高技术通讯 2009 年 第 19 卷 第 6 期:564 ~ 571

doi:10.3772/j.issn.1002-0470.2009.06.003

基于 MRMHC-LSVM 的 IP 流分类①

李文法② 段洣毅 陈 友 程学旗

(* 中国科学院计算技术研究所 北京 100190)

(** 中国科学院研究生院 北京 100039)

摘 要 提出了一种构建轻量级的 IP 流分类器的 wrapper 型特征选择算法 MRMHC-LSVM。该算法采用改进的随机变异爬山(MRMHC)搜索策略对特征子集空间进行随机搜索,然后利用提供的数据在无约束优化线性支持向量机(LSVM)上的分类错误率作为特征子集的评价标准来获取最优特征子集。在 IP 流数据集上进行了大量的实验,实验结果表明基于 MRMHC-LSVM 的流分类器在不影响分类准确度的情况下能够提高检测速度,与当前典型的流分类器 NBK-FCBF 相比,基于 MRMHC-LSVM 的 IP 流分类器具有更小的计算复杂度与更高的检测率。

关键词 流分类,特征选择,改进的随机变异爬山(MRMHC),线性支持向量机(LSVM)

0 引 言

IP 流分类在网络监控,QoS,入侵检测等领域应用广泛。为了能够快速地检测出系统的异常或者对某个特定结点进行实时监控,需要一种高效的 IP 流分类技术。随着网络的高速发展,网络中的数据量越来越大,并且这些数据中含有众多的相关与冗余信息,对这些数据进行修剪与剔除尤为重要。流分类技术必须和数据预处理技术结合起来才能满足现代网络的需求,因此基于特征选择的流分类技术成为当前研究的热点。特征选择针对 IP 流的高维特征空间存在大量的相关与冗余特征的特性,在此高维空间上应用搜索算法来寻找最优的特征子集,剔除那些相关与冗余特征。在得出的最优特征子集上建立的 IP 流分类器不仅可以减少分类器的计算复杂性,而且可以获得很好的检测效果。

特征选择有 filter 和 wrapper 两种模型[1],filter 模型利用数据本身的特性作为特征子集的度量指标,而 wrapper 模型利用机器学习算法的分类正确率作为特征子集的度量指标。一般来说 filter 模型的效率高,效果差;wrapper 模型的效率低,效果好。为了解决这两种特征选择模型存在的问题,发挥它们的优势,很多学者提出了结合 filter 模型和 wrapper 模型的 hybrid 模型[1,2]。虽然 hybrid 模型在性能上有一定的提高,但是效果并不理想。本文采用 wrapper 模型设计了一种高效的特征选择算法,即以改进的随机变异爬山(modified random mutation hill climbing, MRMHC)为搜索策略,以线性支持向量机(linear support vector machine,LSVM)为评估函数的算法,简称 MRMHC-LSVM 算法。该算法不仅可以克服 wrapper 模型计算资源耗用大的缺点,而且利用选择出的特征,在很大程度上提高了流分类器的检测率。

1 相关工作

轻量级的 IP 流分类含有流特征选择与流分类器两个部分,而特征选择包括搜索策略与评估函数。近年来,越来越多的研究者把目光聚集在基于机器学习的流分类上。在文献[3]中,作者采用爬山算法作为搜索策略,相关性与一致性作为评估函数的特征选择算法来选择最优的特征子集。基于该最优特征子集建立的分类器在检测效果影响不大的情况下,降低了时间复杂性,提高了检测速度。文献[4]给出了利用贝叶斯分析技术的分类器。他们首先从网络流数据中产生 248 个流特征,然后利用相关特征选择(correlation feature selection,CFS)方法选择出最优特征子集,实验结果表明,分类器只需要 20 个特征就能达到很好的分类性能。文献[5]提出了一种结合爬山算法与欧式距离的特征选择算法,实验

① 863 计划(2006AA01Z452,2007AA01Z416)和国家 242 信息安全计划(2005C39)资助项目。

② 男,1974 年生,博士;研究方向:流分类,网络安全,数据挖掘;联系人,E-mail: liwenfa@software.ict.ac.cn

(收稿日期:2008-04-18)

结果表明,不同的应用流可以被很好地划分开。当前很多研究都集中在基于特征选择算法的流分类器上,他们利用的特征选择算法是 filter 型的,虽然一定程度上提高了分类器的检测速度,但是分类性能却不理想。用本文提出的 MRMHC-LSVM 算法不仅可以提高特征选择的速度,而且基于它选出的特征建立的流分类器具有高检测速度与检测率。

2 特征选择算法的数学模型

给定一个特征子集 $F = \{f_1, f_2, \ldots, f_N\}$,$N$ 是特征集的大小。一个特征子集可以用一个二进制向量表示:$\mathbf{S} = (s_1, s_2, \cdots, s_N)$,$s_i \in \{0,1\}$,$i = 1,2,\cdots,N$。$s_i = 1$ 表示第 i 个特征 f_i 被选择,反之对第 i 个特征 f_i 不作选择。把无约束优化线性支持向量机在给定的特征子集 S 上所具有的性能 G(S)作为目标函数值,则特征选择问题转化为下列优化问题:

$$\max_{S} G(\mathbf{S}) \tag{1}$$

特征选择的求解优化问题 $\max_{S} G(\mathbf{S})$ 可以通过改进的随机变异爬山(MRMHC)策略来求解。

3 MRMHC-LSVM 特征选择算法

MRMHC-LSVM 算法是 wrapper 型特征选择算法,它利用改进的随机变异爬山(MRMHC)[6]作为搜索策略,对特征空间进行划分,然后在划分的空间上应用无约束优化线性支持向量机(LSVM)进行评估,当评估值或迭代次数达到一定的标准之后,就停止该算法。下面主要从无约束优化线性支持向量机 LSVM 和改进的随机变异爬山策略 MRMHC 两个方面来介绍该算法。

3.1 无约束优化线性支持向量机

支持向量机[7](support vector machine, SVM)是 Vapnik 等人提出的一类新型机器学习方法。它建立在统计学习理论基础之上,能够较好地解决小样本、高维数、非线性和局部最小点等实际问题。其思想由 图1说明。在图中,实心点和空心点代表两类样本,H 为分类超平面,H_1,H_2 分别代表各类中离 H 最近的样本且平行于 H 的面,它们之间的距离称为分割距离。所谓最优分类面就是要求不但能将两类正确分开,而且使分类间隔最大。H_1,H_2 上的样本点称为支持向量。在图 1 中,margin 是分类超平面的间隔,$\| w \|$ 是分类超平面法向量的范数。

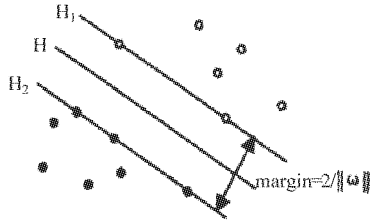

图 1 线性可分情况下的最优分类面

SVM 是建立在结构风险最小化原则基础上的。对于训练样本为 $\{(x_i, y_i)\}_{i=1}^{l} \subset \boldsymbol{R}^n \times \{1, -1\}$ 的二分类问题,根据统计学理论,可建立如下标准的线性 SVM 模型 A

$$\begin{cases} \min \dfrac{1}{2}(\omega^{\mathrm{T}}\omega) + C\sum\limits_{i=1}^{l}\xi_i \\ s.t.\ y_i((\omega^{\mathrm{T}}x_i) + b) \geqslant 1 - \xi_i,\ i = 1,2,\cdots,l \\ \xi_i \geqslant 0,\ i = 1,2,\cdots,l \end{cases} \tag{2}$$

其中 $C > 0$ 为正则化参数,$\xi_i (i = 1,2,\cdots,l)$ 为松弛变量,$\omega \in \boldsymbol{R}^n$ 为分类超平面的法向量,$b \in \boldsymbol{R}$ 为阈值。利用优化理论中的 Karush-Kuhn-Tucker 最优化条件(KKT 条件)和对偶理论,可得对偶优化模型 A'

$$\begin{cases} \max \sum\limits_{i=1}^{l}\alpha_i - \dfrac{1}{2}\sum\limits_{i=1}^{l}\sum\limits_{j=1}^{l}\alpha_i\alpha_j y_i y_j (x_i^{\mathrm{T}}x_j) \\ s.t. \sum\limits_{i=1}^{l} y_i\alpha_i = 0 \\ 0 \leqslant \alpha_i \leqslant C,\ i = 1,2,\cdots,l \end{cases} \tag{3}$$

其中 $\alpha_i (i = 1,2,\cdots,l)$ 为 lagrange 乘子。优化问题 A' 是一个凸二次规划问题,其局部最优解即为全局最优解。若 $\alpha^* = (\alpha_1^*, \alpha_2^*, \cdots, \alpha_l^*)^{\mathrm{T}}$ 为模型 A' 的最优解,则

$$\omega^* = \sum_{i=1}^{l}\alpha_i^* y_i x_i \tag{4}$$

根据 KKT 互补条件,最优解必满足

$$\alpha_i(y_i(\omega^{\mathrm{T}}x_i + b) - 1 + \xi_i) = 0,\ i = 1,2,\cdots,l \tag{5}$$

$$(C - \alpha_i)\xi_i = 0,\ i = 1,2,\cdots,l \tag{6}$$

由(4)-(6)式可知,对应于 lagrange 乘子 $\alpha_i = 0$ 的样本对分类问题不起什么作用,而只有对应于 lagrange 乘子 $\alpha_i > 0$ 的样本(支持向量)对计算 ω^* 起作用,从而决定分类结果,支持向量通常只是全体样本的中的很少一部分。求解上述问题后得到的广义最优线性分类器是

$$\begin{aligned} f(x) &= \mathrm{sgn}\{(\omega^{*\mathrm{T}}x) + b^*\} \\ &= \mathrm{sgn}\{\sum_{i=1}^{l}\alpha_i^* y_i (x_i^{\mathrm{T}}x) + b^*\} \end{aligned} \tag{7}$$

其中 sgn(·)为符号函数，b^* 为分类的阈值，可通过任意一个支持向量求得。模型 A 与 A' 都是约束条件下的优化模型，为了得到线性支持向量机的无约束优化模型，不妨定义

$$g_i(\omega,b) \triangleq 1 - y_i(\omega^{\mathrm{T}}x_i + b),\ i = 1,2,\cdots,l \quad (8)$$

则由(5)(6)式可得

$$\xi_i = \max\{0, g_i(\omega,b)\},\ i = 1,2,\cdots,l \quad (9)$$

将其代入标准的线性 SVM 模型 A 中，可得线性 SVM 的无约束优化模型 B

$$\min \frac{1}{2}\omega^{\mathrm{T}}\omega + C\sum_{i=1}^{l}\max\{0, g_i(\omega,b)\} \quad (10)$$

由最优化理论知，模型 A 与模型 B 等价，模型 B 目标函数中的前两项恰好体现了统计学习理论中的结构风险最小化原则。其中前一项反映了模型的置信范围，后一项反映了模型的训练误差。注意到

$$\|\xi\|_1 = \sum_{i=1}^{l}\max(0, g_i(\omega,b)) \quad (11)$$

$$\|\xi\|_\infty = \max\{0, g_1(\omega,b),\cdots,g_l(\omega,b)\} \quad (12)$$

如果用向量 $\xi = (\xi_1,\xi_2,\cdots,\xi_l)^{\mathrm{T}}$ 的 ∞ 范数来度量模型的训练误差，则可得无约束优化模型 C

$$\min\Phi(\omega,b) = \frac{1}{2}\omega^{\mathrm{T}}\omega + C\max\{0, g_1(\omega,b),\cdots,g_l(\omega,b)\} \quad (13)$$

与 A，A' 相比，SVM 无约束优化模型 B，C 在数学形式上更加简洁明了。但由于优化问题 B，C 是一个和最大值函数有关的一类不可微优化问题，从而给求解带来了困难。一条可行的途径是利用光滑化技术将不可微优化问题转化为可微优化问题，从而易于求解。本文利用极大熵方法作为求解优化问题 C 的一种近似解法。极大熵方法的基本思想是[8,9]：对于极大极小问题 $\min\phi(x) = \max\{f_1(x), f_2(x),\cdots, f_m(x)\}$，利用最大熵原理推导出一个可微函数 $\phi_p(x = \frac{1}{p}\ln(\sum_{i=1}^{m}\exp(pf_i(x))))$，通常称为极大熵函数。用该可微函数来逼近最大值函数 $\phi(x)$，从而把不可微优化问题转化为可微优化问题，使问题简化。通过引入极大熵函数，问题 C 的求解被转化为如下的无约束优化问题 D

$$\min\Phi_p(\omega,b) = \frac{1}{2}\omega^{\mathrm{T}}\omega + \frac{C}{p}\ln(1 + \sum_{i=1}^{l}\exp(pg_i(\omega,b))) \quad (14)$$

其中 $p > 0$ 是参数。由极大熵函数的逼近性质[10]和得出的相关定理[11,12]可以得出，问题 C 和 D 是等价的，并且优化问题 D 的任一局部最优解都是全局最优解。下面给出标准无约束优化算法的子程序。

基本算法：

步骤 1　给定任一初始点 $(\boldsymbol{\omega}^{(0)}, b^{(0)})$ 和正则化参数 C，令 p 为一个充分大的常数；

步骤 2　用无约束优化算法子程序进行 $\Phi_p(\boldsymbol{\omega}, b)$ 的最小化计算；

步骤 3　将最优解 $(\boldsymbol{\omega}^*, b^*)$ 代入(7)式，从而得到广义最优线性分类器

3.2　MRMHC 搜索策略

随机变异搜索策略 RMHC[6] 如同模拟退火算法、遗传算法一样是属于随机搜索策略。它们对搜索空间进行随机划分，在理论上可以得到全局最优解。随机搜索策略存在的困难是当搜索的空间比较大时，算法复杂性增加，得到最优解的过程复杂化。随机变异搜索策略 RMHC 利用 0，1 的二进制编码串 S 来表示特征空间，如长度为 N 的 0，1 串表示特征空间中含有 N 个特征，其中 0 表示该特征不被选择，1 表示该特征被选择。RMHC 的核心思想是：在每一次空间迭代的过程中随机变异 M 个特征，算法过程如下：

(1) 初始化长度为 N 的特征串 S，M 个特征被赋为 1，剩下的 $N - M$ 个特征被赋为 0。

(2) 对特征串 S 应用无约束优化支持向量机在训练集上训练，得出分类器的平均分类错误率 $F(S)$。

(3) 对 S 中 M 个特征进行随机变异。

(4) 回到(2)，直到 $F(S)$ 达到一定的标准或者到达最大迭代次数。

$F(S)$ 的计算方法为

$$F(S) = P_{\mathrm{error}}(S) = \frac{1}{C}\sum_{i=1}^{C}\gamma_i \quad (15)$$

其中 $P_{\mathrm{error}}(S)$ 是基于特征串 S 的分类器的所有类的平均分类错误率，C 是类的数目，γ_i 是每一类的分类错误率。

RMHC 划分速度与空间降维能力可以通过一种类似模拟退火的思想实现得到加强。提高划分速度和降维能力首先在迭代初期每次变异的数目要大，即 M 要大，而在迭代后期，当接近满足的评估标准时，变异的特征数目 M 要小，所以我们对 M 进行了补充，给出了 M 的一个计算公式：

$$M = M_{\max} * \min\left[\frac{(I_{\max} - i_{\mathrm{current}})}{I_{\max}}, P_{\mathrm{error}}(S)\right] \quad (16)$$

式中 M_{max} 是每次迭代允许变异的最大特征数目，I_{max} 是最大迭代次数，$i_{current}$ 是当前迭代次数。在对特征空间进行快速降维的同时,我们希望选出的特征子集空间比较小,这样基于此特征子集的分类器结构就简单,更利于对 IP 流进行快速的分类。MRMHC 不仅对 M 进行了修改,同时在 $F(S)$上也进行了改进,改进后的 $F(S)$如下:

$$F(S) = \alpha \cdot P_{error}(S) + (1 - \alpha) \cdot \frac{|S|}{N} \quad (17)$$

$|S|$是 S 中被选的特征数目，N 是特征总数目。$\frac{|S|}{N}$ 是 S 中被选特征数目在总特征数目上的占有率。平均分类错误率 $P_{error}(S)$ 与占有率 $\frac{|S|}{N}$ 之间通过 α 来权衡，α 越大，$F(S)$更强调分类错误率，α 越小$F(S)$更强调被选择特征的数目,通过权衡,就会使得选出的符合要求的特征子集含有更少或最少的特征数目。

4 基于 MRMHC-LSVM 的流分类器

基于 MRMHC-LSVM 的 IP 流分类器利用 MRMHC-LSVM 选出的特征子集建立 IP 流分类器,分类器使用无约束优化支持向量机。它的详细流程见图 2。

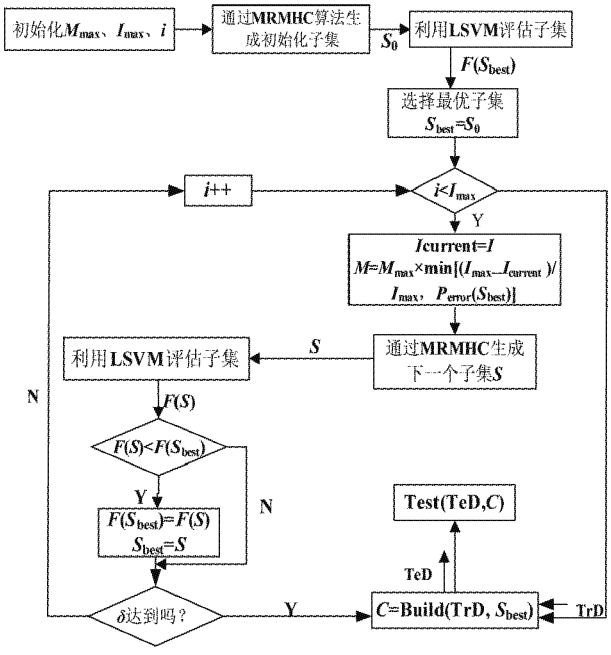

图 2 基于 wrapper 型特征选择的流分类流程图

首先初始化最大迭代次数 I_{max}，每次迭代最大变异数目 M_{max}，当前迭代次数 i。然后初始化特征子集 S_0，在 S_0 上建立无约束优化支持向量机分类器,对分类器的性能进行测试,得出分类器的评估值 $F(S_{best})$，且 $S_{best} = S_0$。进入迭代循环之后,在每一次循环过程中,首先计算本次循环需要变异的特征数目 M,然后基于 M 产生新的特征子集 S,在 S 上建立无约束优化支持向量机分类器,测试出分类器的评估值 $F(S)$。把 $F(S)$ 与 $F(S_{best})$ 进行比较,如果 $F(S)$ 小于 $F(S_{best})$ 则 $F(S_{best}) = F(S)$，$S_{best} = S$，否则判断评估标准 δ 是否达到。当基于选择的特征子集的分类器的评估值达到了预先的标准或者最大迭代次数已经达到,就停止特征选择。在 S_{best} 上建立分类器,然后把建立的轻量级流分类器应用于 IP 流分类中。由于应用特征选择算法之后,特征空间下降,特征空间中相关与杂音特征被剔除,基于选择后特征的分类器的结构变得简洁清晰,检测速度应该会提高很多,分类器的检测率也有一定程度的提高。下节将通过实验验证。

5 实验研究

为了验证特征选择算法对流分类计算复杂性与分类性能的影响,我们进行了大量的实验。首先用 MRMHC-LSVM 在给定的数据集上进行特征选择,然后比较结合 MRMHC-LSVM 的流分类器与没有使用特征选择的流分类器在系统建模时间、检测速度、检测率上的差异,最后比较了基于 MRMHC-LSVM 的流分类器与 Moore[4]用贝叶斯方法在文献[13]的数据集上建立的 NBK-FCBF 流分类器在检测速度与检测率上的差异。

5.1 IP 流数据集与 IP 流特征

我们实验的数据集来自文献[13],在这个数据集中每一个流是由一个元组定义的,元组包括许多特征,协议类型如互联网控制信息协议(Internet control message protocol,ICMP),传输控制协议(TCP),用户数据报协议(user datagram protocol,UDP),在 UDP,TCP 中的主机端口号,TCP 连接的持续时间等。每一个流有 248 个这样的独立特征,并且这些特征是流分类器的输入参数。另外,每一个流还有一个定义流应用类型的类,如 WWW,P2P,MAIL,BULK 等。关于流中特征和类详细的描述见文献[14,15]。

数据集是由 01 到 10 的 10 个小数据集组成,每个小数据集也是由一定数量的流构成的。在整个数据集中 WWW 的流数目是 328091,MAIL 的流数目是

28567,整个数据集所有流的数目为 377526。而其中类 INTTERACTIVE(INT)与类 GAMES 的流数量很小,分别是 110 与 8,这些类由于流数量太小,没有提供足够的信息来进行分类,所以在流分类中将不考虑。

5.2 实验方案

在编号为 01 到 10 的 10 个数据集上进行相应的实验,实验分为训练与测试两个部分。首先在 10 个数据集中的一个数据集上训练模型,然后把其它 9 个数据集作为测试集来测试在前面一个数据集上建立的分类器的性能。这样在每一个数据集上都建立模型,然后用其它 9 个数据集作为测试,就得到了 10 组数据,每一组数据对应一个分类器的性能。在我们的实验中,首先用 MRMHC-LSVM 在训练集上选择特征子集,然后在此特征子集上建立分类器,最后用测试集测试分类器的性能,我们用到的分类器算法是无约束优化支持向量机[7]。分类器的性能主要由训练时间、测试速度、检测效果三个方面来评价,其中检测效果主要有两个指标:

准确率:被正确分类的流数目与流的总数目之比;

召回率:对每一类,被正确分类的流数目与这一类流的总数目之比。

从两个指标的定义可以看出,准确率是针对所有的类的衡量指标,而召回率是针对每一类的衡量指标。所有的实验都是在同一平台下完成,该平台的配置为:Intel processor 3.0GHz,1.00GB RAM,Windows 操作系统。

5.3 特征选择

首先通过 MRMHC-LSVM 对 01-10 的每一个数据集进行特征选择,然后在选择的特征上应用无约束优化支持向量机算法建立分类器。表 1 是 10 个数据集特征选择的结果,其中 MRMHC-LSVM 栏是 MRMHC-LSVM 选择后的特征数目,FCBF 栏是 FCBF (fast correlation based filter)选择后的特征数目。从表中可以看出,FCBF 与 MRMHC-LSVM 对特征都有很大程度的削减,这样提高了分类器的测试速度。在 07-10 数据集上,MRMHC-LSVM 选择的特征数目远远小于 FCBF 选择的特征数目,从这点可以看出,MRMHC-LSVM 在测试速度与训练时间上可能要优于 FCBF,在后面的实验结果中我们也会对此进行论述。

表 1 每一个训练集特征选择之后选择的特征数目

训练集	FCBF	MRMHC-LSVM
01	4	7
02	3	4
03	6	8
04	7	7
05	11	9
06	6	5
07	15	7
08	16	10
09	28	13
10	49	11

5.4 特征选择在计算能力与分类能力上的效果

我们对基于所有特征的分类器与基于 MRMHC-LSVM 的分类器在计算性能与分类能力上的效果进行了比较。在计算性能上主要从建模时间与测试速度两个方面进行对比;在分类能力上主要从分类正确率与召回率上进行对比。建模时间的对比见图 3。图中记录了基于所有特征的分类器与基于 MRMHC-LSVM 选择特征的分类器在 10 个训练集上建立模型的时间。纵坐标的最高值是 1000s。在我们测试平台下最长的建模时间 971s,而基于特征选择的分类器的平均建模时间只有 12.3s。从图 3 可以看出,基于 MRMHC-LSVM 的分类器建模时间相对于基于所有特征分类器的建模时间大大减少,这是因为特征的削减使得无约束优化支持向量机分类器的结构简单化,这样建立这种结构的时间也就相应的减少了。

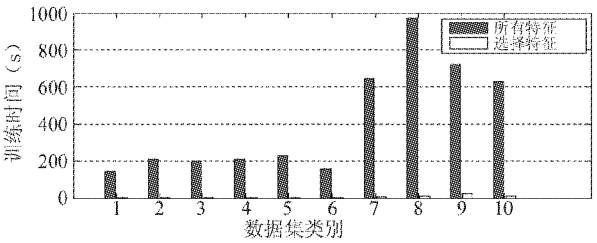

图 3 在 10 个训练集上基于所有特征的无约束优化支持向量机分类器与基于选择特征的分类器训练时间对比图

图 4 提供了两种类型的分类器在测试速度上的区别,测试速度也是经过标准化处理,最大的测试速度是每秒测试 50043 个流,这个速度标准化为 1,其它的测试速度相对于 50043 也进行了标准化,详细的比较见图 4。基于 MRMHC-LSVM 选择特征分类器的测试速度大约是基于所有特征分类器的 1 倍。这是因为基于 MRMHC-LSVM 选择特征的分类器在

结构上比基于所有特征的分类器要精简,这样测试速度也就相应地跟上,这得力于特征选择的效果。

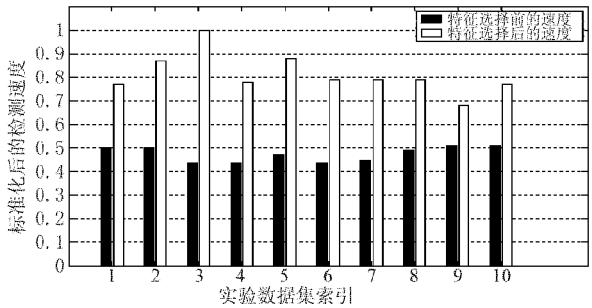

图 4　基于所有特征的无约束优化支持向量机分类器与基于选择特征的分类器检测速度对比

在上面详细的比较了两种类型的分类器在计算性能上的差异,结果表明基于 MRMHC-LSVM 选择特征的分类器在建模时间与测试速度上要远远优于基于所有特征的分类器。建模时间与测试速度上的优势会不会导致分类器分类能力的下降呢? 答案是否定的。图 5 到图 8 详细地展现了基于 MRMHC-LSVM 选择特征的分类器与基于所有特征的分类器在分类能力上的区别。分类能力的比较主要从两方面:比较分类器在检测所有类别上的能力(图 5);比较分类器在检测单个类别 WWW, MAIL, P2P 上的能力(图 6,图 7,图 8)。在图 5 中,基于 MRMHC-LSVM 选择特征的分类器在 10 组数据中,大部分点的数据要高于基于所有特征的分类器,并且它的平均准确率是 0.9889,要高于基于所有特征的分类器的平均准确率是 0.9862,将近 0.27% 的提高。这表明基于选择特征的分类器不仅在测试速度上有优势,同时检测能力上也有一定的提高。可见特征选择能够剔除杂音与冗余特征,不仅精简了分类器的结构,同时也提高了分类器的检测能力。

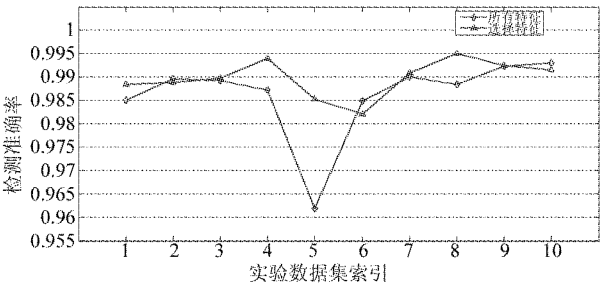

图 5　基于所有特征的无约束优化支持向量机分类器与基于 MRMHC-LSVM 选择特征的分类器在所有类上召回率对比

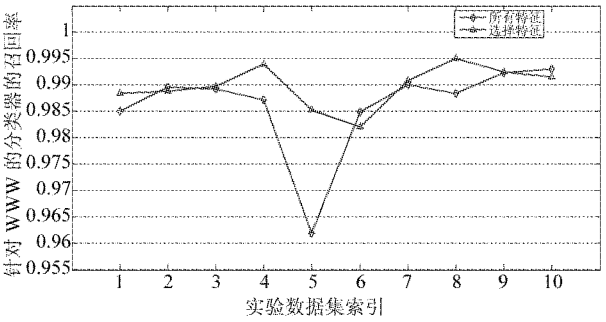

图 6　基于所有特征的无约束优化支持向量机分类器与基于 MRMHC-LSVM 选择特征的分类器在 WWW 上召回率对比

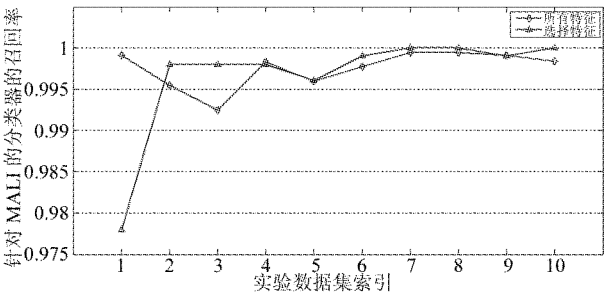

图 7　基于所有特征的无约束优化支持向量机分类器与基于 MRMHC-LSVM 选择特征的分类器在 MAIL 上召回率对比

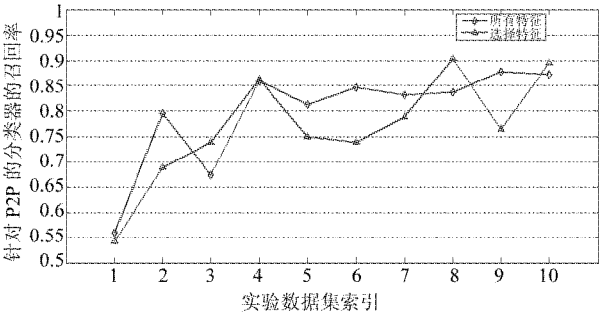

图 8　基于所有特征的无约束优化支持向量机分类器与基于 MRMHC-LSVM 选择特征的分类器在 P2P 上召回率对比

图 6 到图 8 显示了基于 MRMHC-LSVM 选择特征的分类器与基于所有特征的分类器在 WWW, MAIL, P2P 上的召回率的比较。在 WWW, MAIL, P2P 上基于所有特征的平均召回率分别为 0.9952, 0.9975, 0.796。从这三幅图可以看出,基于特征选择的分类器在 WWW 上有更高的平均召回率,我们计算出其平均召回率是 0.9972, 而在 MAIL (0.9956), P2P(0.760)上的平均召回率要低于基于所有特征的分类器。WWW 上的平均召回率差不多增长 0.5%,在 MAIL, P2P 上的平均召回率下降将近 0.19% 和 3.3%。特别是在 P2P 上下降比较多,但是

在 P2P 上基于所有特征的分类器其召回率(79.6%)也不高,这说明数据集中的特征不能很好地反映出 P2P 这种应用,如果再在这样的基础上进行特征选择,效果可能会很差。虽然在平均召回率上有升有降,但是升降差别都不是很大,而在建模时间与测试速度上却有很大的提高,这说明特征选择在保证分类能力不减弱的情况下可以大大提高分类器的时间性能。

5.5 MRMHC-LSVM 与 NBK-FCBF 在计算能力与分类能力上的比较

Williams 在文献[3]中比较了无约束优化支持向量机分类器与 NBK 分类器的计算性能,结果表明 NBK 比无约束优化支持向量机有更小的建模时间,但是测试速度远小于无约束优化支持向量机。这是针对拥有相同数目特征时的结果对比,但是 MRMHC-LSVM 选择的特征数目比 FCBF 选择的特征数目要少,特别是在 07 - 10 四个数据集上,参见表 1。这表明在计算性能上无约束优化支持向量机要优于 NBK。在分类能力上我们也进行了比较,比较结果见图 9、图 10。图 9 比较了基于 MRMHC-LSVM 的分类器与 NBK-FCBF 在 ALL、WWW、MAIL、P2P 上的分类能力,图 10 是基于 MRMHC-LSVM 的分类器与 NBK-FCBF 在 ALL、WWW、MAIL、P2P 上的检测准确率的对比。从图 9,图 10 可知,基于 MRMHC-LSVM 的分类器无论在分类能力上还是在 ALL、WWW、MAIL、P2P 上的检测准确率上均高于 NBK-FCBF,并且平均高出将近 4%,特别是在 P2P 上高出接近 30%,这说明 MRMHC-LSVM 选择的特征子集效果好,特别是针对 P2P 其效果更好。

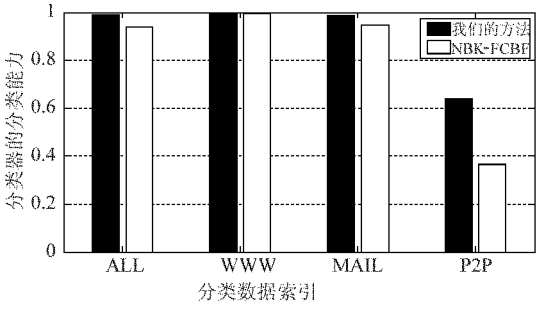

图 9 基于 MRMHC-LSVM 的分类器与 NBK-FCBF 在 ALL,WWW,MAIL,P2P 上的检测率对比

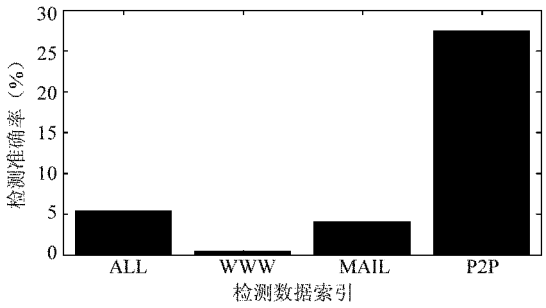

图 10 基于 MRMHC-LSVM 的分类器在检测准确率上相对于 NBK-FCBF 的百分比增长

6 结 论

依据网络流能够准确实时地判断出其所属的应用类型是很多网络活动需要研究的重要内容。传统的基于端口与基于负载内容的分析技术已经显示出很多的弊端,现在越来越多的研究者把目光投向基于机器学习的流分类,这种流分类中的每一个流是由一些特征组成,这些特征共同反映了这个流所属的应用类别。越来越多的特征被产生用于流分类,但是这些特征中有许多冗余与杂音特征,所以特征选择在流分类中扮演着十分重要的角色。

本文给出了一种 wrapper 型的特征选择算法 MRMHC-LSVM 来建立轻量级的流分类器,MRMHC-LSVM 不仅可以降低时间复杂性,而且可以获得最优的特征子集。基于 MRMHC-LSVM 的分类器具有更快的检测速度与更高的分类能力。我们在流数据集上进行了大量的实验,实验结果表明基于 MRMHC-LSVM 的流分类器在不降低分类能力的前提下可以大大降低分类器的计算复杂性,同时与 NBK-FCBF 相比,具有更快的检测速度与更高的分类能力。

在实验中我们发现 P2P 的召回率很低,这可能与特征选择算法有关,在将来的研究中我们将要去研究更好的特征选择算法与分类算法来解决这一问题。本文在单个类别上只比较了 WWW,MAIL,P2P,还有其它如 BULK,SERVICES 等很多类没有进行实验验证与分析,在下一步工作中,我们将继续开展这方面的工作。

参考文献

[1] 陈友,程学旗,李洋等.基于特征选择的轻量级入侵检测系统.软件学报,2007,18(7):1639-1651

[2] Yu L, Liu H. Feature selection for high-dimensional data: a fast correlation-based filter solution. In: Proceedings of the 20th International Conference on Machine Learning, Washington DC, USA, 2003. 856-863

[3] Williams N, Zander S, Armitage G. A preliminary performance comparison of five machine learning algorithms for

practical IP traffic flow classification. *ACM SIGCOMM Computer Communication Review*, 2006, 135(5):5-16

[4] Moore A W, Zuev D. Internet traffic classification using bayesian analysis techniques. In: Proceedings of the ACM SIGMETRICS, Banff, Canada, 2005. 50-60

[5] Zander S, Nguyen T, Armitage G. Automated traffic classification and application identification using machine learning. In : Proceedings of the IEEE LCN, Melbourne, Australia, 2005. 250-257

[6] Skalak D B. Prototype and feature selection by sampling and random mutation hill climbing algorithms. In: Proceedings of the 11th International Conference on Machine Learning, New Brunswick, NJ, Canada, 1994. 293-301

[7] Vapnik V. The Nature of Statistical Learning Theory. New York: Springer Verlag, 1995.17-23

[8] 李兴斯.非线性极大极小问题的一个有效解法.科学通报,1991,36(19):1448-1451

[9] 李兴斯.一类不可微优化问题的有效解法.中国科学(A辑),1994,24(4):371-377

[10] 唐焕文,张立卫,王雪华.一类约束不可微优化问题的极大熵方法.计算数学, 1993,15(3):268-275

[11] 唐焕文,张立卫.凸规划的极大熵方法.科学通报,1994,39(8):682-684

[12] 张志华,郑南宁,史罡.极大熵聚类算法及其全局收敛性分析.中国科学(E辑),2001,31(1):50-70

[13] University of Cambridge Computer Laboratory. The data-sets for flow classification is available. http://www.cl.cam.ac.uk/research/srg/netos/nprobe/data/papers/sigmetrics/index.html,2008

[14] Moore A W, Zuev D, Crogan M. Discriminators for use in flow-based classification: [technical report]. London: Queen Mary University of London, Department of Computer Science, 2005. 6-13

[15] Moore A W. Discrete content-based classification-a data set: [technical report]. Intel Research, Cambridge, 2005. 8-18

IP flow classification based on MRMHC-LSVM

Li Wenfa[*,**], Duan Miyi[*], Chen You[*,**], Cheng Xueqi[*]

([*] Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100080)

([**] Graduate University of Chinese Academy of Sciences, Beijing 100039)

Abstract

This paper proposes a wrapper feature selection algorithm MRMHC-LSVM aiming at modeling lightweight flow classifiers by using the modified random mutation hill climbing (MRMHC) approach as the search strategy to specify candidate subsets for evaluation and using the linear support vector machine (LSVM) algorithm as the wrapper approach to obtain the optimum feature subset. The feasibility of the algorithm was examined by conducting several experiments on flow datasets. The experimental results show that a classifier based on the proposed approach can greatly improve the computational performance without negative impact on the classification accuracy. Further more, this approach is able not only to have smaller resource consumption, but also to have higher classification accuracy than the Naïve Bayes method with Kernel density estimation after Fast Correlation-Based Filter (NBK-FCBF).

key words: flow classification, feature selection, modified random mutation hill climbing (MRMHC), linear support vector machine(LSVM)