



一种基于特征集构建的 Bagging 集成方法 及其在流量分类中的应用

钱亚冠¹, 关晓惠², 吴淑慧¹, 云本胜¹, 任东晓¹

(1. 浙江科技学院大数据科学系, 浙江 杭州 310023;

2. 浙江水利水电学院, 浙江 杭州 310018)

摘要: Bagging 是一种经典的分类器集成方法, 其有效性依赖于基分类器之间的差异度。通过遗传算法为每个基分类器构建独立的特征集, 目的是获得基分类器之间更好的差异性。同时, 根据不同基分类器的分类性能进行优化加权集成, 获得更好的泛化能力。最后, 采用 Softmax 回归作为基分类器, 将改进的 Bagging 集成方法应用到互联网流量分类, 实验结果表明, 改进方法相比经典 Bagging 方法在分类准确率上有显著提高, 与利用决策树集成的随机森林相比也有较好的性能提升。

关键词: Bagging 集成; 特征子集; 遗传算法; 流量分类

中图分类号: TP301.6

文献标识码: A

doi: 10.11959/j.issn.1000-0801.2018094

An approach of Bagging ensemble based on feature set and application for traffic classification

QIAN Yaguan¹, GUAN Xiaohui², WU Shuhui¹, YUN Bensheng¹, REN Dongxiao¹

1. Department of Big-Data Science, Zhejiang University of Science and Technology, Hangzhou 310023, China

2. Zhejiang University of Water Resources and Electric Power, Hangzhou 310018, China

Abstract: Bagging is a classic ensemble approach, whose effectiveness depends on the diversity of component base classifiers. In order to gain the largest diversity, employing genetic algorithms to get independent feature subset for each base classifier was proposed. Meanwhile, for better generalization, the optimal weights for the base classifiers according to their predictive performance were selected. Finally, refined Bagging ensemble based on simple Softmax regression was applied successfully in traffic classification. The experiment result shows that the proposed approach can get more improvement than the original Bagging ensemble in classification performance, and is better than the random-forests to a certain extent.

Key words: Bagging ensemble, feature subset, genetic algorithm, traffic classification

1 引言

传统的机器学习方法是在假设空间中寻找一

个最能接近真实分类函数的假设。为此, 机器学习研究者对各种分类模型进行了大量的研究, 力图提高单个分类器的泛化能力。人工神经网络、



支持向量机等分类模型在不同的应用领域取得了不错的分类性能,但是也存在单个分类器性能提升越来越难、训练时间越来越长的弊端,在大数据环境下尤其突出。集成方法的思路是把多个分类器集成起来,通过对多个分类器的结果进行某种方式的组合来决定最终的分类结果,以期取得比单个分类器更好的性能^[1]。

目前集成分类器的方法有多种,Boosting 和 Bagging 是两种典型集成方法^[1]。Boosting 采用序贯方式产生基分类器(base classifier),利用基分类器之间的相关性提升性能;而 Bagging 是并行方式产生基分类器,利用基分类器之间的不相关性来提升性能。目前的集成框架有采用决策树、神经网络^[2,3],甚至深度神经网络^[4]作为基分类器,也有选择支持向量机作为基本类器^[5]。但集成学习理论的本意是利用弱分类器来构建强分类器,同时弱分类器往往训练时间短,因此选择用简单的 Softmax 回归来构建强分类器。已有的研究表明^[1],基分类器之间差异度越大, Bagging 集成方法的分类性能越好。前期研究表明,即使采用相同的特征选择算法,在不同的训练集上(例如不同时段获得的流量数据)也可获得较大差异的特征子集^[6]。考虑到经典 Bagging 中的基分类器是在相同的特征集上构建分类模型,为进一步增大基分类器之间的差异度,提出每个基分类器采用独立的特征子集。同时,经典 Bagging 集成方法采用等权重投票表决的方式来给出最终的预测结果,而事实上每个基分类器的能力不同,为此采用带权投票方式来进行进一步提高分类精度。

最后,把改进的 Bagging 集成分类方法应用到互联网流量分类领域。流量分类是互联网领域中的一个重要应用,如何准确识别出流量的应用类型对于网络管理、流量控制及网络安全等具有重要的意义^[7]。由于互联网的复杂性、动态性,在各种网络应用层出不穷的情况下,如何准确识别出流量的应用类型目前仍然是个极

具挑战的课题,而利用分类器集成的方式可以克服上述动态性造成的分类误差。通过实际的流量数据进行实验,结果显示改进方法比经典的 Bagging 方法有显著的性能提升,与采用决策树集成的随机森林(random forest)方法相比也有提高,符合研究预期。

2 Bagging 集成方法

简单地说,集成学习就是利用多个分类器的能力来克服单个分类器的不足,图 1 给出了集成方法的框架结构^[1]。一般把参与集成的单个分类器称为基分类器,基分类器由基学习算法(base learning algorithm)训练获得,决策树、感知器等都可以作为基分类器。根据 Kearns 和 Valiant^[8,9]提出的强可学习与弱可学习理论,分类准确率只要略高于 50%的弱分类器(即比随机猜测略好)是可以增强为强分类器的。集成方法正是基于此理论,集成多个弱分类器后获得比单个强分类器更好的分类性能。

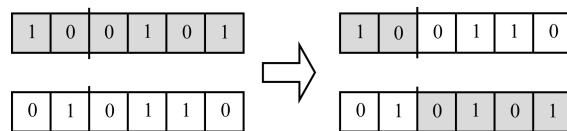


图 1 一个点位的交叉

目前集成分类器的方法有多种, Bagging 和 Boosting 是两种典型集成方法^[1]。经典 Bagging 集成方法采用 bootstrap 抽样方法^[10]获得多个训练集,在每个训练集上获得多个基分类器,最后通过投票的方式决定最终的分类预测标签。bootstrap 抽样构建的 Bagging 算法如下。

算法 1 Bagging 算法

输入: $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$;

基学习器 L ; 基学习器数量 N

for $i = 1, \dots, N$:

$h_i = L(D, D_{bs})$ /*这里 D_{bs} 是 bootstrap 分布*/

end for

输出: $H(x) = \arg \max_{y \in Y} \sum_{i=1}^N \text{sign}(h_i(x) = y)$

Bagging 是通过并行方式产生基分类器, 利用基分类器之间的差异性来提升性能。为方便分析, 假设分类标签集合为 $Y = \{-1, +1\}$, 目标函数为 f , 每个基分类器具有独立的泛化误差 ε , 即对于每个基分类器 h_i , 有 $P(h_i \neq f(x)) = \varepsilon$ 。把 N 个上述两分类器用 Bagging 方式集成后的假设函数为:

$$H(x) = \text{sign} \left(\sum_{i=1}^N h_i(x) \right) \quad (1)$$

由式 (1) 可知, 当超过一半的基分类器犯错时, 集成分类器 H 才犯错。根据 Hoeffding 不等式, 集成分类器 H 的泛化误差为:

$$P(H(x) \neq f(x)) = \sum_{k=0}^{N/2} \binom{N}{k} (1-\varepsilon)^k \leq \exp \left(-\frac{1}{2} N(2\varepsilon-1)^2 \right) \quad (2)$$

式 (2) 表明, 不相关的基分类器越多, 泛化误差越小。因此, 如何在有限的训练集上得到尽可能多的、差异性显著的基分类器是 Bagging 集成方法成功的关键。本文正是通过进一步强化基分类器的差异性来提升 Bagging 的分类能力。在大数据环境下, 选择 Bagging 作为集成框架的另一个优点是可以充分利用目前多核处理器的并行能力来产生基分类器。

3 基于特征子集构建方法

经典 Bagging 方法采用 bootstrap 重抽样来产生不同的训练集, 增加基分类器的个体差异度, 从而提高泛化能力。但是 bootstrap 产生的训练集与原始数据集仍有 63.2% 的重合度, 对于像 k -近邻这样的稳定分类器而言, 并不能产生个体差异很大的基分类器。考虑到经典 Bagging 集成方法是在相同的特征子集下构建基分类器, 从特征子集入手来增大基分类器的差异度。同时, 经典的 Bagging 方法假定每个基分类器的投票权重相等, 不符合基分类器能力不同的实际情况。本文提出

加权集成的思路, 利用梯度下降的优化方法获得权重系数。

3.1 基于遗传进化的特征子集选择

所谓特征选择, 就是从高维的特征空间中去除相关性强的冗余特征, 获得最优的特征子集。特征选择算法一般包括子集产生过程、评价函数、停止准则和验证过程这 4 个部分, 其中子集产生过程是搜索特征子集空间的过程, 是计算复杂度最高的部分。由于特征子集的搜索空间与特征数成指数关系, 用蛮力法搜索整个特征子集空间将是一个 NP 难问题。实际应用中通常采用启发式搜索, 本文采用随机搜索的遗传算法来获得不同训练集上的特征子集, 以保证训练的基分类器之间有最大的差异度。

特征选择的子集产生过程采用遗传算法进行搜索, 首先需要对染色体进行编码, 一个染色体表示一个特征子集。采用 0/1 方式编码染色体, 例如染色体编码为 00101000, 表示特征子集 {3, 5}, 即第 3 和第 5 个特征被选取, 这里假设用整数索引特征。第 2 个步骤是初始化一个种群 P , 它表示一个随机生成的染色体集合。第 3 个步骤是计算每个染色体的适应度, 模拟生物对环境的适应能力。本文中的适应能力是指该特征子集是否有利于分类器的性能提高, 因此定义适应度评价函数为:

$$F(C) = J(X_c, D) - \text{penalty}(X_c) \quad (3)$$

其中, C 表示染色体, X_c 表示染色体 C 对应的特征子集。 $J(X_c, D)$ 是对特征子集的评估, 取分类准确率为评价指标; $\text{penalty}(X_c)$ 是特征数目的惩罚项, 防止特征子集过大。第 4 个步骤是按照适应度对种群中的染色体排序, 适应度高的染色体会被高概率选中用于繁殖下一代, 本文采用基于排序轮盘赌的选取方法。第 5 个步骤是把选出的染色体进行交叉繁殖, 图 1 和图 2 是常见的几种交叉方式。第 6 个步骤是交叉繁殖后的后代染色体进行变异操作。最后用步骤 3 的适应度评价函



数评估新繁殖的染色体,如果优于其双亲染色体,则从种群中替换双亲染色体。步骤3至步骤7反复迭代执行,直到满足最优终止条件。由于上述进化过程中存在随机选择双亲染色体的行为,因此可以避免迭代过程陷入局部最优,最终有可能找到全局最优解。算法结束后,种群 P 中的染色体按适应度排序,据前列的染色体即是需要的特征子集。算法2描述了上述基于遗传算法的特征子集选择过程。

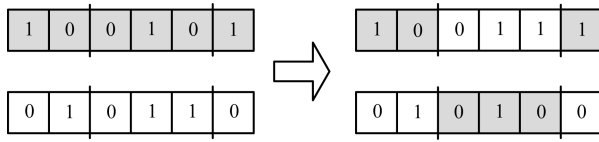


图2 两个点位的交叉过程

算法2 GAFeatureSelect //基于遗传算法的特征子集选择

输入: $A = \{a_1, a_2, \dots, a_m\}$ //特征空间

$D = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$

for $i = 1, \dots, K$ //初始化 K 个染色体 (特征子集)

$X_{c_i} = \text{RandomSelect}(A)$ //从特征空间随机选择若干特征

$C_i = \text{Code}(X_{c_i})$ //进行染色编码

end for

$P = \{C_i\}, i = 1, \dots, K$ //初始化种群

do

for $i = 1, \dots, |P|$

$F(C_i) = J(X_{c_i}, D) - \text{penalty}(X_{c_i})$ //计算特征子集的适应度

end for

$\text{Parent}_1 = \text{RouletteSelect}(F(C_i))$ //轮盘赌方式选择适应度高的子集

$\text{Parent}_2 = \text{RouletteSelect}(F(C_i))$

$C' = \text{Crossover}(\text{Parent}_1, \text{Parent}_2)$ //交叉繁殖新的特征子集

$C' = \text{Mutate}(C')$ //变异

if $F(C') \geq F(\text{Parent}_1)$ or $F(C') \geq F(\text{Parent}_2)$

Add(P, C') //把 C' 加入种群

Delete(P, Parent_1) or Delete(P, Parent_2)

end if

while $F(C') \leq \eta$

输出: $X_c = \text{Max}(P)$ //适应度最大的染色体作为最优特征子集返回

3.2 基分类器的加权集成

经典的 Bagging 方法假定每个基分类器在投票中的权重相等,而本文提出的方法是在不同的特征子集上训练基分类器,这些分类器在分类能力上会存在一定的差异,因此本文进一步提出加权集成的思路,赋予每个基分类器不同的投票权重。图3是改进后的 Bagging 集成框架,可以发现,每个基分类器是在单独的特征子集上构建的。假设训练集为 $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$, $y^{(i)} \in \{1, 2, \dots, k\}$ 是 k 个类标签的索引值。本文采用 Softmax 回归作为基分类器,它是 Logistic 回归往多分类器上的推广。Softmax 回归的假设函数为:

$$h(x) = [p(y^{(1)} | x), p(y^{(2)} | x), \dots, p(y^{(k)} | x)]^T$$

$$= \frac{1}{\sum_{j=1}^k e^{\theta_j^T x}} [e^{\theta_1^T x}, e^{\theta_2^T x}, \dots, e^{\theta_k^T x}]^T \quad (4)$$

其中, θ_i^T 是参数向量,假设函数 $h(x)$ 的输出是由每个类标签 $y^{(i)}$ 的后验概率构成的。

假设第 i 个 Softmax 回归基分类器的输出是向量 $h_i(x)$, 所有的输出向量经加权求和集成后通常不满足概率的规范性,采用 Softmax 函数再次变换到 $[0, 1]$ 区间:

$$P = \text{Softmax} \left(\sum_{i=1}^N w_i \times h_i(x) \right) \quad (5)$$

设 $P = [p_1, p_2, \dots, p_k]^T$, 满足 $p_i \in [0, 1]$, $\sum_{i=1}^k p_i = 1$ 。由此,重新赋权后的集成分类器的假设函数变为:

$$H(x) = \arg \max_i (P_i) \quad (6)$$

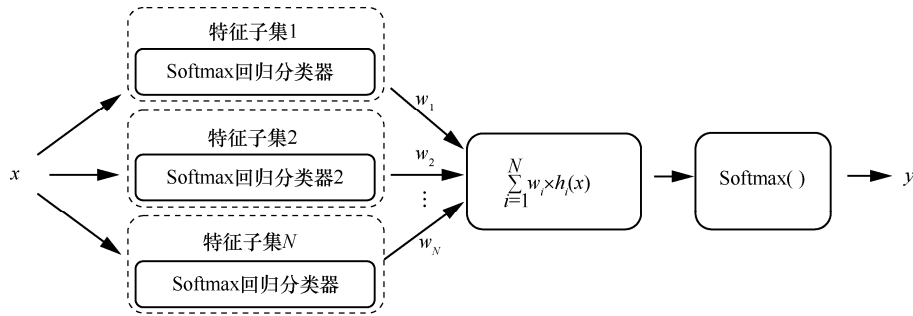


图3 基于特征子集构建的加权 Bagging 集成框架

由于两次代价函数容易过早地进入饱和区，采用交叉熵函数作为加权集成分类器的代价函数进行训练：

$$C = -\frac{1}{m} \sum_{i=1}^m \sum_{j=1}^k \mathbb{I}(H(x^{(i)}) = y_j) \ln p_j \quad (7)$$

其中， $\mathbb{I}(x)$ 是指示函数，即当 x 是 true 时， $\mathbb{I}(x) = 1$ ，否则 $\mathbb{I}(x) = 0$ 。 y_j 是第 j 个类的输出标签， p_j 是第 j 个类的后验概率。由式 (5) 可知，代价函数是权重向量 $W = (w_1, w_2, \dots, w_N)$ 的函数。可以通过迭代的方式更新权重向量：

$$w_i^{(t+1)} = w_i^{(t)} - \lambda \frac{\partial C}{\partial w_i^{(t)}} \quad (8)$$

$$\frac{\partial C}{\partial w_i} = -\frac{1}{m} \sum_{i=1}^m \sum_{j=1}^k \mathbb{I}(H(x^{(i)}) = y_j) [h_i(x^{(i)})]_j \ln(1 - p_j) \quad (9)$$

其中， $[\cdot]_j$ 表示向量中的第 j 个分量。通过最速梯度下降法迭代收敛到最权重 $W^* = (w_1^*, w_2^*, \dots, w_N^*)$ ，代入式 (6)，即最终的集成分类器，见算法 3。

算法 3 WeighedBagging

输入： $A = \{a_1, a_2, \dots, a_m\}$

$D = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$ ；

Softmax 回归学习器 L ；基学习器数量 N ；步长 $\lambda = 0.1$

for $i = 1, \dots, N$

$D_i = \text{bootstrap}(D, D_{\text{bs}})$ //bootstrap 抽样

$X_i = \text{GAFeatureSelect}(A, D_i)$ //为基分类

器构建特征子集

$h_i = L(D_i, X_i)$ //这里 D_{bs} 是 bootstrap 分布

$W = \text{random}(N)$ //随机初始基分类器的权重

end for

while(1)

$P = \text{Softmax}\left(\sum_{j=1}^N w_j \times h_j(x_i)\right)$ //训练基分类

器并将它们加权输出

$$\frac{\partial C}{\partial w_i} = -\frac{1}{m} \sum_{i=1}^m \sum_{j=1}^k \mathbb{I}(H(x^{(i)}) = y_j) [h_i(x^{(i)})]_j \ln(1 - p_j)$$

$$w_i^{(t+1)} = w_i^{(t)} - \lambda \frac{\partial C}{\partial w_i^{(t)}}$$

until $\sum_i \frac{\partial C}{\partial w_i} < 0.0001$

输出： $W^* = (w_1, w_2, \dots, w_N)$

4 流量分类中的应用

流量分类是互联网领域中的一个重要应用，如何准确识别出流量的应用类型对于网络管理、流量控制及网络安全等具有重要的意义。由于互联网的复杂性、动态性，如何准确识别出流量的应用类型目前仍然是个极具挑战的课题。由于数据分组加密技术的出现，深度分组检测 (DPI) 技术显得力不从心，而基于流量统计特征的机器学习方法不依赖于特征字串，因此成为流量分类领域的新兴技术^[11-13]。

所谓的基于机器学习的流量分类方法就是通过机器学习算法，从流量训练数据中建立分类模



型, 从而实现对流量类型的预测。这种方法的优点是可以克服数据加密的限制, 同时仅利用 IP 和 TCP 这两层数据分组头部的信息, 不受隐私保护的制约。但是, 互联网流量行为的高度不确定性, 导致不同地点、不同时间段获取的数据集之间存在较大的差异性。因此, 不同的数据集训练获得的模型对预测结果就会产生较大的波动, 而 Bagging 集成机器学习方法则可以有效地克服这种波动性。进一步在经典的 Bagging 集成方法基础上, 引入特征子集和加权集成基分类器的思想, 提高互联网高动态环境下的流量分类准确性。

4.1 流量数据集

本文实验室数据的来源有两个: 一是英国剑桥大学 Moore 等提供的公开流量数据集^[14], 二是从校网中心的某台交换机上获得的流量数据, 该交换机汇聚了某幢男生宿舍访问外网的所有网络流量。采用两个不同的数据集合在一起产生集成分类器, 目的是把不同地点和不同时间获取的流量数据训练集成分类器, 以期获得更好的泛化能力。Moore 等提供的实验数据是通过连续采集 24 h 的网络流量, 并按 28 min 为间隔随机抽取 10 个数据块, 本文只选用其中的 5 个数据块。

校网中心的数据选在周一晚上 21:30—22:30、周二下午 15:00—16:00、周三上午 10:30—11:30、周五晚上 19:30—20:30、周六下午 16:00—17:00 和周日上午 8:30—10:30。其中, 前 5 天的数据用于训练基分类器, 周日的数据用于测试。为保护隐私, 只截取数据分组的分组头部分, 并通过 Tcpdpriv 工具对 IP 地址进行了匿名化处理。由于 Moore 流量数据集由 248 个特征构成, 把校网中心获取的数据预处理成与 Moore 数据集同样的特征集。

考虑到 Moore 数据集中的某些流量类型的定义与本文的数据有差异, 最后两个数据集都统一选用 WWW、mail、FTP、P2P、database、multimedia 等几种类型的数据流。提出的集成方法在 Moore 的 5 个数据集和本文采集的 5 个数据集上共训练

10 个基分类器, 每个基分类器采用 Softmax 回归线性模型, 并采用自己的特征子集和加权重。

4.2 实验分析

把改进方法命名为 Bagging+, 与经典的 Bagging 方法、随机森林进行性能比较。性能评估采用召回率(recall)、精度(precision)和 F-measure 这 3 个指标:

$$\text{recall} = TP / P \quad (8)$$

$$\text{precision} = TP / (TP + FP) \quad (9)$$

$$\text{F-measure} = \frac{2TP}{2TP + FP + FN} \quad (10)$$

其中, P 为测试集中事先标识为正例的样本数, TP 为分类器正确预测为正例的样本数, FP 为被分类器错误地将正例预测为负例的样本数。F-measure 是召回率和精度的调和平均, 是一个能比较好地反映分类性能的综合指标。

表 1 给出了 Bagging+与经典 Bagging 方法之间的性能比较。尽管这两种集成方法均采用了 Softmax 回归线性分类器作为基分类器, 由于 Bagging+采用了不同的特征子集训练, 且采用优化权重集成, 可以发现, 召回率、精度和 F-measure 这 3 个指标均比经典 Bagging 方法有提升。WWW、mail、FTP-control 这 3 种流量类型的识别率提高幅度不大, 因为经典 Bagging 方法已具有非常好的识别率。但是 FTP-PASV、P2P 和 multimedia 的识别率提高显著。从 F-measure 指标看, multimedia 从 39.7%提高到 96.9%, 幅度超过 50%。而 FTP-PASV 和 P2P 尽管绝对指标只达到 67%左右, 但是比经典方法提高了 40%左右, 提升效果也是显著的。

表 2 给出了 Bagging+方法与随机森林的性能比较。随机森林的集成框架为 Bagging, 基分类器为决策树。由于决策树是一种不稳定的分类器, 不同数据集可以产生不同的决策树, 因此通过 Bagging 集成后能极大地提高分类性能。通过比较可以发现, 随机森林比经典的 Bagging 方法的性

表1 Bagging+方法与 Bagging 方法比较

	召回率		精度		F-measure	
	Bagging	Bagging+	Bagging	Bagging+	Bagging	Bagging+
WWW	99.1%	99.9%	97.6%	98.4%	98.3%	99.1%
mail	97.8%	100.0%	97.4%	99.6%	97.6%	99.8%
FTP-control	100.0%	100.0%	98.0%	98.0%	99.0%	99.0%
FTP-PASV	15.7%	60.5%	76.5%	84.4%	26.5%	67.5%
P2P	19.2%	51.1%	50.4%	98.0%	27.8%	67.1%
database	100.0%	100.0%	82.9%	88.9%	90.7%	94.1%
FTP-data	96.7%	100.0%	97.5%	99.8%	97.1%	99.9%
multimedia	33.3%	94.7%	49.2%	99.3%	39.7%	96.9%

表2 Bagging+方法与随机森林比较

	召回率		精度		F-measure	
	随机森林	Bagging+	随机森林	Bagging+	随机森林	Bagging+
WWW	97.1%	99.9%	99.3%	98.4%	98.2%	99.1%
mail	100.0%	100.0%	98.7%	99.6%	99.3%	99.8%
FTP-control	96.0%	100.0%	100.0%	98.0%	98.0%	99.0%
FTP-PASV	68.8%	60.5%	82.7%	84.4%	70.8%	67.5%
P2P	39.4%	51.1%	77.1%	98.0%	52.1%	67.1%
database	98.5%	100.0%	100.0%	88.9%	99.2%	94.1%
FTP-data	100.0%	100.0%	70.9%	99.8%	83.0%	99.9%
multimedia	92.0%	94.7%	91.4%	99.3%	91.7%	96.9%

能要提高不少,但改进的 Bagging+方法则比随机森林有较好的提升,特别是 P2P 的 F-measure 指标从 52.1%提高到 67.1%、FTP-data 从 83.0%提高到 99.9%。FTP-PASV 则有小幅下降,从 70.8%降到 67.5%,其他类别基本持平。因此,总体上看, Bagging+方法比随机森林方法有提高,特别是占比少的流量类别改进比较明显。

从实验结果看,改进 Bagging 集成方法通过遗传算法搜索特征子集,再通过加权集成基分类器可以提高分类正确率。Bagging 集成的性能主要取决于基分类器的差异度,遗传算法是一种随机搜索优化方法,在不同的数据集上这种随机性会得到差异性较大特征子集,而不同的特征子集又进一步加大了基分类器之间的差异性。通过交叉熵代价函数最小的方式获得的加权集成可以较好

地确定不同分类器对最后判决的贡献度,比同等权重的投票表决更加精确。

5 结束语

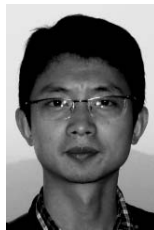
本文提出基于特征集构建的 Bagging 集成方法,利用遗传算法在不同的数据集上获取特征子集,不同的基分类器在独立的特征子集上训练获得,这样可以最大限度地挖掘基分类器之间的差异性。同时,还进一步采用加权集成的方法优化基分类器的投票组合,进一步提高分类器的预测性能。最后把这种改进的集成分类方法应用到互联网流量的分类中,目的是克服网络的动态性带来的分类模型的不稳定性。通过实际的流量数据测试,这种新的集成分类器具有较好的泛化能力,适合应用于互联网这种高度动态环境。



参考文献:

- [1] ZHOU Z H. Ensemble methods: foundations and algorithms[M]. Boca Raton: CRC Press, 2012.
- [2] LI H, WANG X, DING S. Research and development of neural network ensembles: a survey[J]. Artificial Intelligence Review, 2017: 1-25.
- [3] AMOZEGAR M, KHORASANI K. An ensemble of dynamic neural network identifiers for fault detection and isolation of gas turbine engines[J]. Neural Networks, 2016(76): 106-121.
- [4] INOUE H. Fast and accurate inference with adaptive ensemble prediction in image classification with deep neural networks[J]. arXiv preprint arXiv:1702.08259, 2017.
- [5] WANG Q, LUO Z H, HUANG J C, et al. A novel ensemble method for imbalanced data learning: bagging of extrapolation-SMOTE SVM[J]. Computational Intelligence and Neuroscience, 2017(3): 1827016.
- [6] 高文, 钱亚冠, 吴春明, 等. 网络流量特征选择方法中的分治投票策略研究[J]. 电子学报, 2015, 43(4): 795-799.
GAO W, QIAN Y G, WU C M, et al. The divide-conquer and voting strategy for traffic feature selection[J]. Acta Electronica Sinica, 2015, 43(4): 795-799.
- [7] 钱亚冠, 张旻. 基于过抽样技术的 P2P 流量识别方法[J]. 电信科学, 2014, 30(4): 109-113.
QIAN Y G, ZHANG M. P2P traffic identification based over-sampling technique[J]. Telecommunications Science, 2014, 30(4): 109-113.
- [8] KEARNS M. Learning Boolean formulae or finite automata is as hard as factoring[R]. Technical Report TR-14-88 Harvard University Aiken Computation Laboratory, 1988.
- [9] KEARNS M, VALIANT L. Cryptographic limitations on learning Boolean formulae and finite automata[J]. Journal of the ACM (JACM), 1994, 41(1): 67-95.
- [10] EFRON B, TIBSHIRANI R. An introduction to the bootstrap[M]. New York: Chapman & Hall, 1993.
- [11] TONGAONKAR A, TORRES R, ILIOFOTOU M, et al. Towards self-adaptive network traffic classification[J]. Computer Communications, 2015(56): 35-46.
- [12] SOYSALA M, SCHMIDT E G. Machine learning algorithms for accurate flow-based network traffic classification: evaluation and comparison[J]. Performance Evaluation, 2010, 67(6): 451-467.
- [13] SINGH H. Performance analysis of unsupervised machine learning techniques for network traffic classification[C]//2015 Fifth International Conference on Advanced Computing & Communication Technologies (ACCT), May 15-16, 2015, Haryana, India. Piscataway: IEEE Press, 2015: 401-404.
- [14] MOORE A W. Dataset[EB]. 2017.

[作者简介]



钱亚冠 (1976—), 男, 博士, 浙江科技学院理学院大数据科学系副教授, 主要研究方向为互联网流量分类、机器学习与大数据处理、对抗性机器学习。



关晓惠 (1977—), 女, 浙江水利水电学院副教授, 主要研究方向为机器学习与大数据处理、对抗性机器学习。



吴淑慧 (1975—), 女, 博士, 浙江科技学院理学院大数据科学系讲师, 主要研究方向为量子计算与机器学习。

云本胜 (1980—), 男, 博士, 浙江科技学院理学院大数据科学系讲师, 主要研究方向为数据挖掘、服务计算。

任东晓 (1982—), 女, 博士, 浙江科技学院理学院大数据科学系高级工程师, 主要研究方向为机器学习与大数据处理、对抗性机器学习。