

基于 Bagging-SVM 集成分类器的网页作弊检测

唐寿洪 朱 焱 杨 凡

(西南交通大学信息科学与技术学院 成都 610031)

摘 要 网页作弊不仅造成信息检索质量下降,而且给互联网的安全也带来了极大的挑战。提出了一种基于 Bagging-SVM 集成分类器的网页作弊检测方法。在预处理阶段,首先采用 K-means 方法解决数据集的不平衡问题,然后采用 CFS 特征选择方法筛选出最优特征子集,最后对特征子集进行信息熵离散化处理。在分类器训练阶段,通过 Bagging 方法构建多个训练集并分别对每个训练集进行 SVM 学习来产生弱分类器。在检测阶段,通过多个弱分类器投票决定测试样本所属类别。在数据集 WEBSPPAM-UK2006 上的实验结果表明,在使用特征数量较少的情况下,本检测方法可以获得非常好的检测效果。

关键词 网页作弊,集成分类器,特征选择,信息熵,弱分类器

中图法分类号 TP181 文献标识码 A DOI 10.11896/j.issn.1002-137X.2015.1.053

Web Spam Detection Based on Integrated Classifier with Bagging-SVM

TANG Shou-hong ZHU Yan YANG Fan

(School of Information Science and Technology, Southwest Jiaotong University, Chengdu 610031, China)

Abstract Web spam not only declines the quality of information retrieval, but also causes troubles to the security of Internet. This paper proposed a Bagging-based integration of SVM to detect Web spam. In preprocessing stage, a technique referring to K-means is introduced to solve the class-imbalance problem of dataset firstly, and then an optimal feature subset is culled by using CFS. Finally the optimal feature subset is discretized by the information entropy. In the stage of classifier training, several training datasets are obtained by Bagging and each training dataset is utilized to produce weak classifier respectively after SVM learning. In detection stage, test samples are voted by weak classifiers obtained before determining their categories. Experimental results on the WEBSPPAM-UK2006 reveal that the proposed method can achieve better results with less number of features.

Keywords Web spam, Integrated classifier, Feature selection, Information entropy, Weak classifier

1 引言

互联网已成为现代社会的主流媒体,并逐渐改变着人们的生活方式。人们不仅可以使互联网查找有用信息,而且还可以使用互连网实现网上购物、在线支付、网上银行、炒股等活动。据 CNNIC 数据显示,截止到 2013 年 12 月底,我国网民规模达到 6.18 亿,互联网普及率为 45.8%,网络购物用户规模达到 3.02 亿,网络购物使用率提升至 48.9%^[1]。从 CNNIC 数据显示结果可以看出,人们的生活越来越离不开互联网。搜索引擎无疑将成为人们日常生活中不可或缺的工具,但其也面临着越来越多的挑战和危险。网页作弊就是其中的一个挑战,因为它不仅会造成搜索引擎检索信息质量的下降,而且给互联网用户造成了严重的经济损失。

网页作弊是指采用不正当手段使得网页在搜索引擎检索结果中的排名高于其实际应得排名的行为^[2]。一般而言,网页作弊分为以下 3 种类型:基于链接的网页作弊、基于内容的

网页作弊和基于隐藏的网页作弊。基于链接的网页作弊是一种专门针对网页之间链接关系的作弊技术,作弊者根据 Page-Rank 算法和 HITS 算法的特点,有目的地构建网页之间的链接关系,从而提高自身在搜索引擎中的重要性评分。基于内容的网页作弊是一种针对网页文本内容的作弊技术,作弊者通常会根据 TFIDF 模型的特点,人为地在网页中添加大量的热门关键词,从而使该网页能够在用户进行热门关键词查询时被检索到,并使该网页获得较高的相关性分数,从而达到提升作弊网页排名的目的。基于隐藏的网页作弊是通过欺骗用户端浏览器去浏览作弊网页的一种手段。作弊者通常会采用颜色模式进行作弊,即将 HTML 文档的 body 标签中的关键词汇设置为网页背景色以达到关键词对用户不可见的目的。有的作弊者甚至还会采用一种重定向技术来进行网页作弊,即当用户端浏览器加载一个 URL 之后,网页将会跳转到另外一个 URL 所指向的页面。由于搜索引擎很难将重定向之后的网页爬取下来,因此作弊者可以将 URL 对应的原始

到稿日期:2014-01-19 返修日期:2014-05-09 本文受四川省学术和技术带头人后备人选培养基金(X800912371309)资助。

唐寿洪(1986—),男,硕士生,主要研究方向为万维网异常挖掘、机器学习、人工智能,E-mail: magicdreams@163.com;朱焱(1965—),女,教授,硕士生导师,CCF 会员,主要研究方向为万维网异常挖掘、机器学习、人工智能、网络信息安全、大数据管理与智能分析;杨凡(1989—),男,硕士生,主要研究方向为万维网异常挖掘、机器学习。

网页做成一个内容十分丰富的页面,然后将其送至搜索引擎检索,在用户点击该网页之后,通过 meta 刷新或 JavaScript 脚本跳转到作弊网页,从而达到网页作弊的目的。

一些恶意网站通过作弊手段使其在搜索引擎结果中获得高排名以欺骗互联网用户点击,一旦用户点击该网站链接,一些恶意软件就会随之在访问者的电脑上自行安装。在用户不知晓的情况下,非法分子将会盗取用户个人信息,如用户密码、财政信息、网上银行证书等^[3]。近些年来,越来越多的互联网用户遭遇虚假钓鱼网站、诈骗交易、交易劫持、网银被盗等针对网络购物的安全攻击。据 360 互联网安全中心统计,2013 年我国电脑遭受病毒攻击约为 1.21 亿次^[4],全年新增各类钓鱼网站约为 220.1 万个,平均每天截获新增钓鱼网站约为 6030 个^[5]。为了维持互联网的安全秩序,提高搜索引擎检索信息的质量和效率,针对当前网页作弊技术,我们需要采取相应的措施将作弊网页从大量的网页资源中检测出来并加以惩罚,这项工作对于规范互联网安全秩序十分重要。

2 相关工作

自 Henzinger 等^[6]在 2002 年指出网页作弊是当代搜索引擎所面对的主要挑战之后,国内外研究学者开始提出了大量检测网页作弊的算法。Gyöngyi 等^[7]提出采用 TrustRank 算法来检测网页作弊,即首先选出一些信任值比较高的网页作为种子网页,然后通过种子网页的出链将信任值扩散传播给其他网页,最后对所有网页的信任值进行汇总。如果网页的信任值很高,那么该网页将被认为是值得信任的网页。Wu 等^[8]提出了一种检测链接农场作弊的方法,该方法通过在网页数据集中产生作弊种子集并采用“ParentPenalty”算法来扩展种子集,然后对作弊网页的邻接矩阵做降权处理,最后通过常用的排名算法 HITS 或 PageRank 对数据集中的网页进行重新排名,其能够非常有效地抵御链接农场作弊和“KTC effect”的攻击。不同网页特征对搜索引擎排名的重要程度不同,Egele 等^[3]根据这一原理筛选出重要的特征,然后根据这些重要特征信息将作弊网站识别出来并将其从搜索引擎结果列表中移除,从而使得指向有用网页的链接能够在搜索引擎结果列表中排名更靠前,最终更加方便互联网用户查找有效信息。Suhara 等^[9]首先利用 ham 语料库创建 LDA 模型,然后利用 LDA 模型提取网页的主题分布特征,最后将网页的主题分布特征与网页的内容特征相结合来建立分类器,实验结果表明该检测方法具有非常好的检测效果。

链接劫持是指将正常网站的排名分数劫持以后提供给作弊网站以提高作弊网站在搜索引擎中的排名。Chung 等^[10]通过对信任指数比较高的被劫持网站以及该网站的邻居网站进行链接结构分析,提出了一种改进的 PageRank 算法来检测链接劫持。为了更加有效地找出被劫持的网站,他们评估了网站的可信度,并检测值得信赖的网站是如何被邻居网站中不值得信赖的网站劫持的。此外,他们还定义了两种被劫持分数,用以计算一个网站被作弊者劫持的概率,实验结果表明,这种改进的 PageRank 算法能够非常有效地检测出链接劫持。Araujo 等^[11]通过对链接质量和链接网页之间的差异度进行分析,提出了一种基于链接质量分析和语言模型的网页作弊检测方法;通过将网页的链接特征和网页文本的内容特征结合起来分析,并采用语言模型 KL 差异度来描述两个

相互链接的网页之间的关系,这种方法能够有效地检测出内容与链接相结合的作弊方式。

沈国阳等^[12]通过对网页之间的链接关系进行分析,定义了两种链接的时间特征,即入链增长率和入链死亡率,并采用这两种指标来度量网页入链在一段时间内的变化,如果在短期内网页入链增长比较快或在短期内网页入链死亡率比较高,那么该网页是作弊网页的概率就非常大。余慧佳等^[13]根据用户访问日志建立用户浏览图并对 Web 图进行链接分析,在建立的 Web 图比原始的 Web 图更小的情况下,获得的检测效果比原始 Web 图下获得的检测效果更好。刘奕群等^[14]提出采用群体智能的方法来检测作弊网页,通过对大规模用户访问日志进行收集和分析,他们发现用户访问作弊网页的模式与访问普通网页的模式是不同的,并以此提取出作弊网页的特征,然后结合群体智能的方法将作弊网页从海量的网页资源中检测出来。

3 Bagging-SVM 相关原理

3.1 Bagging-SVM 基本思想

给定一个集合 S ,若对集合 S 进行 T 轮 bootstrap 抽样,那么将得到 T 个子集 $S_k, k=1,2,\dots,T$,且每个集合 S_k 均含有 N 个样本。若给定一个基分类器,如果对每个子集 S_k 都进行基分类器学习,那么每个子集 S_k 将产生一个弱分类器 $\varphi(x, S_k)$,然后将 T 个弱分类器集成为一个强分类器 $\Phi(x, S)$ 。当输入测试样本时,强分类器 $\Phi(x, S)$ 将输出 k 个弱分类器 $\varphi(x, S_k)$ 投票的结果,即投票中占多数的类别将作为测试样本的类别。Bagging 的实现过程如图 1 所示。

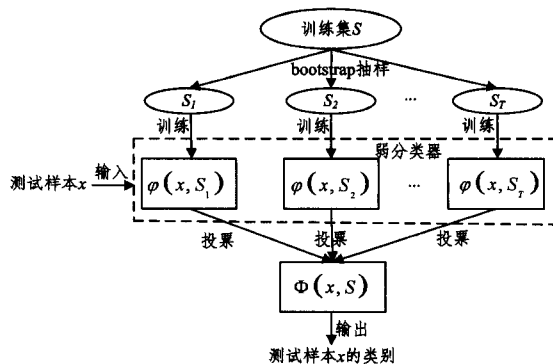


图 1 Bagging 实现过程

3.2 支持向量机

支持向量机(support vector machines, SVM)是基于结构风险最小化原理的统计学习方法。其基本思想是将输入向量通过非线性变化映射到高维空间中,在高维空间寻找最优分类平面,然后使用最优分类平面解决二分类问题,如图 2 所示, H 为最优分类平面示意图。

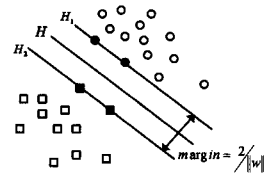


图 2 最优分类平面示意图

对于一个给定的二分类问题,假设训练集合为 $D=\{(x_i, y_i) \mid x_i \in R^r, i=1,2,\dots,n\}$,其中 y_i 是 x_i 的类标并且 $y_i \in$

$\{1, -1\}$, 1 表示正类, -1 表示负类。寻找一个最优分类平面 $w^T x_i + b$ 可以转换为求解以下最优化问题:

$$\begin{cases} \min \|w\|^2 \\ \text{满足: } y_i \cdot (w^T x_i + b) \geq 1, (i=1, 2, \dots, n) \end{cases} \quad (1)$$

当样本不可分时,需要放松边距的约束,引入松弛变量 $\xi_i > 0$ 和惩罚项 C ,那么将得到一个新的最优化问题:

$$\begin{cases} \min \frac{1}{2} \|w\|^2 + C \sum_{i=1}^n \xi_i \\ \text{满足: } y_i \cdot (w^T x_i + b) \geq 1 - \xi_i, i=1, 2, \dots, n \end{cases} \quad (2)$$

由最优化理论可知,上述优化问题可以转化为对偶问题来求解:

$$\begin{cases} \max w(\alpha) = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n y_i y_j \alpha_i \alpha_j \langle x_i, x_j \rangle \\ \text{满足: } \sum_{i=1}^n y_i \alpha_i = 0, 0 \leq \alpha_i \leq C, i=1, 2, \dots, n \end{cases} \quad (3)$$

求解最优解 $w^* = \sum_{i=1}^K \alpha_i^* y_i x_i$, 其中 $\alpha_i \neq 0$, 其对应的样本称为支持向量 $(x_1, x_2, \dots, x_m)$, K 为支持向量的个数。最终的判别函数为:

$$f(x) = \text{sign}(\sum_{i=1}^K \alpha_i^* y_i (x_i \cdot x) + b^*) \quad (4)$$

对于非线性问题, SVM 将输入样本通过非线性变换映射到高维空间 Γ 中, 从而使得样本在高维空间 Γ 中线性可分。那么式(3)可以转换为:

$$\begin{cases} \max w(\alpha) = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n y_i y_j \alpha_i \alpha_j K \langle x_i, x_j \rangle \\ \text{满足: } \sum_{i=1}^n y_i \alpha_i = 0, 0 \leq \alpha_i \leq C, i=1, 2, \dots, n \end{cases} \quad (5)$$

其中核函数 $K(x_i, x)$ 由式(6)给出:

$$K(x_i, x) = \exp(-\frac{\|x_i - x\|^2}{2\sigma^2}) \quad (6)$$

本文采用的核函数为径向基函数(RBF), 最终得到的判别函数为:

$$f(x) = \text{sign}(\sum_{i=1}^n \alpha_i^* y_i K(x_i, x) + b^*) \quad (7)$$

4 基于 Bagging-SVM 集成的网页作弊检测框架

本文采用基于 Bagging-SVM 集成分类器的网页作弊检测算法检测网页作弊。本算法的核心是: 给定一个训练集, 对训练集进行 T 轮 bootstrap 抽样形成 T 个训练特征子集, 然后对 T 个训练特征子集分别进行 SVM 学习产生 T 个弱分类器, 最后把形成的多个弱分类器集成为一个强分类器。具体的检测框架如图 3 所示。

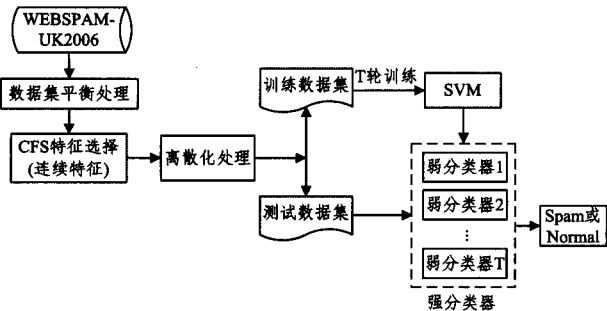


图 3 基于 Bagging-SVM 集成分类器的网页作弊检测框架

在对数据集 WEBSpam-UK2006 进行分析时, 经常会遇到许多问题, 如数据集的不平衡性、高维性、特征之间的组合

优化问题等。因此, 在数据集预处理阶段, 首先采用 K-means 方法解决数据集的不平衡问题。然后, 采用 CFS 特征选择方法选出最优特征集合以达到降低数据维度的目的。最后, 对最优特征集合进行信息熵离散化处理以提高机器学习的准确性和效率。预处理完成以后, 对最优特征集合进行 T 轮 bootstrap 抽样形成 T 个特征子集, 然后分别对每个特征子集进行 SVM 学习产生弱分类器。最后, 将多个弱分类器集成为一个强分类器。在检测阶段, 当输入测试样本后, 强分类器能够很好地预测出输入样本的类别。

5 数据预处理

5.1 数据集不平衡处理

数据集不平衡是指在数据集中样本的类别分布不均衡, 即数据集中不同类别的样本数量差异较大。数据集的不平衡将会导致机器学习的偏向性, 即分类器会倾向于把一个样本预测为多数类样本, 这就会导致少数类样本的识别率较低。在实际的问题中, 人们往往对少数类样本更感兴趣, 因此如何解决数据集的不平衡问题将成为数据挖掘和机器学习领域中的关键。本文采用 K-means 算法解决数据集不平衡的问题。假设在数据集 D 中, 被标注为类别 C_1 的样本数有 k_1 个, 被标注为类别 C_2 的样本个数有 k_2 个 ($k_1 \gg k_2$)。为了解决数据的不平衡性, 需要从被标注类别 C_1 的所有样本中挑选出最具代表性的样本以达到与类别 C_2 样本个数平衡。本文中, 每个样本代表一个 d 维的向量, 采用 K-means 算法解决数据集不平衡问题, 见算法 1。

算法 1 K-means 解决数据不平衡

从 n 个向量 $(v_1, v_2, \dots, v_n)$ 中随机选出 k 个向量 $(w_1, w_2, \dots, w_k)$ 作为初始的聚类中心, 每个聚类 C_i 与聚类中心 w_i 相对应。

重复:

对于每一个输入向量 v_j , 计算其与聚类中心 w_i 的距离, 同时将向量 v_j 分配到离它最近的一个聚类 C_i 中, 其中 $i \in \{1, \dots, k\}, j \in \{1, \dots, n\}$ 。

更新当前每个聚类 C_i 的聚类中心 w_i :

$$w_i = \sum_{v_j \in C_i} v_j / |C_i|$$

计算误差函数:

$$E = \sum_{i=1}^k \sum_{v_j \in C_i} |v_j - w_i|^2$$

直到误差函数改变不明显或者聚类中心不再改变为止。

当 K-means 算法完成后, 我们将获得 k 个聚类。样本离聚类中心的距离越近, 该样本就越具有代表性。假设聚类 C_i ($i=1, 2, \dots, k$) 含有 n_i 个样本, 我们将从聚类 C_i 中选取 $n_i * k_2/k_1$ 个离聚类中心 w_i 最近的样本。通过以上处理, 将得到 N 个被标注为类别 C_1 的样本, 此时类别 C_1 的样本个数将与类别 C_2 的样本个数基本平衡。其中, 操作符 $[\cdot]$ 表示取整。

$$N = \sum_{i=1}^k [n_i * k_2/k_1] \quad (8)$$

5.2 基于 CFS 的特征选择

CFS 特征选择是指对特征集合中的特征子集进行启发式评估。一个好的特征子集应该具备两个条件: 1) 特征子集中的每个特征与类具有很高的相关性; 2) 特征子集中的特征之间的相关性比较低。其评估方法如下:

$$\text{Merit}_i = \frac{k r_{cf}}{\sqrt{k + k(k-1) r_{ff}}} \quad (9)$$

其中, $Merits$ 表示对一个包含 k 个特征的特征子集 S 进行的评估; $\overline{r_{cf}}$ 表示特征与类的平均相关度; $\overline{r_{ff}}$ 表示特征与特征之间的平均相关度。 $\overline{r_{cf}}$ 与 $\overline{r_{ff}}$ 可以通过信息增益的方法求得, 下面先给出计算信息增益的方法。

如果给定一个特征 Y , y 为 Y 可能取到的每一个值, 那么 Y 的信息熵为:

$$H(Y) = - \sum_{y \in Y} p(y) \log_2 p(y) \quad (10)$$

若已知特征 X , 则 Y 的信息熵为:

$$H(Y|X) = \sum_{x \in X} p(x) \sum_{y \in Y} p(y|x) \log_2 p(y|x) \quad (11)$$

那么, 通过式(10)、式(11)可以得到如下信息增益:

$$IG = H(Y) - H(Y|X) \quad (12)$$

信息增益值越大, 说明特征 X 与特征 Y 之间相关性越高。

5.3 基于信息熵的离散化处理

离散化技术有助于提高机器学习的效率和预测准确率。本文采用一种基于信息熵的方法来确定连续特征的潜在断点^[15], 从而达到对数据集进行离散化的目的。给定一个特征集合 S 、一个特征 A 和一个潜在断点 T , 则特征集合 S 的信息熵为:

$$Ent(S) = - \sum_{i=1}^2 P(C_i, S) \log_2 P(C_i, S) \quad (13)$$

其中, $P(C_i, S)$ 表示在集合 S 中类标为 C_i 的样本比例。

假设断点 T 将特征 A 的取值集合划分为两个集合 S_1 (A 的取值小于等于 T) 和 S_2 (A 的取值大于 T), 那么关于特征 A 的加权信息熵 $Ent(A, T; S)$ 为:

$$Ent(A, T; S) = \frac{|S_1|}{|S|} Ent(S_1) + \frac{|S_2|}{|S|} Ent(S_2) \quad (14)$$

对特征 A 进行离散化处理时, 从所有的候选断点中选出一个断点 T_A 使得 $Ent(A, T_A; S)$ 取值最小, 则断点 T_A 将被视为特征 A 的一个断点。整个离散化过程将在子集 S_1 和 S_2 中循环进行下去, 直到信息增益 $IG(A, T; S)$ 小于阈值 δ 为止。

$$IG(A, T; S) = Ent(S) - Ent(A, T; S) \quad (15)$$

$$\delta = \frac{\log_2(N-1) + \delta(A, T; S)}{N} \quad (16)$$

$$\delta(A, T; S) = \log_2(3^k - 2) - k Ent(S) + k_1 Ent(S_1) + k_2 Ent(S_2) \quad (17)$$

其中, N 表示在集合 S 中的样本个数; k, k_1, k_2 分别表示在集合 S, S_1, S_2 中的类别个数。

6 实验结果及其分析

6.1 数据集及其准备

本实验所采用的数据集是由雅虎实验室发布的 WEBS-PAM-UK2006^[16]。此数据集是由 Carlos Castillo 等研究学者于 2006 年 5 月开始收集的域名为 .uk 的网页, 并且由 33 名志愿者参与标注, 标注类别为“normal”、“spam”和“borderline”。其中, WEBS-PAM-UK2006 包含 7.79 千万个网页, 有超过 30 亿个链接和 11400 个域名。“borderline”类样本是类别不确定的样本, 在实验过程中不会使用到, 而且还会降低机器学习的性能和检测的准确性, 因此在实验过程中我们过滤了被标注为“borderline”的样本。此外, 我们将网页的两个直接特征、内容特征和链接特征结合在一起。通过上述的准备

工作, 数据集一共包含 8489 个网页, 其中, 6511 个网页被标注为“normal”, 1978 个网页被标注为“spam”, 数据集的分布如图 4 所示。

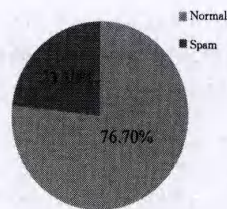


图4 数据集分布

上述分布图中, 被标注为“normal”的网页与被标注为“spam”的网页数量之比大约为 3.3:1, 可以看出数据集是不平衡性的。本文采用 K-means 算法将标注为“normal”的多数类样本聚类为 3 个聚类, 样本聚类信息如图 5 所示。根据聚类 1、聚类 2 和聚类 3 中样本的分布, 从聚类 1 中选出 1108 个最具代表性的样本, 从聚类 2 中选出 336 个最具代表性的样本, 从聚类 3 中选出 534 个最具代表性的样本。最终得到 1978 个被标注为“normal”的样本, 这与被标注为“spam”的样本平衡。

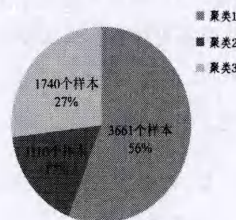


图5 聚类结果

6.2 分类器评价指标

为了更好地评价实验结果, 本实验采用误检率(FPR)、精准率(Precision)、召回率(Recall)和 F 值作为算法性能的评价标准。为定义上述各项指标, 需要用到机器学习方法的基础评价——混淆矩阵, 如表 1 所列。

表1 两类混淆矩阵

实际	预测	
	正类	反类
正类	TP	FN
反类	FP	TN

根据表 1 所描述的混淆矩阵, 可以定义如下常用的分类器评价指标:

$$\text{误报率: } FPR = \frac{FP}{FP + TN}$$

$$\text{精准率: } Precision = \frac{TP}{TP + FP}$$

$$\text{召回率: } Recall = \frac{TP}{TP + FN}$$

$$\text{F 值: } F = \frac{2 * Precision * Recall}{Precision + Recall}$$

6.3 实验结果

为了更好地说明本文算法性能的优势, 选取了朴素贝叶斯(NaiveBayes)、交替决策树(ADtree)、决策树 J48、随机森林(RandomForest)作为对比实验, 并采用十倍交叉法进行测试, 实验结果见表 2—表 4。表 2 给出了在数据集不平衡的情

况下各种检测算法性能的对比。从表 2 可以看出,数据集不平衡时,SVM 比其他算法具有更好的检测效果。在对数据集进行 K-means 平衡处理后,各算法性能均有了明显的提高,实验结果见表 3。其中,表 2、表 3 中实验结果所使用的数据集中均含有 139 个特征。在对平衡数据集进行 CFS 特征选择后,实验数据集中只包含 11 个最优的特征,实验结果见表 4。表 5 给出了在对平衡数据集进行 CFS 特征选择后,对各分类器进行 Bagging 集成后获得的实验结果。从表 5 可以看出,由 Bagging 与 SVM 组合所获得的分类器各项性能指标均达到最高,这充分说明了 Bagging-SVM 集成分类器对于网页作弊的检测效果是非常理想的。

表 2 不平衡数据集

分类算法	特征数	FPR	Precision	Recall	F 值
NaiveBayes	139	0.111	0.695	0.833	0.758
ADtree	139	0.066	0.765	0.704	0.733
J48	139	0.069	0.758	0.708	0.732
RandomForest	139	0.068	0.767	0.742	0.754
SVM	139	0.058	0.795	0.747	0.771

表 3 平衡数据集

分类算法	特征数	FPR	Precision	Recall	F 值
NaiveBayes	139	0.040	0.958	0.914	0.935
ADtree	139	0.054	0.945	0.928	0.936
J48	139	0.057	0.943	0.942	0.942
RandomForest	139	0.039	0.960	0.924	0.942
SVM	139	0.044	0.956	0.958	0.957

表 4 平衡数据集+CFS 特征选择

分类算法	特征数	FPR	Precision	Recall	F 值
NaiveBayes	11	0.039	0.960	0.926	0.943
ADtree	11	0.037	0.961	0.921	0.941
J48	11	0.033	0.966	0.920	0.942
RandomForest	11	0.046	0.953	0.926	0.939
SVM	11	0.042	0.957	0.938	0.947

表 5 平衡数据集+CFS 特征选择+Bagging

分类算法	特征数	FPR	Precision	Recall	F 值
NaiveBayes	11	0.038	0.960	0.926	0.943
ADtree	11	0.050	0.950	0.934	0.942
J48	11	0.038	0.961	0.925	0.943
RandomForest	11	0.042	0.957	0.931	0.944
SVM	11	0.038	0.961	0.937	0.949

结束语 本文在分析当前网页作弊特点的基础上,提出了基于 Bagging-SVM 集成的网页作弊检测算法。首先,我们对数据集不平衡性问题进行了分析,并通过 K-means 算法成功地解决了数据集的不平衡性问题。然后通过计算特征与特征之间的相关性以及特征与类的相关性,采用 CFS 的特征选择方法从数据集中选出最优实验特征子集。最后采用信息熵的方法对最优实验特征子集进行离散化处理,以进一步提升机器学习的效率。在训练分类器的过程中,通过对最优实验特征集进行 bootstrap 抽样形成多个训练特征子集,然后采用基分类器 SVM 分别对每个训练特征子集进行学习以形成弱分类器,最后将形成的多个弱分类器进行集成以形成强分类器。实验结果表明,在使用较少特征情况下,基于 Bagging-SVM 集成的分类器能够获得非常好的检测效果。

参 考 文 献

[1] 中国互联网信息中心.《第 33 次中国互联网络发展状况统计报告》[R]. 2014. http://www.cnnic.net.cn/hlwfzyj/hlwxxzbj/hlwtjbg/201401/t20140116_43820.htm

[2] Gyöngyi Z, Garcia-Molina H. Web spam taxonomy [C]//Proceedings of the 1st International Workshop on Adversarial Information Retrieval (AIRWeb 2005). 2005;39-47

[3] Egele M, Kolbitsch C, Platzer C. Removing web spam links from search engine results[J]. Journal in Computer Virology, 2011, 7 (1):51-62

[4] 360 互联网安全中心. 2013 年中国网站安全研究报告[R]. [2014-01-01]. <http://awterbbwfk.15.yunpan.cn/lk/QpvTmqTwb9ci7>

[5] 360 互联网安全中心. 2013 年中国网购安全报告[R]. [2014-03-12]. <http://aqv4kwspvd.15.yunpan.cn/lk/Q4zjDEguzcwnx>

[6] Henzinger M R, Motwani R, Silverstein C. Challenges in Web search engines[C]//ACM SIGIR Forum. ACM, 2002;11-22

[7] Gyöngyi Z, Garcia-Molina H, Pedersen J. Combating web spam with TrustRank[C]//Proceedings of the 30th international conference on Very large data bases (VLDB 2004). 2004;576-587

[8] Wu B, Davison B D. Identifying link farm spam pages[C]//Special Interest Tracks and Posters of the 14th International Conference on World Wide Web. ACM, 2005;820-829

[9] Suhara Y, Toda H, Nishioka S, et al. Automatically generated spam detection based on sentence-level topic information[C]//Proceedings of the 22nd International Conference on World Wide Web Companion. 2013;1157-1160

[10] Chung Y, Toyoda M. A Method for Detecting Hijacked Sites by Web Spammer using Link-based Algorithms[J]. IEICE Transactions on Information and Systems, 2010, E93-D (6): 1414-1421

[11] Araujo L, Martinez-Romo J. Web spam detection; new classification features based on qualified link analysis and language models[J]. IEEE Transactions on Information Forensics and Security, 2010, 5(3):581-590

[12] Shen G, Gao B, Liu T Y, et al. Detecting link spam using temporal information[C]//Proceedings of the 6th IEEE International Conference on Data Mining. 2006;1049-1053

[13] Yu H, Liu Y, Zhang M, et al. Web spam identification with user browsing graph [M] // Information Retrieval Technology. Springer Berlin Heidelberg, 2009;38-49

[14] Liu Y, Chen F, Kong W, et al. Identifying Web Spam with the Wisdom of the Crowds[J]. ACM Transactions on the Web (TWEB), 2012, 6(1):2-32

[15] Fayyad U, Irani K. Multi-interval discretization of continuous-valued attributes for classification learning[C] // International Joint Conference on Artificial Intelligence(IJCAI). 1993;1022-1027

[16] Castillo C, Donato D, Becchetti L, et al. A reference collection for web spam[J]. ACM Sigir Forum. ACM, 2006, 40(2):11-24