東南大學

毕业设计(论文)报告

题　目 基于机器学习的 Webspam 检测方法

软件学院　院（系）　软件工程　专业

学　　号　71Y14125

学生姓名　李昌懋

指导教师　杨望

起止日期 2018 年 1 月至 2018 年 6 月

设计地点　东南大学软件学院

东南大学毕业（设计）论文独创性声明

本人声明所呈交的毕业（设计）论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得东南大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

论文作者签名：

日期：　　年　月　日

+

东南大学毕业（设计）论文使用授权声明

东南大学有权保留本人所送交毕业（设计）论文的复印件和电子文档，可以采用影印、缩印或其他复制手段保存论文。本人电子文档的内容和纸质论文的内容相一致。除在保密期内的保密论文外，允许论文被查阅和借阅，可以公布（包括刊登）论文的全部或部分内容。论文的公布（包括刊登）授权东南大学教务处办理。

论文作者签名：　　导师签名：

日期：　年　月　日　　日期：　年　月　日

摘要

webspam 是目前一种主要的 web 安全问题，通过在网页中嵌入 spam 信息，达到地下产业搜索引擎优化的效果。 本课题计划基于自然语言处理和机器学习算法，实现一种高效准确的检测方法对于 webspam 进行检测。该检测方法对于网页进行了格式化文本的解析，通过自然语言处理和机器学习算法建立了一套基于格式标记和词袋的特征集合。基于已标记样本训练机器学习算法，通过特征工程选择最有效的测度，实现高效的 webspam 机器学习检测方法。 具体方法是对于网页内容进行分块处理，基于对标签等格式特征的标记提取特征矩阵，使用机器学习方法基于已标记的特征矩阵进行训练，得到网页结构的特征模型，并使用自然语言处理中优势率的算法对块文本进行处理，处理后将文本优势率特征和网页结构特征模型结合起来进行检测。除此之外，本论文设计的检测方法具有一定学习能力，主要设计是通过增量学习的方法使模型不断适应新的格式特征数据，同时根据文本分词在数据中出现的次数更新优势率字典，实现了具有对新数据学习功能的 webspam 检测算法。

关键词：Webspam，机器学习，自然语言处理，特征工程

Abstract

Currently, webspam is an important web security problem. Some malicious people embed spam information in the web page to make their underground industry have a good rank in search engines. This project realizes a method to detect webspam based on the Natural Language Processing(NLP) and Machine Learning(ML). The method extracts the web page and build a complete feature set according to marked data by using NLP and ML, trains machine learning model based on the marked data sample and selects the most effective feature set, finally realizes the effective webspam detection algorithm. The specific detail is to separate the web page into block and extract feature array based on some html format feature, such as tag, and use machine learning method to train the marked array data, and get the format feature model, then deal with text in the block by using Odds Ratio algorithm, after text analysis, combine the text Odds Ratio feature and format feature model to detect the webspam. Furthermore, the detection method can learn from the new data which realized by incremental learning. The method uses incremental learning to learn new html format data and at the spam time updates the Odds Ratio dictionary based on the occurrence time of the spam words and non-spam words and finally realize a webspam detection method with learning ability for new data.

KEY WORDS: Webspam, Machine Learning, Natural Language processing, Feature Engineering

目　　录

摘要 I
Abstract II
第一章　绪论 3
1.1　研究背景及意义 3
1.2　论文主要研究内容 3
1.3　论文章节安排 4
第二章　webspam 基本概念及类型 5
2.1　webspam 基本概念 5
2.2　webspam 类型 5
2.3　webspam 检测研究现状 6
第三章　基本概念和所用技术 7
3.1　HTML 超文本标记 7
3.2　独热编码(one-hot encoding) 7
3.3　优势率(Odds Value) 8
3.4　python 编程语言及其机器学习库 sklearn 9
3.5　增量学习(Incremental Learning) 11
第四章　系统框架设计 12
4.1　系统环境 12
4.2　系统设计 14
4.2.1　阶段开发原则 14
4.2.2　易用性原则 14
4.2.3　业务完整性原则 14
4.2.4　可扩展性原则 14
4.3　系统模块 14
4.3.1　数据获取模块 14
4.3.2　数据训练模块 14
4.3.3　数据检测模块 15
4.4　数据存储设计 15
4.4.1　网页分块的存储 15
4.4.2　网页结构特征的存储 15
4.4.3　网页结构特征矩阵的存储 15
4.4.4　停词表及优势率字典的存储 17
4.4.5　网页结构机器学习模型的存储 17
第五章　网页分块设计及特征提取 18
5.1　html 分块 18
5.2　生成 html 块中标签树标签集合的特征向量 18
5.2.1　生成 html 块中叶标签属性的特征向量 19
5.2.2　特征向量拼接和处理 19
5.2.3　文本内容的特征提取 20
第六章　机器学习算法选取与检测算法设计 21
6.1　机器学习算法选取 21

6.2 检测算法设计.....21
第七章 效果测试与分析.....23
7.1 测试标准.....23
7.2 测试方法.....23
7.3 测试结果.....23
7.4 讨论分析.....24
第八章 总结与展望.....26
8.1 总结.....26
8.2 展望.....26
致 谢.....28
参考文献.....29

第一章　绪论

1.1 研究背景及意义

2018 年 1 月第 41 次中国互联网络发展状况统计报告[1]指出，截至 2017 年 12 月，中国互联网用户数达到 7.72 亿，普及率达到 55.8%，超过全球平均水平（51.7%） 4.1 个百分点。亚洲平均水平（46.7%）为 9.1 个百分点。比 2016 年新增网民 4074 万人，提高 5.6 个百分点。截至 2017 年 12 月，中国网页数量为 2604 亿，同比增长 10.3 个百分点。其中，静态页面数为 199.6 亿，占页面总数的 75.6%。受众最广的三大网络应用分别是即时通信(93.3%)，搜索引擎(82.8%)，网络新闻(83.8%)。截至 2017 年 12 月底，搜索引擎用户达到 6.3956 亿，比 2016 年增加 3716 万，年增长率 6.2%，利用率 82.8%。网页的爆炸性增长对搜索引擎的准确度提出了更高的要求。一些虚假页面或者含有垃圾信息的网页降低了用户的搜索体验甚至给用户带来利益上的损害。如何处理非相关网页，质量低下网页，大量重复的页面成为了搜索引擎的重要课题。

由于搜索引擎在网络传播和营销方面的重要作用，很多个人网站，博客加入了广告联盟，网站的在搜索引擎上的排名直接与利益挂钩[4]。在商业或者个人利益的驱使下，搜索引擎优化(Search Engine Optimization，简称 SEO)诞生了。搜索引擎优化经过获取各种搜索引擎决定排名的算法来对目标网站进行相关操作从而提升目标网站在搜索引擎上的排名，最终提高用户访问量。有恶意牟利者对网页进行一些超出合理范围内的优化，经过欺骗搜索引擎的方式，使得网站在搜索引擎中的排名高于原本正常应该得到的排名，这种行为就叫做 webspam。

Webspam 降低了搜索引擎的搜索结果准确性，并且降低了用户的正常搜索体验。它的快速增长严重降低了搜索结果的质量，并毁坏了搜索引擎的良好声誉。它被公认为是互联网搜索的最大挑战。将 webspam 检测出来，能够极大的优化搜索引擎的排序结果。

1.2 论文主要研究内容

本项目的主要研究内容是基于自然语言处理和机器学习算法，实现一种高效的检测方法对 Webspam 进行检测。检测方法需要对网页进行格式化文本解析，并通过自然语言处理为每个网页建立一套基于格式标记和词袋的特征集合，然后基于已标记样本训练机器学习算法,并后通过特征工程选择最有效的测度,实现一个高效的 Webspam 机器学习检测方法。主要研究内容有以下几个方面:

（1）对 HTML 的分块处理。

是在 SGML 定义下的一种描述性语言，是一种描述文档结构的标记语言。由于 Webspam 只存在于网页中的一小部分，所以有必要将 HTML 分块进行处理。HTML 分块是本课题的关键问题之一。

（2）生成 HTML 块的特征向量。

为了在后面使用机器学习的算法，我们必须将这些 HTML 块抽象成特征向量。如何将 HTML 合理有效地转化成能够精确表示该块特征的特征向量也是本课题的难点之一。

(3) 对 HTML 块进行特征选择。

使用特征工程的方法对以上 HTML 块进行特征选择，将 HTML 分为 spam 和非 spam 块也是解决本问题的重中之重。

(4) 处理 HTML 块中的文本内容。

网页中垃圾文本内容是。使用自然语言处理技术将文本内容分类处理也是本课题的难点之一。

（5）选择合适的机器学习算法。

目前机器学习的算法多种多样，选择一个有效的机器学习算法是本课题成功的关键之一。

（6）对检测系统的设计。

设计一套完整有效的 webspam 检测系统是本课题的最终目标。该检测系统除了完成基本的检测功能之外，还需要有一定的自我学习功能。

1.3 论文章节安排

本论文按照课题研究和开发过程来组织全文内容，本文总共有八章。
各个章节的内容概要如下:

第一章 绪论，介绍了本课题的背景和意义，并给出了本课题的主要研究内容。

第二章 webspam 基本概念及类型，简要介绍 webspam 的基本概念及类型。

第三章 基本概念和所用技术，简要介绍本课题所用到的基本概念和技术。

第四章 系统框架设计，简要介绍本课题总体基本框架。

第五章 网页分块设计及特征提取，介绍并分析了提取网页块中特征的主要方法，主要包括提取网页结构特征和提取网页文本特征

第六章 机器学习算法选取与检测算法设计，主要目的是分析各个机器学习算法的优缺点，选取出适合用于 webspam 分类的机器学习算法并设计出检测算法。

第七章 效果测试与分析，通过对效果的测试，主要从正确率，误报率等指标进行总结，并进行模型优化。

第八章 总结与展望，对本文所写内容要点进行总结，同时给出下一步工作的展望。

第二章　webspam 基本概念及类型

2.1 webspam 基本概念

搜索引擎的设计者会用搜索引擎索引(Search engine indexing)来收集，解析和存储数据，从而建立一个快速准确的信息检索系统。索引的设计需要结合很多方面上的概念，比如语言学，数学，信息学，计算机科学，人体工程学等等。而 Webspam 就是通过故意利用这些设计上的漏洞，通过各种方式提升自己的网页排名。它有很多方法，比如重复一些不相关的词，提升索引资源的相关性等等。这些行为都和检索系统本来的目的是不一致的。

虽然有很多搜索引擎优化的方法的目的是提升搜索内容的质量和尽量显示对用户有用的内容，但 Webspam 的行为仍然被当作搜索引擎优化的一种。搜索引擎会使用很多种算法来决定相关性排序。它们中的一些算法包括决定是否搜索关键词会出现在网页的文本内容或者链接中。很多搜索引擎会检查 Webspam 的情况并且将可疑网页移出它们的排名。或许经常被用户抱怨搜索系统中的错误匹配，搜索引擎运营商会不得不研究技术找出有 Webspam 的网页并且将它们剔除排名中去。用一些不道德的手段将网页排名变高也被当作黑帽子搜索引擎优化(black hat SEO)归于到搜索引擎优化范围中。这些方法致力于破坏搜索引擎的规则和标准。但除此以外，这些垃圾注入者(spam perpetrators)也冒着被搜索引擎惩罚的风险。总体来说，Webspam 类型被分为内容 spam 和链接 spam。接下来我们具体讨论 Webspam 都有哪些类型。

2.2 webspam 类型

Gyöngyi 等对 webspam 进行了详尽描述。他们把通常的 spam 分成两种类型：第一类含有各类增加排名的方法，对搜索引擎的排序算法修改内容或链接；第二类含有各种隐藏 spam 信息的技术，比如隐藏文本内容、伪装文本内容和重定向文本内容等。根据 Gyöngyi 的论点，将 web spam 大致分成内容 spam(Content Spam)、链接 spam(Link Spam)和隐藏 spam(Hiding Spam)三个类型[4]。三种类型的具体内容如表：

表 2-1 webspam 的三种主要类型

类型	原理	主要方法
内容 spam	通过篡改网页内容信息欺骗搜索引擎的排名算法	关键词堆砌、元标签填充
链接 spam	构造一套复杂的链接结构以获取更高的网页排名	链接农场、链接交换、蜜罐诱饵、维基百科 spam、Splogs
隐藏 spam	把篡改网页的内容或者链接隐藏，使用户看不到但是搜索引擎爬虫可以爬取到	文本内容隐藏、Cloaking、重定向

2.3 webspam 检测研究现状

根据 webspam 的类型，有三种不同的解决方法，它们分别是根据内容进行 webspam 检测，根据链接进行 webspam 检测，对于隐藏的 spam 进行检测[4]。根据内容进行 webspam 检测的主要原理是通过分析一些内容属性如关键字来对 spam 进行匹配。根据链接进行 spam 检测主要是有一些著名的算法，如 Gyöngyi 等人提出的 TrustRank 算法[4]。

根据检测 webspam 的方法，可以分为规则检测法和机器学习检测法。规则检测法就是用一些固定的规则，比如 spam 关键词匹配。关键词匹配是指人工先筛选出具有 spam 特征的关键词，然后对网页进行爬取匹配。它同时也属于根据内容进行 spam 检测。虽然这种方法漏报率较低，但是这种方法的缺点是误报很多甚至在结果中一半的内容都是误报。随着机器学习领域的发展，有一些机器学习的检测方法也用于 webspam 检测。机器学习的相关方法往往准确率高，并且误报率低，本文正是采取机器学习的方式对 webspam 进行检测。

第三章　基本概念和所用技术

3.1 HTML 超文本标记

HTML(Hypertext Markup Language)超文本标记语言是一种用来创建网页的标准标记语言。HTML，CSS 和 Javascript 形成了万维网技术的基石。网页浏览器从网络服务器或者本地存储于获取到 HTML 文档，然后再讲文档转化为多媒体的网页内容。HTML 在语义上描述了一个网页的结构。HTML 元素是 HTML 页面的构造块。在网页构造的时候，图片和其他互动形式的对象会嵌入到加载页中。HTML 提供了一种方法通过结构化的文本语法去创建结构化的文档，比如标题，段落，列表，链接，引用等。HTML 定义了标签来描述网页的格式和特性, HTML 标签的书写格式如下 :<tagname >内容< tagname/>。从 HTML 的总体结构上 , 可以把 HTML 分为 HEAD 部分和 BODY 部分。HTML 文本的框架结构一般由 TABLE ,DIV , TR , TD , FRAME 等标签决定 ,整个网页被这些标签分为表或者块, 表或者块的布局 、再构成网页的架构。跟踪这些标签就可以勾划出整个网页的结构。像 <img />和<input />之类的标签就是直接将内容填到网页。其他像<p>的标签会包含文本信息的并且可能会有子标签。浏览器不会显示标签，但是会用它们来解释出网页内容。

在 HTML 中还能够使用像 JavaScript 的脚本语言嵌入一些程序逻辑。这些程序逻辑会影响到网页的行为和内容。CSS 定义了网页的各种外观属性。作为 HTML 和 CSS 标准的维护者，W3C 联盟从 1997 年开始就鼓励使用 CSS 来改变网页外观，而不是显式的写在 HTML 中。

HTML 的标准经过很多变迁。从 1996 年开始 HTML 说明就一直被 World Wide Web Consortium(W3C 联盟)所维护。然而，2000 年，HTML 也成为了一种国际标准(ISO/IEC 15445:2000))。HTML 4.01 在 1999 年晚些时候发布，之后又在 2001 年出了修订版。2004 年，HTML5 在 Web Hypertext Application Technology Working Group(WHATWG) 开始发展，在 2008 年和 W3C 宣布联合，在 2014 年 10 月 28 日完成标准化工作。由于本文对象主要是教育网的网站，而教育网网站中包含有 HTML4 和 HTML5 标准的网页，因此我们会兼顾 HTML4 和 HTML5 标准的标签，不会刻意区分。

HTML 的结构主要包括标签、内容和属性。属性是用在标签里来控制标签的行为。HTML 属性是一个 HTML 元素的修饰者。一个属性或者定义了一个元素默认的功能或者提供了一些如果没有他们就不能正确执行的功能。在 HTML 语法中，这些属性会被添加到 HTML 标签的起始标签中。一些属性是所有标签都共用的，而另一些属性是只有特定的标签有的。有些标签使用相同的标签可能有不同的功能。后面我们会具体分析如何将这些特征转化为算法可用特征。

3.2 独热编码(one-hot encoding)

独热编码原来是电子信号编码领域中的概念，是指一组只有 0 和 1 的合法数字位。在机器学习中常常被用来对数据进行预处理。每一位的 0 和 1 分别表示某个特征是否存在。把这些特征的存在与否用 0 和 1 表示并形成特征矩阵。因为实际的机器学习中，特征不一

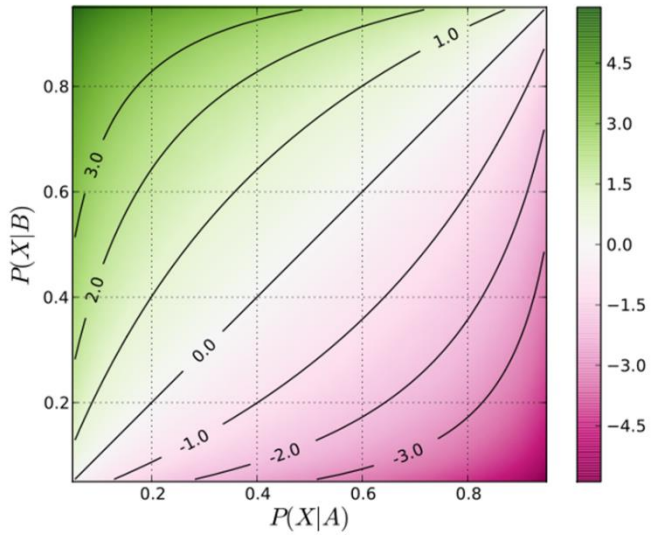
定是连续值，有时候可能有枚举值。此时使用独热编码进行数据处理是一种好的选择。比如性别可以分为男性和女性等等。此时就可以使用二位独热编码 0 和 1 来标识男性还是女性。类似这样，我们可以将特征进行数字化或者矩阵化，变为机器学习算法可以训练的数据。比如有三个特征属性如下：性别：["男"，"女"]，所在地区：["欧洲"，"美国"，"亚洲"]，所用浏览器：["火狐"，"谷歌"，"赛风"，"IE"]。针对其中某个样本，如["男"，"美国"，"IE"]，要将此枚举类型的特征矩阵化，我们可以采用独热编码的方式对上述的样本编码，“男”对应着[1，0]，“美国”对应[0，1，0]，“IE”对应[0,0,0,1]。那么合并矩阵后的特征矩阵化的结果为：[1,0,0,1,0,0,0,0,1] 。

在本文中对于网页结构特征的提取会采取如上做法。以标签特征为例，网页标签是有限个的，我们假设总共的标签集合为[a, body, div, head, img, li, link, meta, script, style, table, ul, video]，如对于某部分网页内容的标签树为:a-li-table-ul-div-body，则根据独热编码的方法，就可以生成这部分网页内容标签的特征矩阵为 [1,1,1,0,0,1,0,0,0,0,1,1,0]。这种方法的优点是能够处理非连续型数值特征，而且在一定程度上也扩充了特征。比如性别本身是一个特征，经过 one hot 编码后，就变成了男或者女两个特征。而这种方法也有一定的缺点，当特征类别较多的时候，数据经过 one-hot 编码可能会变得很稀疏。也是因为这个原因，所以在本课题中处理网页文本特征的时候没有使用 one-hot 编码方式。

3.3 优势率(Odds Value)

在统计学上，优势率(Odds Value)是在一定数据集中统计属性 A 的出现程度和属性 B 的出现程度之间的数量关系。如果数据集中每个元素都有或者没有 A，并且有或者没有 B，在这里这两个属性都进行了适当的定义，那么有一个比值可以描述 A 的出现程度和 B 的出现程度的数量关系。这个比值我们叫做优势率。优势率可以用下面的步骤进行计算:（1）在含有 B 的元素统计中有 A 的元素个数并计算它们的比值。(2) 在不含有 B 的元素中统计有 A 的元素个数并计算它们的比值。(3)用步骤 1 的结果除以步骤 2 的结果得到优势率。

如果比值大于 1，那么 A 就被认为和 B 有联系。但是要注意 B 不一定是 A 的生成原因。比如也可能有个 C，造成了 A 和 B。优势率比较了在一个特定条件下一个结果的发生



情况和如果没有这个特定条件结果发生的情况。从技术角度讲，优势率测量的是某两种二元数显之间的影响大小。它在 logistic regression 具有重要作用。见示意图 3-1。

图 3-1　A、B 优势率关系示意图

优势率特征选择方法又称几率比特征选择方法[12]。它的特征是只把训练文本分为两种类型，一种是正面类，可以被叫做主题类或目标类，即和指定主题一致的文档，而其他的文档都归为负面类[12]。因此优势率特征选择方法是一种二元特征选择方法。它通过词条对正面样本的贡献与对负面样本的贡献进行对比，得到特征评估函数如下：

$$OR(t) = \log(\frac{odds(t|pos)}{odds(t|neg)}) = \log(\frac{P(t|pos)\left(1-P(t|neg)\right)}{P(t|neg)\left(1-P(t|pos)\right)}) \tag{3-1}$$

其中 OR(t) 为词条 t 的优势率分值， P(t|pos) 表示正面样本中出现词条 t 的概率，而 P(t|neg) 表示负面样本中出现词条 t 的概率。从公式(3-1)中可以看出，优势率方法注重于对目标类的评估，词条 t 在正面样本中概率越高，在负面样本中出现概率越低，则该词条 t 对目标类的分类贡献度越大[12]。

在本文中，设 A 为词条 t 和 spam 样本一同出现的网页块个数，B 为词条 t 和非 spam 样本一同出现的网页块个数， C 为所有的 spam 样本的网页块个数， D 为所有非 spam 样本的网页块个数。则能够将公式变形为：

$$OR = \frac{A(D-B)}{B(C-A)} \tag{3-2}$$

由于优势率方法专注于目标类，而把其他不相关的文档都标记为负面类，这种二元性质使得它特别适合用于 webspam 词汇的特征提取及文档描述。从公式(3-2)中我们可以看出，优势率方法给那些只出现在目标类中而几乎不出现在负面类中的词条打高分。而且由于没有考虑词条在文档中的出现的次数，即词频，而只考虑了词条的文档频率，使得这个评估函数倾向于不选择高频的词作为文本的最佳特征，因为高频词条通常会出现分布在多个类中，而这恰恰是 webspam 词汇的特征，即总体频率较低。

3.4 python 编程语言及其机器学习库 sklearn

本课题所用编程语言为 python。python 语言具有易使用，可移植性强，可扩展性和可嵌入性强，并且代码库丰富的特点。它的特性用三个词概括就是: “优雅”、“明确”、“简单”。python 版本有 2 和 3，由于版本 2 的代码库较为丰富，且使用者最广泛，所以本课题采取版本 2。特别是对于机器学习来说，python 对于其他语言具有如下优势：（1）Python 具有完善的程序资源库。（2）Python 在机器学习领域被广泛运用。（3）Python 可以多平台运行而且是开源的。

图 3-3 机器学习库 sklearn

sklearn(图 3-3)是基于 numpy 和 scipy 的一个 python 机器学习算法库。sklearn 如果不是最流行的机器学习算法库，那么也算得上是最流行的机器学习算法库之一。它拥有大量的数据挖掘和数据分析功能，使其成为研究人员和开发者的首选库。它设计的非常优雅，它让我们能够使用同样的接口来实现所有不同的算法调用。本文主要使用其中的监督学习的分类方法如 svm，naive_bayes 等。以 svm 为例，其继承关系如图 3-2：

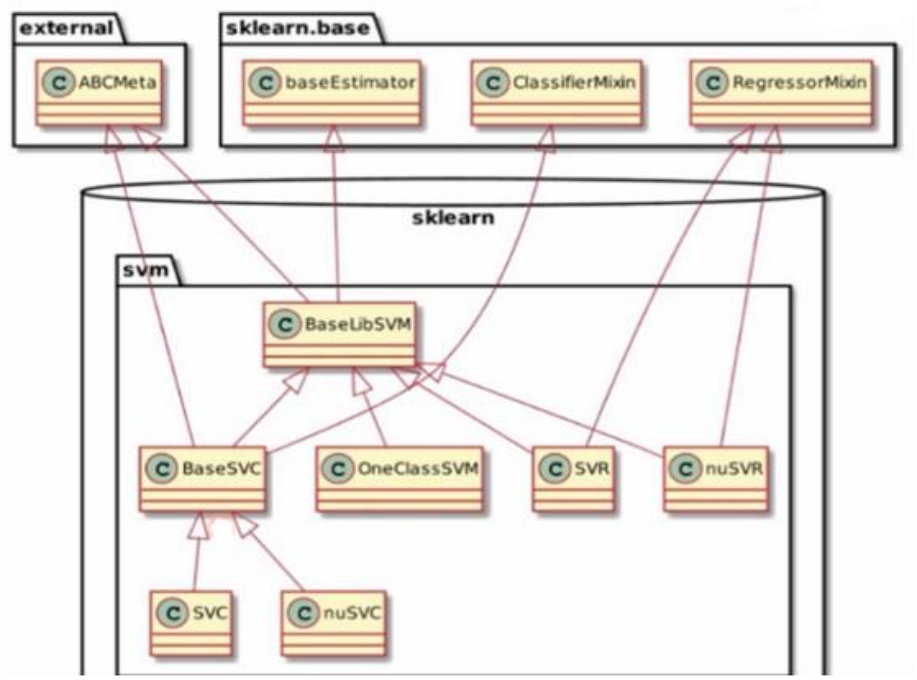


图 3-2 sklearn 中 svm 继承关系图

这里 SVC 即 support vector classifier，SVR 即 support vector regression，svm 既可以作为分类器，也可以作为回归器，所以，它们分别继承实现了 ClassifierMixin 和 RegressorMixin。

在 sklearn 里面，我们可以使用完全一样的接口来实现不同的机器学习算法，通俗的流程可以理解如下：

（1）数据加载和预处理

（2）定义分类器（回归器等等），譬如 svc = svm.svc()

（3）用训练集对模型进行训练，只需调用 fit 方法，svc.fit(X_train, y_train)

（4）用训练好的模型进行预测：y_pred=svc.predict(X_test)

（5）对模型进行性能评估：svc.score(X_test, y_test)

3.5 增量学习(Incremental Learning)

增量学习是一种机器学习方法，指一个学习系统能不断地从新样本中学习新的知识，并能保存大部分以前已经学习到的知识。增量学习非常类似于人类自身的学习模式。在增量学习中输入数据连续地被用来扩展已存在模型，即进一步训练模型。它是一种监督学习和非监督学习的技术。它被用于数据随着时间变化而变化的情况或者数据的大小超过系统内存。可以使用增量学习的算法被称为增量机器学习算法。很多传统的机器学习算法本子上支持增量学习，其他算法可以通过一些适配来实现它。增量学习的主要目的就是让学习模型在不忘记已有数据模型的情况下适应新数据。它没有重新训练模型。增量学习进成被用来处理数据流或者大数据。增量学习主要表现在两个方面：一方面由于其无需保存历史数据，从而减少存储空间的占用；另一方面增量学习在当前的样本训练中充分利用了历史的训练结果，从而显著地减少了后续训练的时间。增量学习主要有两方面的应用:一是用于数据库非常大的情形,例如 Web 日志记录;二是用于流数据,因为这些数据随着时间在不断的变化,例如股票交易数据.

在本课题中由于我们会将网页分块，分块的数量远远大于本身的网页数量，而且从根本上来说 webspam 检测也是一种流数据检测，webspam 的类型会随着时间的变化而变化，所以需要用增量学习的手段来解决这个问题。

第四章　系统框架设计

方案总体基本框架如下: 从服务器数据库中获取要检测网页链接的原始数据，然后爬取网页内容，对 HTML 进行分块，每一块主要分为标签、内容和属性。将分好的 HTML 块进行人工分类，分为 spam 和非 spam 的内容块，对于处理后的内容块生成其特征向量。选取一种或者几种机器学习算法，将特征向量输入到算法中训练构建分类器模型，评价分类器效果，调整模型。选取最优的分类器模型，完善 spam 监测系统，用实际数据进行测试，评价效果。

图 4-1 为方案流程图:

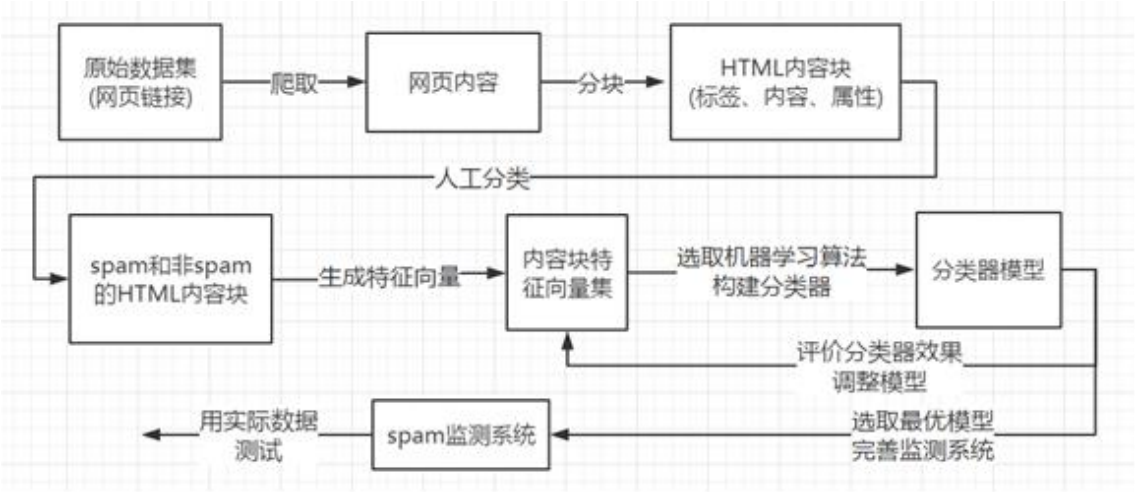


图 4-1 方案流程图

4.1 系统环境

本课题使用的编程语言是 python，版本是 2.7.1。操作系统:Windows，Linux，MacOS 均可，无特殊要求。其中 python 除了 sklearn，numpy 等机器学习模块以外需要安装的外来模块有:结巴分词器——用来进行中文分词，BeautifulSoup——用来进行网页结构解析，csv 生成器——生成特征向量的 csv 文件。下面简要介绍一下这几个模块。

结巴分词器是目前最流行的中文分词工具之一。开发者可以指定自己自定义的词典或者停用词典。它支持的分词模式如表 4-1 所示。

表 4-1 结巴分词器支持的分词模式

分词模式	方法	适用场景	优势	劣势
精确模式	将句子最精确地切开	文本分析	——	——
全模式	将句子中全部能组词的词语分出	——	速度非常快	不能解决歧义
搜索引擎模式	对长词切分	搜索引擎分词	提高召回率	——

BeautifulSoup(图 4-2) 是目前最流行的 html 解析工具之一。它可以从 HTML 或 XML 文件中提取数据。它的 4 种解析器及其适用场景见如表 4-2。由于本课题需要兼容 HTML5，并且需要极好的容错性，因此选用 html5lib 解析器。

表 4-2 BeautifulSoup 的解析器

解析器	使用方法	优势	劣势
Python 标准库	BeautifulSoup(markup, "html.parser")	Python 的内置标准库，执行速度适中，文档容错能力强	Python 2.7.3 or 3.2.2)前 的版本中文档容错能力差
lxml HTML 解析器	BeautifulSoup(markup, "lxml")	速度快，文档容错能力强	需要安装 C 语言库
xml XML 解析器	BeautifulSoup(markup, ["lxml", "xml"]) BeautifulSoup(markup, "xml")	速度快，唯一支持 XML 的解析器	需要安装 C 语言库
html5lib	BeautifulSoup(markup, "html5lib")	最好的容错性，以浏览器的方式解析文档，生成 HTML5 格式的文档	速度慢，不依赖外部扩展

图 4-2 网页解析库 BeautifulSoup

CSV(Comma-Separated Values)即逗号分隔值，可以用 Excel 打开查看。由于是纯文本，任何编辑器也都可打开。与 Excel 文件不同，CSV 文件中：（1）值没有类型，所有值都是字符串。（2）不能指定字体颜色等样式。（3）不能指定单元格的宽高，不能合并单元格。（4）没有多个工作表。（5）不能嵌入图像图表。在 CSV 文件中，以,作为分隔符，分隔两个单元格。像这样 a,,c 表示单元格 a 和单元格 c 之间有个空白的单元格。依此类推。在本课题中，为了将所提取的特征矩阵持久化存储，以便于直接取用。我们需要使用 python 的

csv 模块进行对 csv 文件的读取和写入。

4.2 系统设计

系统设计是根据系统分析的结构，运用系统科学的思想，设计出能最大限度满足要求的目标的新系统的过程/。系统设计共有几大原则，它们分别是:阶段开发原则，易用性原则，业务完整性原则，可扩展性原则。下面具体讨论改检测系统完成这项原则的情况。

4.2.1 阶段开发原则

阶段开发原则是指具体的开发分阶段进行，对于本检测系统的开发过程可以采取以下几个阶段：一是搭建基本的框架和环境，如 python 环境，机器学习框架等。二是数据获取模块的开发，三是数据训练模块的开发，四是数据检测模块的开发。

4.2.2 易用性原则

易用性原则是指减轻使用人员的负担，有一些自动化的功能。对于本检测系统来说，本检测系统自动化的收集数据，分析数据，训练数据，检测数据，学习新数据，极大的方便了使用人员。

4.2.3 业务完整性原则

业务完整性原则是指对于任务进行中的特殊情况及时，正确的响应，保证业务数据的完整性。对于本检测系统来说，具有完善的异常捕获及处理机制，能够及时的解决由于数据异常导致的系统不稳定。

4.2.4 可扩展性原则

可扩展性原则是指系统设计要考虑到未来发展的需要，各个功能模块之间耦合度小，便于系统的扩展。对于本检测系统来说，数据获取，数据训练，数据检测全部是分开的模块，中间只有规定好的接口，扩展性得到了充分保证。除此之外，由于特征提取模块耦合度低，特征筛选也便于进行。

4.3 系统模块

本系统分为数据获取模块，数据训练模块和数据检测模块。如果作为学习开发用途，使用者可以使用所有模块，如果只是为了检测，那么使用者只使用数据检测模块即可。

4.3.1 数据获取模块

数据获取模块主要是根据网页链接，对教育网内容进行爬取，并将所爬取的网页内容传到数据训练模块或者数据检测模块。如果是训练数据，则需要存入磁盘，训练时再读入，如果是检测数据，则只需要将内容传入检测模块即可。

4.3.2 数据训练模块

数据训练模块主要是根据从上一模块得出的数据，生成特征矩阵及自然语言模型，代入相关算法进行训练然后得到特征模型。开发者可也可以使用自己选择的机器学习算法进行训练。本文所用的机器学习算法是先验为多项式分布的朴素贝叶斯。后文我们会讲到选择它的原因。

4.3.3 数据检测模块

从数据获取模块获取到新的网页，根据下文所描述的检测算法进行检测。根据检测结果将修正模型。所用具体算法如本文 6.2 所述。

4.4 数据存储设计

本课题使用文件系统进行存储。主要原因是因为文件系统不需要额外的数据库连接等操作，直接读写明显比连接数据库读写效率要高，虽然有不易索引的缺点，但是对于本文的数据处理阶段的数据没有索引的需要，所以直接使用文件系统进行存取和处理是最佳选择。

4.4.1 网页分块的存储

网页分块之后，需要把每一块的信息以块为单位持久化存入文件系统中，以便后续的读取和处理。其中，我们所需要的块内信息有标签，文本及标签属性。把这些信息以健值对的方式用 python 代码直接存入文件系统。具体示例在后文阐述。

4.4.2 网页结构特征的存储

所用网页结构的机器学习特征需要存入文件系统。有两种特征，一种是网页的 HTML 标签，另一种是网页标签属性的存储。在使用 one-hot 特征编码的前提下，对于 HTML 标签的存储，由于 HTML 标签的数量是有限个，因此我们直接将所有标签按一行一个标签存入文件，读取时用列表读取，读取后的列表可以对数据进行生成特征矩阵等操作。同样也是使用 one-hot 特征编码，对于属性标签的存储，属性标签分为健和值的存储。由于键也是有限个，存储方法和 HTML 标签存储方法一致。对于值会进行相关筛选后存储。筛选后的存储方法和 HTML 标签存储方法一致。图 4-4(a)为标签存储片段，图 4-3(b)为属性键存储片段，图 4-4(c)为某属性值存储片段。

4.4.3 网页结构特征矩阵的存储

特征矩阵是用于机器学习算法的输入值，因此我们需要将特征矩阵进行存储。机器学习常用的数据存储文件是前文介绍的 CSV 文件，因此本课题也同样使用 CSV 文件对特征矩阵进行存储，方法就是通过调用 CSV 代码库生成 CSV 文件，当需要训练时再读取并输入到相关机器学习算法中。

```
a
abbr
acronym
address
applet
area
article
aside
audio
b
base
basefont
bdi
bdo
big
blockquote
body
br
button
canvas
```

(a)

```
accesskey
class
dir
id
style
title
charset
href
name
rel
rev
shape
target
type
align
height
width
border
bgcolor
```

(b)

```
line-height
font-family:宋体
color:rgb(0,0,0)
text-decoration:none
FONT-SIZE
LINE-HEIGHT
TEXT-INDENT
overflow:hidden
position:relative
margin-top
padding-left
width
height
background-color:#dfe7f5
border:#94afde
padding-top
color:#1C74B7
display:none
CURSOR:hand
```

(c)

图 4-4 网页结构特征存储片段

4.4.4 停词表及优势率字典的存储

停词表中的词是分词时会被忽略的词。停词表是由每行一个词的多行文本文档所存储。优势率字典中每一行会存入一个分词及这个在正向文档及负向文档出现的次数还有它的优势率，值与值之间使用空格隔开，以方便读取。图 4-5(a)为优势率字典片段。图 4-5(b)为停词表片段。

```
1  汽枪 288 0 5627.87103059
2  皇冠 274 0 5340.67417913
3  秃鹰 101 0 1916.48796992
4  百家乐 163 1 1558.85774059
5  打鸟 70 0 1326.29751448
6  气枪 58 0 1099.66846914
7  波音 56 0 1061.99552656
8  汽狗 55 0 1043.16958628
9  太阳城 53 0 1005.53874814
10 比分 51 0 967.93594041
11 鸟枪 47 0 892.814291031
12 真钱 45 0 855.295386905
13 投注网 44 0 836.546401339
14 投注 166 3 794.123929591
15 麻将 39 0 742.905982906
16 棋牌 78 1 738.967153285
17 配件 37 0 705.498514116
18 世博 36 0 686.805199629
19 开户 107 2 677.15769665
```

(a)

```
1  2017
2  text
3  charset
4  gb2312
5  html
6  !
7  "
8  #
9  $
10 %
11 &
12 '
13 (
14 )
15 *
16 +
17 ,
18 -
19 --
```

(b)

图 4-5 相关字典片段

4.4.5 网页结构机器学习模型的存储

生成机器学习的模型后，需要将模型持久化存储起来，以便之后使用。这里我们使用的是 sklearn 库中的 joblib。对于大数据而言，joblib 存储模型非常高效，且容易使用。如果想要再次使用同一模型，我们可以使用 joblib 读取之前存在磁盘中的模型，并将其赋值于 sklearn 库中机器学习的模型对象上。图 4-6 为机器学习模型存储文件示意图。

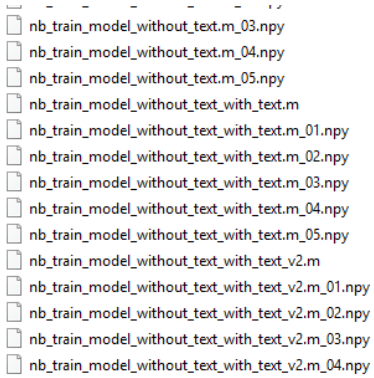


图 4-6 机器学习模型存储文件示意图

第五章　网页分块设计及特征提取

5.1 所选特征集合

本文主要是利用 html 网页的一些基本特征来进行模型搭建。这些特征包括网页的标签，网页的属性及网页的文本内容。其中网页的标签和网页的属性我们统称为网页的结构特征，网页的文本内容我们称为网页的文本特征。本文会对网页的结构特征和网页的文本特征分别搭建模型，将所搭建的模型相结合得出检测算法。如表 5-1 为所用的特征集合表。如图 5-1 为所选的网页结构特征维度大小图。

表 5-1 所用的特征集合

特征类别	特征内容	主要处理方法
网页结构特征	Html 标签树和 Html 标签属性	机器学习搭建模型
网页文本特征	Html 文本内容	自然语言处理搭建模型

5.2 html 分块

Html 分为 html 标签和标签的属性及标签中的内容。考虑到需要针对 webspam，一个 html 块要包括标签树的标签集合，叶标签(即 html 树中没有子标签的标签)的属性以及叶标签中的文本内容。首先所有的 html 一定是分为两大块 head 和 body，分完之后我们根据 head 和 body 中的具体情况再进行细分。

head 中情况比较简单。主要有 meta 和 title 中的信息我们需要关注。其中 title 中的信息可以直接做简单的分词处理。meta 中的信息存在于它的属性中，所以我们需要提取出 meta 标签的属性中的文本。

Body 中情况比较复杂。我们可以根据标签是否有文本内容来分块。有文本的内容的标签比如<p>，我们提取出标签树，内容和叶标签的属性。没有文本内容的标签如<img>，我们提取出标签树和该标签的属性。分块后的内容根据需要存入到文件系统中，每一块的存入格式，如图 5-1 示例：

```
tag:-a-div-body-html-[document]
elem:红利来木雕
attrs:{u'href': u'http://www.lu838.info/list/?1.asp'}
```

图 5-1 块存入格式示例

其中，tag 表示标签树，elem 表示文本内容，attrs 表示叶标签属性

5.3 生成 html 块中标签树标签集合的特征向量

Html 的标签是有限个的，使用枚举类型将每一个标签编号，然后以有为 1 无为 0 将标签是否存在表示出来生成特征向量。不考虑表示标签之间的父子关系，因为对于 spam 来说，其没有明显特征。比如<table>， 等外层标签可以互相嵌套很多层，但是他们的父

子关系实际上难以对 spam 的判断提供帮助。另外，一些与 spam 的判断有关系的内层子节点之间的父子关系往往是固定，比如如果<p>和<a>同时出现，那<a>一定是<p>的子节点。所以如果考虑父子关系的话，反而会把简单问题复杂化。

5.3.1 生成 html 块中叶标签属性的特征向量

叶标签属性是键值对。如果没有为 0。因为属性的键是有限个的，所以使用枚举类型将每一个属性键编号，有为 1 无为 0。属性的值共分为 5 类，分别是数值型，健值对类型，链接型，HTML 内部值类型，自定义字符型。其中数值类型的特征是只有数字或者数字后出现 px,em,ex,%,in,cm,mm,pt,pc 等单位的值。键值对类型的特征是";"将键值与键值之间隔开，":"将键与值之间隔开，比如 style 属性的值是 CSS 的健值对代码。链接型的特征是 http 或者/组成的链接。HTML 内部值类型是 HTML 中属性本身具有枚举类型的值，如 target 里有_blank,_parent,_self,_top 等可供选择的值。自定义字符类型是用于标题或者内容或者给当前标签做记号的值如 title,id 等。以上 5 种类型的值需要进一步的选择后再生成特征向量。键和值应该一一对应，所以我们应该为每一个键生成一个一维特征向量表示出现在该键的属性，如果在这个块中没有这个键那么是 0 向量。值共有下面 5 种类型：.

(1) 数值型，一般后面有单位 px 等,如 height。

数值类型的特征是只有数字或者数字后出现 px,em,ex,%,in,cm,mm,pt,pc 等单位的值。我们不选择数值型属性的值作为特征，因为数值属性的值灵活度高，每一个 html 基本都有其定制化的大小和长度，特征不明显。所以我们只是在之前生成键特征向量的时候标识一下有此键即可。

(2) 健值对类型，如 style，style 属性的值是 CSS 的健值对代码。

健值对类型的特征是";"将键值与键值之间隔开，":"将键与值之间隔开。我们考虑键值对类型的值作为特征，属性中的键值也有键和值之分，其中值的类型大体有数值类型，函数类型，内部值类型三种类型，数值类型我们一样不考虑值只考虑键，函数类型的值和内部值类型我们直接和键一起作为一个特征项，因为其内容相对固定且具有明显特征。

(3) 链接型，如 href, 链接型的特征一般是有 http 或者/组成的路径组成。

我们不能把具体的链接当做特征项，因为链接本身不具有明显特征，所以我们只存入键不考虑链接型的值。

(4) 自定义字符型，如 title。

自定义字符类型按照自然语言处理的方法进行处理。本课题使用优势率算法。

(5) HTML 内部值类型，

比如 target 里有_blank,_parent,_self,_top 等可供选择的值，再比如 font-family 中有宋体等值。内部值类型由于是有限个，给其编号后有为 1 无为 0 存入。

5.3.2 特征向量拼接和处理

对于上述生成的特征向量，我们需要将它们拼接成为一个特征矩阵才能输入到机器学习算法中。用 one-hot 编码方式，特征矩阵中有标签树信息，属性键信息和属性值信息。对于机器学习算法来讲，特征矩阵的列数必须是定值，即对于不同的对象，必须有固定个特

征，因此我们需要将矩阵进行调整，将缺省的属性用 0 补齐，从而得到每一块的特征矩阵。将每一块的特征矩阵一行一行的拼接得到所有训练数据的特征矩阵。其中每一个行第一位的 0 或 1 表示该训练对象也就是该块是否是 spam，即机器学习算法的输出结果值，1 为 spam，0 为非 spam。生成特征向量后，将特征向量按照前述存储方式存入磁盘文件系统中，以便于重复使用。

5.3.3 文本内容的特征提取

对文本内容的特征提取有好多已有的方法如卡方，信息熵等。对于 spam 这一问题，经过分析，本论文选择了一种叫优势率的方法。优势率，正如前文所介绍的，是专门用于二元分类的一种分类算法，这种算法可以很好的突出正向类(这里指我们需要分出的 spam 类)，弱化负向类(这里指非 spam 类)。选择优势率的原因如前文所述, 考虑到 Webspam 的特征。

根据优势率的计算公式我们可以通过计算样本中词条 t 和 spam 样本同时出现的块数目、词条 t 和非 spam 样本同时出现的块数目、所有的 spam 样本的块数目、所有非 spam 样本的块数目得出词条 t 的优势率。

对于每一个网页的头部信息，我们把文本提取出来并分词计算平均优势率作为这一部分的代表优势率。对于每一个 html 块，我们将里面所有的文本提取出来后分词计算平均优势率作为这一块的代表优势率。根据已得到的优势率会有两个阈值。优势率有两个阈值，大于某个阈值一定是 spam 词，这个阈值我们叫做绝对阈值，小于某个阈值一定不是 spam 词，这个阈值我们叫做可能性阈值。后面本文会具体阐述如何根据这两个阈值并结合网页结构的机器学习模型来进行检测。

第六章　机器学习算法选取与检测算法设计

6.1 机器学习算法选取

将 Html 结构特征向量输入到备选机器学习算法中进行训练，生成特征模型。对生成的模型通过测试数据进行评估，调整模型。由于数据量过大，内存不够，并且还要兼顾后面的学习功能。这里的机器学习算法我们引入前面介绍过的增量学习算法(Incremental Learning)。在 sklearn 库中，支持增量学习的分类算法有先验为多项式分布的朴素贝叶斯(MultinomialNB), 先验为伯努利分布的朴素贝叶斯(BernoulliNB), 感知机(Perceptron), 随机梯度下降分类器(SGDClassifier), 被动攻击分类器(PassiveAggressiveClassifier)。这几个类分别适用的场景见表 6-1:

表 6-1 部分机器学习算法及其使用场景

机器学习算法	适用场景
MultinomialNB	稀疏的多元离散值
BernoulliNB	二元离散值或稀疏的多元离散值
Perceptron	速度快，适用于特征维度很大情况
SGDClassifier	适用于大规模问题
PassiveAggressiveClassifier	适用于大规模问题

由于本算法数据规模比较大，按道理应该使用随机梯度下降或者被动攻击分类器。然而由于使用增量学习，所以部分数据规模是较小的，所以不宜采取适用大规模数据的方法以防止过拟合。又因为我们提取出的特征大部分为很稀疏的多元离散值，所以不能选择感知机。所以我们需要在 MultinomialNB 和 BernoulliNB 进行选择。经过试验，MultinomialNB 在代表数据集上的效果要好于 BernoulliNB。于是我们选择了 MultinomialNB 作为我们的机器学习算法。

6.2 检测算法设计

在设计算法之前，我们要明确此算法需要实现的功能。这个检测算法功能主要只有两个，即判断和学习。判断是指判断一个网页中是否包含 spam。根据之前的网页分块，判断网页中含有 spam 的标准是头部含有 spam 或者任意一个 body 块中含有 spam。至于具体如何判断块中是否是 spam，本文会在下面做出阐释。学习是指网页结构模型和此算法对于新的网页数据具有学习功能，比如如果存在一个完全没出现的词该如何将此词纳入到算法和模型中，具体本文会在下面的内容中讲解。

首先，我们先分析如何实现对块中内容的判断。根据观察发现，在优势率字典中，优势率有两个阈值，大于某个阈值一定是 spam，这个阈值我们叫做绝对阈值，小于某个阈值一定不是 spam，这个阈值我们叫做可能性阈值。大于绝对阈值，则一定是 spam，小于可能性阈值，就一定不是 spam。那么在两个阈值中间的该如何判断呢？在两个阈值中间的我

们就要结合网页结构模型来判断，如果网页结构模型预测是 spam，那么即认为这个是 spam，如果网页结构模型预测不是 spam，即认为不是 spam。这样我们就把文本和网页结构特征的判断结合起来了。

此外，我们再分析如何实现模型和算法对新数据的学习功能。其实学习的行为有两种，一种是纠错性学习，即产生了与之前模型不一致的行为需要学习，一种是不存在性学习，即对于之前不存在的数据进行学习。

对于纠错性学习，这里我们主要针对网页结构模型，通过学习新数据来对于网页结构模型进行纠错。那么在什么情况下需要纠错呢？其实很简单，这里的不一致是指模型预测的情况与最后判断结果不一致，在上面的检测算法中，只有两种情况会产生不一致，即大于绝对阈值的情况和小于可能性阈值的情况。如果大于绝对性阈值，但模型却预测这不是 spam 的结构，此时我们需要纠正模型。如果小于可能性阈值，但模型却预测这是 spam 的结构，此时我们也需要纠正模型。那么该如何纠正呢？前面我们提到网页结构模型的学习是使用了增量学习的算法。通过使用 sklearn 的函数给旧模型输入新的数据去训练就可以得到更新后的模型。用此更新后的模型再进行下一次判断和学习。

对于不存在性学习，这里我们主要针对优势率字典，即自然语言模型，新数据中可能有存在的词汇和不存在的词汇，对于不存在单词根据之前判定记为 spam 或者非 spam 存入优势率字典。对于存在的单词，我们需要更新该词在优势率字典中和 spam 文档一起出现的次数与和非 spam 文档一起出现的次数还有其优势率。设 A 是该词和 spam 文档一起出现的次数，B 是该词和非 spam 文档一起出现的次数，优势率为 OR，则对于存在单词有：如果该单词所在块被判定为 spam 则有:

$$B_{new} = B_{old} \text{ 且} A_{new} = A_{old} + 1 \text{且} OR_{new} = (1 + \frac{1}{A_{old}}) \times OR_{old} \tag{6-1}$$

如果该单词所在块被判定为非 spam 则有:

$$A_{new} = A_{old} \text{ 且} B_{new} = B_{old} + 1 \text{且} OR_{new} = (\frac{1}{1+\frac{1}{B_{old}}}) \times OR_{old} \tag{6-2}$$

式 6-1 和式 6-2 是根据前面提到的优势率公式 3-2 所推出的结果。根据式 6-1 和 6-2 可以更新优势率字典。

通过以上两种学习方式，网页结构模型和自然语言模型都在不断学习中更新，对于新类型的数据完全不需要再次专门训练。这种边检测边学习的方法可以适应不断变化的 spam 类型，大大提高了准确度并减小了误报率。

第七章　效果测试与分析

本测试所用测试数据集是教育网内部的总共 600 条左右的 spam 页面和非 spam 页面。测试数据集的大小是原来训练数据集的数量的三倍左右。通常情况下，机器学习结果的测试数据集的大小应该是训练数据的三分之一，但考虑到我们这是一个在学习中检测的算法，需要考虑到学习的因素，所以我们扩大了测试数据集的大小，减小了训练数据集的大小，以测试学习算法的效果。测试对比的集合分为测试集和结果集，比对测试集和结果集中的网页链接从而得到效果评估。测试集是通过人工分类得到的集合。结果集是通过检测算法分类后得到的集合。

7.1 测试标准

测试的评估标准有四个，一个是准确率，一个是错误率，一个是误报率，一个是漏报率。在本测试中，准确的结果是指分类正确的结果，即经过分类的结果和测试集分类中的结果一样。那么准确率就是指分类正确的结果数量占总结果数量的比率。错误的结果是指分类错误的结果，即经过分类的结果和测试集分类中的结果不一致。那么错误率就是分类错误的结果占总结果数量的比率。很容易看出准确率等于 1-错误率。误报的结果是指本身不是 spam 却被分到了 spam 类，那么误报率就是结果为 spam 但本身是非 spam 网页数量与非 spam 网页数量的比率。漏报的结果是指本身是 spam 却被分到了非 spam 类，那么漏报率就是结果为非 spam 但本身是 spam 网页数量与 spam 网页数量的比率。

7.2 测试方法

根据最后输出结果以及原本标记，统计四种情况的数量。设本身是 spam 但被错分成非 spam 的数量为 A，本身是非 spam 但被错分成 spam 的数量为 B，本身是 spam 也被正确分成 spam 的数量为 C，本身是非 spam 也被分成非 spam 的数量为 D，设正确率为 R，错误率为 W，误报率为 Wb，漏报率为 Lb，则

$$\mathrm{R} = \frac{C+D}{A+B+C+D} \tag{7-1}$$

$$\mathrm{W} = \frac{A+B}{A+B+C+D} \tag{7-2}$$

$$\mathrm{Wb} = \frac{B}{B+D} \tag{7-3}$$

$$\mathrm{Lb} = \frac{A}{A+C} \tag{7-4}$$

根据以上公式对数据的统计和计算，我们可以评估出该系统(或算法)的 webspam 检测率，对于其中某些错误的个例可以进行针对性分析研究，以发现系统(或算法)的问题。

7.3 测试结果与比较

根据统计结果，对于这个数据集的统计数据为: A 为 0, 即没有被漏报。B 为 22，即有

22 个被误报。C 为 281，即被分对的 spam 有 281 个。D 为 295，即被分对的非 spam 有 295 个。由此根据以上的计算方法，可以得到正确率为(281+295)/(281+295+0+22)=0.9632，错误率为 0.0368，漏报率为 0，误报率为 22/(22+295)=0.0694。具体情况见图 7-1。

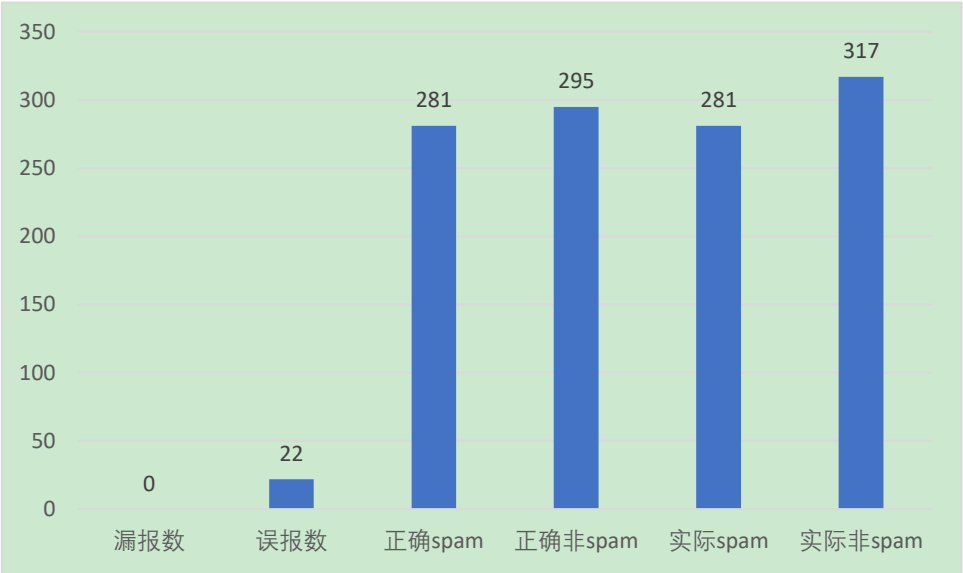


图 7-1 本算法测试结果

根据如上算法结果与目前实际运行的关键词匹配算法在相同数据集上的结果比较得出(见图 7-2)，新算法由于采取了网页结构特征与网页文本内容相结合进行检测的方法，能够在没有漏报的情况下大大减小了误报。

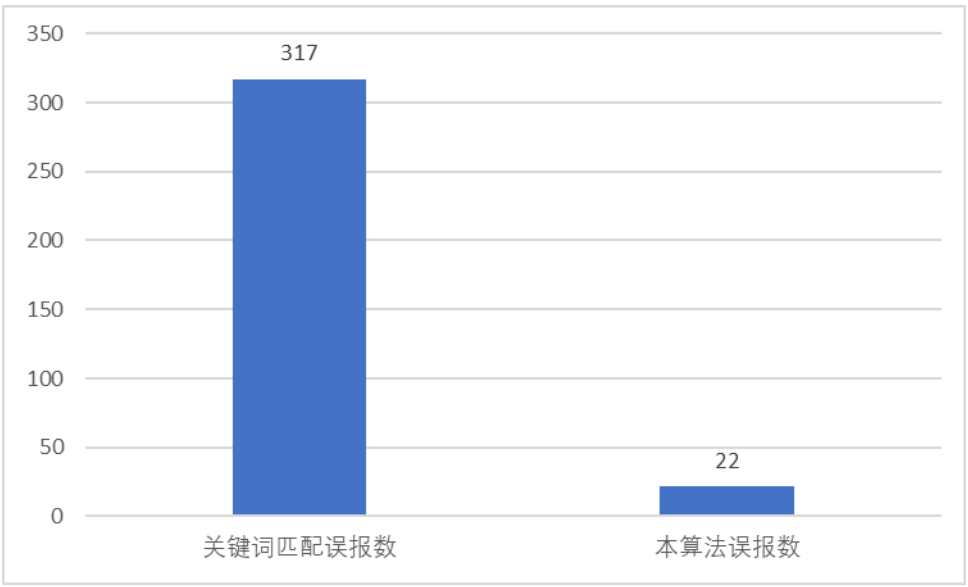


图 7-2 比较结果

7.4 讨论分析

从以上结果我们可以得出，该算法基本可以解决这个分类问题，但仍然存在例外情况。分析了例外的情况后我们分析了误报的原因。由于这个系统主要依赖数据去学习，那么初始数据就显得非常重要。其实误报的主要原因都还是初始数据不完善导致的。初始的优势率字典的建立决定了最终结果的好坏。而决定初始优势率字典的，就是初始的数据。

我们可以挑选一些样本来分析其正确和错误的原因。先挑选正确的样本，如图 7-2 所示是某个被正确检测出来的 spam 网页样本片段。按照之前分块的原则，这里会被分为很多块来分析。每一块都会被系统检测为 spam 网页。分词优势率字典也会随着数据量的增加而更新。根据网页结构特征和自然语言处理中优势率筛选的方法这个例子中得到了充分的发挥。

如图 7-3 所示是误报的样本片段之一。在文本中，其中有一个关键词的优势率特别大，但这恰恰不完全是一个 spam 类的关键词，只是在初始数据中在 spam 块中出现的次数比较多（如图 7-4），而其他的很多关键词的优势率不能将这种优势率抵消，所以导致产生误报。

```
<div class="c1_c_u">
  <ul>

    <li><a href="../News/show.asp?ViewID=4257">南航后勤集团接待服务中心调味品类原料采购中…<strong>News</strong></a>
    <em>2018-1-27</em></li>

    <li><a href="../News/show.asp?ViewID=4256">南航后勤集团接待服务中心水产类原料采购中标…<strong>News</strong></a>
    <em>2018-1-27</em></li>

    <li><a href="../News/show.asp?ViewID=4255">南航后勤集团接待服务中心蔬菜类原料采购中标…<strong>News</strong></a>
    <em>2018-1-27</em></li>

    <li><a href="../News/show.asp?ViewID=4254">南航后勤集团接待服务中心家禽类原料采购中标…<strong>News</strong></a>
    <em>2018-1-27</em></li>

    <li><a href="../News/show.asp?ViewID=4253">南航后勤集团车辆运输中心润滑油采购项目中标…<strong>News</strong></a>
    <em>2018-1-27</em></li>

    <li><a href="../News/show.asp?ViewID=4252">南航后勤集团车辆运输中心轮胎采购项目中标公…<strong>News</strong></a>
    <em>2018-1-27</em></li>

    <li><a href="../News/show.asp?ViewID=4251">南航后勤集团车辆运输中心维修配件采购项目中…<strong>News</strong></a>
    <em>2018-1-27</em></li>

    <li><a href="../News/show.asp?ViewID=4250">南航后勤集团车辆运输中心供油服务项目中标公…<strong>News</strong></a>
    <em>2018-1-27</em></li>
```

图 7-3 误报样本片段

```
配件 37 0 705.498514116
```

图 7-4 初始数据中某关键词在优势率字典中的值

通过以上对于误报的分析。可以看出，它们出现的主要原因有两个。一是此系统虽然在后期检测对数据有一定程度的学习，但是还是对初始数据的情况有极大的依赖性。如果前期的初始数据是不完善的会导致本身不是 spam 的词成为 spam 特征词。二是优势率标准的选择。在本课题中，我们选择了平均优势率作为标准。然而，我们从误报中可以看出，平均主义虽然能很大程度提高准确率，但也会造成一些由于初始数据不完善导致的误报。因此，我们建议的解决误报的方法是是在通过数据筛选出优势率字典后，需要人工对于初始优势率字典再次进行一定程度的筛选完善，越是完善的初始数据，准确率会越高。

第八章　总结与展望

本文独创性的提出了一种结合网页文本的自然语言处理模型和网页结构的机器学习模型对webspam判断的方法，在一定程度上提高了webspam检测的准确率，降低了webspam检测的误报率。下面我对毕设过程及毕设作品本身做一些总结和展望。

8.1 总结

经过几个月的努力，本次毕业设计基本完成。在本次毕业设计的过程中，指导老师杨望老师尽职尽责。在杨老师的悉心指导下，经过思考和讨论，我们确定了研究过程和方法。在这些思想方法的指导下，我通过对相关知识栈的学习和扩展，分析到了问题的根本，找到了该问题的解决方法，并在目标数据集以高准确率通过了测试。通过这次毕业设计，不仅让我学到了更多的知识，对现在最火的机器学习行业有了更深的了解，而且提升了我在处理机器学习或者处理数据相关的问题方面的经验，这些经验一定会在以后的学习研究中给我带来很大的帮助。

webspam 其实是已经存在很多年的搜索引擎欺骗问题。有句话说的好，因为有商业利益的关系，只要有搜索排名，尽可能提升网页排名的方法的就一定会存在并将长期存在。因为有利益的存在，即使一部分 webspam 方法被通过修改搜索引擎排名算法等方法解决了，新的方法也会不断的被研究出来。这是一个正义与邪恶之间的斗争。这种斗争将永无中止。然而，万变不离其宗，搜索引擎的排名可以有不同的标准，然而往往都离不开网页文本信息的提取和网页链接的指向。 虽然我们永远不能通过修改排名算法来完全制止 webspam 的发生，但是抓住了这些根本，我们很大程度的就可以把他们检测出来再进行处理。在本次设计中，我们主要的工作就是将 webspam 检测出来。

虽然最终采取了前述的分块的算法并最终实现， 但实际上在这种算法被想出来之前，也就是在毕设前期，我通过对相关技术栈的学习也想过一些其他的方法，并进行过一些前期探索。比如将整个网页的中文提取出来用机器学习算法搭建模型，但后来经过实验，效果非常差，原因是 webspam 仅仅出现在网页的一小部分，如果不分块，那么相当于把不是 spam 的字符也分到了 spam 中，所以结果很不明显。于是在老师的指导下，经过思考及对网页结构的分析，找到出了前述的分块方法，后期证明分块是一个非常重要的决定，只有分块提取信息才能使得这个问题得到最终解决。

还有一个难点是在检测中学习。经过对相关技术了解学习及自我思考创新，在老师的指导下我解决了如何将学习和检测过程结合起来。一开始，本来要使用 TF-IDF 作为网页文本处理的方法，但是后来根据 webspam 文本的低词频特性发现这种基于文档内词频的算法不能选取，于是采用了优势率的方法。本来要对分词也同样采取 one-hot 编码方式，但是由于词汇量太多导致的矩阵维度过大问题，后来经过研究只使用优势率的判断就可以基本解决自然语言问题，既无需要使用高维度矩阵，又可以相对准确的将垃圾词汇筛选出来。在解决学习数据规模比较大的过程中，发现了增量学习的方法。通过使用增量学习的方法，既解决了学习数据规模大的问题，又可以通过这种方法实现在检测中学习。

8.2 展望

虽然本课题最终的结果准确率很高，但是仍然有很多待研究和解决的问题。通过分析

漏报和误报的测试数据，我们可以得出漏报和误报的原因，从而发现这个系统需要解决的问题。本课题主要有以下几个问题需要后续继续研究。

一是初始优势率词典的构建。本文算法的成功非常依赖于初始优势率词典的构建。后续的研究可以对初始优势率词典的构建进行优化。比如选取特征更加明显的数据初始化优势率字典中的值，或者对于优势率字典进一步进行人工筛选，这样才能得到更好的结果。

二是优势率指标的确定。本文是选用所有分词的平均优势率，这种方式有一定缺陷，就是特别依赖于初始化优势率字典的特征筛选过程。希望后续能够研究出一种方式解决优势率指标的问题，即使初始化优势率字典稍微有出入，也可以在后期检测学习中在更大程度上解决误报的问题。

三是块与块之间的关系。在本文中，对于每一个网页来说，我们是以块作为一个单独个体进行研究的，并没有考虑块与块之间的联系，比如块与块之间的文本相似度。后续的研究可以将一个网页内块与块的相似度，比如文本相似度考虑进去，从而得到更加准确的结果。

四是对于数据的存储，本课题是使用文件系统存储各类数据，虽然有容易存取读入的优势，但却有不方便索引和搜索的缺陷。之后的研究可以使用一些文本数据库例如 Elasticsearch 存储相关必要数据，使本系统更加便于搜索和索引。

综上所述，本课题虽然完成了课题要求，但还是有一些可以完善的地方。后续的研究可以从上述这几个方面入手对系统进行修正和完善。

致　谢

四年的时光如白驹过隙一瞬而过，至此已接近尾声。这四年来的时光中充满了我的蜕变和成长。从刚入学时的编程小白到如今的专业程序员，想想四年来的辛苦没有白白付出。在这里我要首先感谢我的指导老师杨望老师，没有他的一丝不苟的悉心指导就没有这个项目的完成。其次要感谢我的父母在这二十多年里的辛勤付出，给我稳定的生活来源让我可以专心完成这个项目。还要感谢四年以来所有给我上过课的老师们，帮助过我的同学，给大家说声谢谢了，是你们在我遇到问题时给我答疑解惑，帮助我走出难题。最后，感谢所有帮助关心过我的人，祝你们在人生的道路上一帆风顺，所向披靡！

参考文献

[1] CNNIC.第 41 次中国互联网络发展状况统计报告[EB/OL]. [2018-01-31] http://cnnic.cn/hlwfzyj/hlwxzbg/hlwtjbg/201801/P020180131509544165973.pdf

[2] Largillier T, Peyronnet S. Webspam demotion: Low complexity node aggregation methods[J]. Neurocomputing, 2012, 76(1):105-113.

[3] Yang Y J, Yang S H, Hu B G. Fighting WebSpam: Detecting Spam on the Graph Via Content and Link Features[M]// Advances in Knowledge Discovery and Data Mining. Springer Berlin Heidelberg, 2008:1049-1055.

[4] 杨向军. Web spam 检测系统的设计和实现[D]. 华南理工大学, 2010.

[5] 于兵兵. Web Spam 检测及网页排序算法的研究[D]. 西安电子科技大学, 2012.

[6] 计华. Web Spam 特征分析及其检测技术研究[D]. 山东师范大学, 2015.

[7] 胡瑜, 王立志. 基于 HTML 结构特征的网页信息提取[J]. 辽宁石油化工大学学报, 2009, 29(3):65-69.

[8] 李铭岳, 周军. 基于改进 HTML-Tree 的中文网页特征向量提取方法[J]. 信息技术, 2009, 33(1):10-14.

[9] 宋斌, 方小璐. 基于网页特征的 TFIDF 改进算法[J]. 网络新媒体技术, 2002, 23(1):18-20.

[10] 刘慧, 马军, 雷景生,等. 基于词频的权值计算在邮件过滤算法中的应用[J]. 计算机工程, 2006, 32(17):60-62.

[11] 李晓明, 闫宏飞, 王继民. 搜索引擎 ：原理、技术与系统[M]. 科学出版社, 2012.

[12] 杜一平, 刘燕君. 基于优势率的改进二元特征提取方法[J]. 计算机系统应用, 2010, 19(2):106-109.