公司
2016-02-22 通信软件处理软件工程师 <http://njupt.91job.gov.cn/view/id/472166> (一) 公司介绍 南京日晖信息技
2016-02-23 乐尚教师 <http://njupt.91job.gov.cn/view/id/472167> 岗位要求: 1、本科及以上学历,幼教或理科或机
2016-02-23 SAPFICO/MM/SD/PP实训顾问 <http://njupt.91job.gov.cn/view/id/472168> 1.沟通表达能力好,形象气质

图 2 经过 Scrapy 采集得到的文本文件截图

5.2 数据计算平台

数据平台的设计使用朴素贝叶斯算法、多类别 SVM 算法、随机森林算法。使用朴素贝叶斯算法完成岗位信息的技术与非技术分类;使用多类别 SVM 算法完成与技术相关的开发、测试、技术支持和其他的分类;使用随机森林算法完成职位预测功能。具体介绍多类别 SVM 的实现。工作大致分为以下几个步骤:

(1)选择文本训练数据集和测试数据集:训练集和测试集都是类标签已知的,都是由 Scrapy 从网上爬取的各大招生就业信息,经过朴素贝叶斯分类后形成的所有技术相关的就业信息。

(2)训练集文本预处理:包括分词、去停用词、建立词袋模型(倒排表)。系统使用了 MManalyzer 完成分词的操作,使用停用词字典完成停用词去除,并将字典保存于 vocab 变量中。

(3)选择文本分类使用的特征向量(词向量):使用卡方统计量和 TD-IDF 提取特征向量。卡方统计具体代码如下:

```
common.FeatureMap.java 完成整个过程的调度;
其中 public Map<String, Double> processOneLabel(int label)
函数提供了计算卡方统计量的函数。
int N=item.get(label).size()+Left_Label(label).size();
;
intA=docCountContainingWordInLabel;
intB=docCountContainingWordNotInLabel;
intC=docCountNotContainingWordInLabel;
intD=docCountNotContainingWordNotInLabel;
Int temp= (A * D - B * C);
double chi= (double) N * temp * temp/ ((A + C) * (A +
B) * (B + D) * (C + D));
word_frequency.put(word,chi);
PublicMap<String, Double>sortmap( Map<String, Double> word
_frequency)
函数将 Map 的值按照 CHI 进行排序;
public Map<String, Double> top N ( Map<String, Double> sorte
dMap, Double n)函数依据 chi 从排序好的 map 中选取 N 个 word
作为该 label 的特征值。
利用 TD-IDF 进行进一步提取,代码如下:
DifferentSchoolAnalyzer 中会调用 component.DocumentTFIDF
FComputation.java 文件的 compute 完成 TF-IDF 的计算
private double multiple(int word_in_one_document,int word_
showtimes_in_one_document
intword_showtimes_in_one_document,intword_showtimes_in_
alldocuments,int all_documents_num){
double tf=(double) word_showtimes_in_one_docu
```

```
ment/(double) word_in_one_document;
double idf=Math.log10((double) all_documents_n
um/word_showtimes_in_alldocuments);
return tf * idf;}
```

最终产生的特征向量编号如图 3 所示。

电子技术:0 正式:1 员工福利:2 移动:3 体系:4 判断:5 敏捷:6 弹性:7 公司:8 较强:9 html:10 勘察:11 审美:12 栈:13 决策:14 内部:15 简历:16 仿真:17 领域:18 准备:19 栋:20 ios:21 分配:22 完成:23 调制解调器:24 有意者:25 研:26 同学们:27 信息技术:28 逻辑思维:29

图 3 选取出的特征向量

各个特征向量对应的 TF-IDF 如图 4 所示。

1 1.0 1:-1.0 2:-1.0 3:-1.0 4:-1.0 5:-1.0 6:0.4285975678138519 4
2 1.0 1:-0.6401515151515151 2:-1.0 3:-0.4662921348314607
3 1.0 1:-1.0 2:-1.0 3:-0.4662921348314607 4:-1.0 5:-1.0 6:-1.0
4 1.0 1:-1.0 2:-1.0 3:-0.6746575342465754 4:-1.0 5:-1.0 6:-1.0
5 1.0 1:-1.0 2:-1.0 3:-0.6199999999999999 4:-1.0 5:-1.0 6:-1.0
6 1.0 1:-1.0 2:-1.0 3:-0.7121212121212122 4:-1.0 5:-1.0 6:-1.0
7 1.0 1:-0.6833333333333333 2:-1.0 3:-1.0 4:-0.086538461538
8 1.0 1:-1.0 2:-1.0 3:-0.5622119815668203 4:-1.0 5:-1.0 6:-1.0
9 1.0 1:-1.0 2:-1.0 3:-1.0 4:-1.0 5:-1.0 6:-1.0 7:-0.141566265038
10 1.0 1:-1.0 2:-1.0 3:-0.5497630331753554 4:-1.0 5:-1.0 6:-1.0

图 4 各特征向量对应的 TF-IDF 值

(4)输出 LIBSVM 支持的量化的训练样本集文件,并基于类别和特征向量来量化文本训练集,使其能够满足使用 LIBSVM 训练所需要的数据格式。

调用 LIBSVM 的接口函数如下所示:

```
public class ClassPrediction {
//对原始样本进行归一化
public void svmScale ( int lower, int upper, String save _
filename, String restore_filename)
//训练数据集生成模型文件
public voidsvmtrain( String[]
options,String training_set_file, String model_file);
//根据模型,对测试数据进行预测
public voidsvmpredict ( String [ ] options ,String test_file,
String model_file,String output_file);
```

(5)测试数据集预处理:同样包括分词(需要和训练过程中使用的分词器一致)、去停用词、建立词袋模型(倒排表),但是这时需要加载训练过程中生成的特征向量,用特征向量排除多余的不在特征向量中的词。

(6)输出 LIBSVM 支持的量化的测试样本集文件:格式和训练数据集的预处理阶段的输出相同。

(7)使用 LIBSVM 训练文本分类器:使用训练集预处理阶段输出的量化的数据集文件,最终输出分类模型文件。

(8)使用 LIBSVM 验证分类模型的精度:使用测试集预处理阶段输出的量化的数据集文件和分类模型文件来验证分类精度。

在主要参数设置上,采用 C_SVC 类型、RBF 核函数、多项式核中 degree 值为 3,惩罚系数为 1,损失函数

中 e 为 0.1,交叉验证次数为 10。

5.3 用户服务平台

用户服务平台采用 SSH 框架^[12],调用数据采集单元和数据计算平台接口,完成自动化数据采集和分类过程^[13-14]。分类结果如图 5 所示。

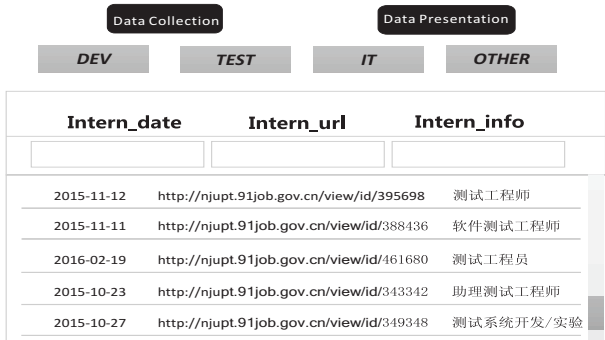


图 5 数据分类结果展示

当点击“Data Collection”和“Data Presentation”按键之后,招聘信息会经过采集、存储、分类等操作,在前端页面进行展示。图中所示为点击“TEST”按键后的结果展示,都是与测试工程师相关的工作岗位^[14],可见其能够完成数据的特征采集和招聘信息分类的功能。

5.4 数据存储单元

数据存储单元搭建 MongoDB 副本集并实现了读写分离功能。对副本集的集群设计如图 6 所示。

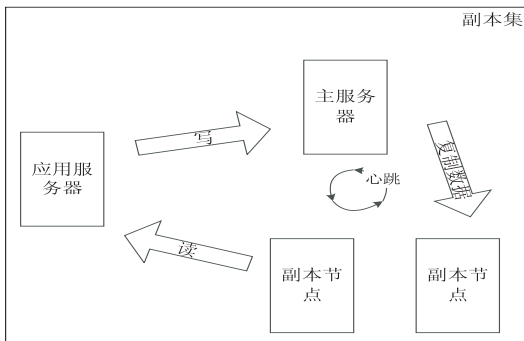


图 6 MongoDB 副本集设计

实验环境中主服务器选用一台性能卓越的机架式服务器, $id=1$ 。从服务器为两台 PC, id 值分别为 2 和 3。

因为主服务器上进行写操作,为防止数据因为误删等人工原因造成数据丢失,配置 id 为 3 的从服务器 ($slaveDelay:0$) 实时同步于主服务器, id 为 2 的从服务器 ($slaveDelay:3\ 600$) 每隔 3 600 s 同步于主服务器。

为了保证数据的安全性,设计 id 为 3 的从服务器 ($hidden:true$),从而不能被外界程序访问,并且设置 ($priority:0$) 表示当主服务器宕机后,该从服务器将不参与新的主服务器的选举。

当服务器发生宕机等突发事件时,数据访问端会依次按照优先级顺序切换到备份服务器上,从而使得数据访问具有容错性和实时性。

6 结束语

在如今的大数据时代,就业咨询智慧服务系统旨在通过分析海量数据为应届生求职方向提供正确的决策建议。Scrapy 框架完成数据采集,认知计算相关算法完成数据分类, MongoDB 集群用于海量数据存储。系统具备海量数据计算能力,能够有效进行特征采集和招聘信息分类的工作,能对学生能力进行准确定位,可有效为院校学生求职岗位的选择提供智能化的辅助咨询服务。

参考文献:

[1] 马旭. 探究 Tomcat 虚拟路径功能应用[J]. 中国新通信, 2016(2): 67.

[2] Kouzis-Loukas D. Learning scrapy[M]. Birmingham, UK: Packt Publishing Ltd, 2016.

[3] 阿培丁. 机器学习导论[M]. 北京: 机械工业出版社, 2009.

[4] Liu Chaoping, Li Feng. The design and implementation of exquisite course website[C]//International symposium on information technology in medicine & education. [s. l.]: [s. n.], 2012: 341-344.

[5] 邓珍荣, 唐兴兴, 黄文明, 等. 一种 Web 服务器集群负载均衡调度算法[J]. 计算机应用与软件, 2013, 30(10): 53-56.

[6] Harrington P. Machine learning in action[M]. Greenwich, CT: Manning, 2012.

[7] Zrelli S, Ishida A, Okabe N, et al. ENM: a service oriented architecture for ontology-driven network management in heterogeneous network infrastructures[C]//Network operations and management symposium. [s. l.]: IEEE, 2012: 1096-1103.

[8] 刘石忠. 云计算在智能化城市体系中的应用[J]. 无线互联科技, 2012(11): 32.

[9] Sebastiani F. Machine learning in automated text categorization[J]. Journal of ACM Computing Surveys, 2002, 34(1): 1-47.

[10] 霍福华, 尹宇孚. 基于 J2EE 架构的五层 Web 开发模型研究[J]. 通讯世界, 2017(1): 225-226.

[11] 霍多罗夫, 迪洛尔夫. MongoDB 权威指南[M]. 程显峰, 译. 北京: 人民邮电出版社, 2013.

[12] Kim H, Howland P, Park H, et al. Dimension reduction in text classification with support vector machines[J]. Journal of Machine Learning Research, 2005, 6(1): 37-53.

[13] 闻剑峰, 石屹嵘. 以分布式计算实现电信数据分析业务加速的研究[J]. 电信科学, 2012, 28(2): 22-26.

[14] Zhao Wei, Li Ming, Liu Jinhua, et al. Design and implementation of national meteorological computing resource management system based on grid[C]//International conference on information science and engineering. [s. l.]: [s. n.], 2012: 182-185.