概念漂移数据流分类中的多源在线迁移学习算法

秦一休[1] 文益民[1,2] 何 倩[1]
(桂林电子科技大学计算机与信息安全学院 广西 桂林 541004)[1]
(广西可信软件重点实验室桂林电子科技大学 广西 桂林 541004)[2]

摘 要 现有概念漂移处理算法在检测到概念漂移发生后,通常需要在新到概念上重新训练分类器,同时"遗忘"以往训练的分类器。在概念漂移发生初期,由于能够获取到的属于新到概念的样本较少,导致新建的分类器在短时间内无法得到充分训练,分类性能通常较差。进一步,现有的基于在线迁移学习的数据流分类算法仅能使用单个分类器的知识辅助新到概念进行学习,在历史概念与新到概念相似性较差时,分类模型的分类准确率不理想。针对以上问题,文中提出一种能够利用多个历史分类器知识的数据流分类算法——CMOL。CMOL算法采取分类器权重动态调节机制,根据分类器的权重对分类器池进行更新,使得分类器池能够尽可能地包含更多的概念。实验表明,相较于其他相关算法,CMOL算法能够在概念漂移发生时更快地适应新到概念,显示出更高的分类准确率。

关键词 多源迁移学习,在线学习,概念漂移,数据流分类
中图法分类号 TP391 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2019.01.010

Multi-source Online Transfer Learning Algorithm for Classification of Data Streams with Concept Drift

QIN Yi-xiu[1] WEN Yi-min[1,2] HE Qian[1]
(School of Computer Science and Information Security, Guilin University of Electronic Technology, Guilin, Guangxi 541004, China)[1]
(Guangxi Key Laboratory of Trustworthy Software, Guilin University of Electronic Technology, Guilin, Guangxi 541004, China)[2]

Abstract The existing algorithms for classification of data streams with concept drift always train a new classifier on new collected data when new concept is detected, and forget the historical models. This strategy always lead to insufficient training of classifier in a short time, because the training data for the new concept are always not collected enough in initial stage. And further, some existing online transfer learning algorithms for classification of data streams with concept drift only take advantage of single source domain, which sometimes lead to poor classification accuracy when the historical concepts are different with the new concept. Aiming to solve these problems above, this paper proposed a multi-source online transfer learning algorithms for classification of data stream with concept drift (CMOL), which can utilize the knowledges from multiple historical classifiers. The CMOL algorithm adopts a dynamic classifier weight adjustment mechanism and updates classifier pool according to the weights of classifiers in it. Experiments validate that CMOL can adapt to new concept faster than other corresponding methods when concept drift occurs, and get higher classification accuracy.

Keywords Multi-source transfer learning, Online learning, Concept drift, Data stream classification

在很多生产实践中,数据按照时间顺序以"流"的形式不断产生,例如网页访问产生的数据、社交网络产生的数据等,这些不断产生的数据形成了"数据流"。数据流环境中包含的概念在不同的时间段内可能会不相同,即数据流中的数据分布常常随着时间的推移不断发生变化。例如,用户的网页访问记录随着社会热点的改变而不断发生变化,这将导致数据流中可能会包含多个概念。数据流中的这种数据分布变化被称为概念漂移。概念漂移问题的出现,打破了传统机器学习数据分布固定的假设,导致传统的机器学习算法无法适用于概念漂移问题,给机器学习算法带来了挑战。

概念漂移由Schlimmer等[1]于1986年首次提出,此后很多学者对概念漂移问题进行了深入研究并取得了不少成果:Hulten等[2]提出了CVFDT算法,Kolter等[3]提出了DWM算法,Jr等[4]提出了RCD算法,Li等[5]提出了REDLLA算法,Zhao等[6]提出了CDOL算法,文益民等[7]提出了RCOTL算法,等。文献[8-11]针对概念漂移的相关工作进行了详细

到稿日期:2018-06-02 返修日期:2018-07-11 本文受国家自然科学基金(61363029,61866007),广西区自然科学基金(2018GXNSFDA138006),广西可信软件重点实验室立项资助课题(KX201721),广西高校图像图形智能处理重点实验室课题资助项目(GIIP201505),广西云计算与大数据协同创新中心项目(YD16E12)资助。

秦一休(1992-),男,硕士,主要研究方向为机器学习、迁移学习;文益民(1969-),男,博士,教授,主要研究方向为机器学习、数据挖掘与推荐系统,E-mail:ymwen2004@aliyun.com(通信作者);何 倩(1979-),男,博士,教授,主要研究方向为云计算、分布式计算和信息安全。

综述。此外，概念漂移的研究成果在很多领域得到了广泛应用，如教育[12]、医药[13]等。

现有概念漂移处理算法，在检测到概念漂移发生后，通常认为现有分类器不再适合新到概念，需要为新到概念重新训练分类器，而往往“遗忘”以往训练的分类器。由于在概念漂移发生初期，能够获取到的属于新到概念的样本较少，导致新建的分类器在短时间内无法得到充分训练，分类性能通常较差，因此在概念漂移发生时，如何使新建分类器在短时间内快速适应新到概念显得尤为重要。被“遗忘”的历史分类器往往是使用大量样本训练得到的，通常会有较好的分类性能，如果新到概念的样本与某个历史分类器具有相似的数据分布，此时若丢弃已经训练充分的历史分类器将会造成资源的浪费。

近年来，迅速兴起的迁移学习尝试利用已有相关但不同领域的知识来解决目标领域仅含有少量标签样本或没有标签样本的问题[14]。迁移学习的目的与概念漂移发生初期仅能获取到少量带标签样本的情景类似，因此在概念漂移发生初期，可以合理地使用历史分类器的知识辅助新到概念分类器进行分类，迁移学习为解决概念漂移初期分类模型的分类准确率较差的问题提供了新思路。

迁移学习算法通常基于一个批量式的学习环境，即假定目标领域训练样本部分已知或全部已知，使用已经获取的目标领域样本选择性地迁移源领域知识。但是在概念漂移场景中，样本以“流”的形式到达，每次仅能获取到一个训练样本或者部分训练样本，在获取到样本后需要立即对样本进行分类，并根据获取到的样本动态地调整分类模型，因此现有迁移学习算法难以适用于概念漂移分类问题。

现有的多源在线迁移学习算法仅适用于静态数据流，在发生概念漂移时，无法及时地调整分类模型以适应新到概念。针对此问题，本文提出一种分类器池更新规则，以保证分类器池中的分类器尽可能属于不同的概念，并结合现有的多源在线迁移学习算法提出了一种基于多源在线迁移学习的概念漂移分类算法——CMOL。

1　相关工作

概念漂移问题的出现，引起了学者的广泛关注。文益民等[8]将概念漂移分类算法分为两种：基于单分类器的概念漂移分类算法和基于多分类器的概念漂移分类算法。基于单分类器的概念漂移分类算法使用单个分类器处理概念漂移，根据数据流中的概念变化不断地调整分类器。Hulten 等[2]利用决策树的特点提出了 CVFDT 算法，该算法是典型的基于单分类器的概念漂移分类算法。在 CVFDT 算法中，当决策树的某一个叶子不再满足 Hoeffding 边界时，为该节点产生一个替换子树，当该子树的准确率高于原有节点时，使用该子树替换原有节点。陆莉莉等[15]指出 CVFDT 算法需要经常检测子树的准确率，这会影响算法的效率，基于此，提出了能够使用并行化窗口进行概念漂移检测的算法——S-CVFDT。李燕等[16]结合决策树和贝叶斯提出了 CDSMM 算法，该算法使用决策树作为基本分类模型，并且使用贝叶斯分类器过滤数据流中的噪声以提升算法的抗噪性能。Li 等[5]提出了 REDLLA 算法，REDLLA 算法借用 k-means 聚类算法的思想为无标签样本打标签，在决策树的叶子节点中聚类，根据聚簇之间的差异判断是否发生了概念漂移。Vinayagasundaram 等[17]提出了 AGDT 算法，该算法采用高斯约束来确定节点中的最佳划分属性。为了处理概念漂移，AGDT 算法使用固定大小的窗口来确定决策树中需要更新的节点。

文益民等[8]将基于多分类器的概念漂移分类算法分为两种：使用集成学习方法对数据流分块进行学习的算法和使用在线学习方法对整个数据流进行学习的算法。

前者通常在每个数据块上训练一个分类器，并且按照一定的规则存储训练得到的分类器。Street 等[18]首先使用集成学习的思想处理概念漂移，提出了 SEA 算法。在 SEA 算法中，每获取到一个数据块，该算法就使用该数据块新建一个分类器。如果分类器池未满，则将用新到数据块训练的分类器直接添加到分类器池中；如果分类器池已满，当且仅当新建分类器能有效提升分类器池的分类性能时才被添加，并且根据预先设置的度量标准替换掉不必要的历史分类器。Ramamurthy 等[19]提出了 EB 算法，该算法将训练的分类器存入全局分类器池中并且不删除分类器。EB 算法在每次获取到数据块时，使用全局分类器池中的分类器判断新到数据块所对应的概念是否为新概念。若是新概念，则需要在新到数据块上新建分类器，并加入到全局分类器池；若不是新概念，则从全局分类器池中选取部分分类器判断新到数据块所对应的概念是否是重现概念。若是重现概念漂移，则从全局分类器池中挑选合适的分类器对数据流进行分类。Brzezinski 等[20]提出了 AUE2 算法，AUE2 算法能够处理多种类型的概念漂移。AUE2 算法每获取到数据流中的一个样本，就使用分类池中的分类器加权投票进行预测。AUE2 算法每获取到一个数据块的样本，就在该数据块上新建一个分类器。当分类器池的数量小于某一阈值时，直接将新建分类器存储到分类器池中，否则使用新建分类器替换掉准确率最差的分类器。当分类器池的内存占用大于某一阈值时，Hoeffding 树中最不活跃的叶子节点会被修剪掉从而减少内存占用。Sun 等[21]提出了 DTEL 算法，该算法在每获取到一个数据块时，在新到数据块上新建一个分类器，并将其存入分类器池，在达到分类器池上限时，按照“多样性”原则调整分类器池。DTEL 算法在调整分类器池时使用 Q 统计度量每个分类器与其他分类器之间的“多样性”。根据 Q 统计的结果获取分类器池“多样性”的最大值，并且根据该值删除使得多样性取得“最大值”对应的分类器。

后者通常在获取到数据流中的一个样本时，使用集成分类器中的每个分类器对获取到的样本进行学习，然后使用获取到的样本更新集成分类器中的每个分类器，当集成分类器无法对数据流进行准确预测时，需要新建分类器。Kolter 等[3]提出了 DWM 算法，该算法每隔一定量的样本，就会根据设定的阈值删除分类器池中的历史分类器，并且根据分类器池对当前获取到的样本的分类情况判断是否需要新建分类器，并将新建的分类器加入到分类器池中。辛轶等[22]提出了 IKm-DHecoc 算法，该算法引入了基于 KnnModel 算法的动态层次编码，对每个数据块进行预学习，并根据学习结果更新编码。

迁移学习在近年来得到了快速发展，文献[14,23-25]对迁移学习算法进行了深入综述。Wu 等[26]结合多源迁移学习和在线学习提出了 HomOTLMS 算法，该算法为每个源领域分类器设置一个权重，用权重代表每个源领域分类器对目标领域的分类能力的大小，并且根据分类性能不断调节各个分类器的权重。唐诗淇等[27]结合局部精度和 OTL 算法[6]，提出了 LC_MSOTL 算法。该算法每获取到目标领域中的一个样本，就使用局部精度选择最合适的源领域分类器与目标领域分类器的加权组合来对获取到的样本进行分类。上述在线迁移学习算法仅能处理静态数据流。当数据流中包含概念漂移时，无法及时调整分类模型。

Zhao 等[6]结合在线迁移学习和概念漂移提出了 CDOL 算法。CDOL 使用单个历史分类器的知识辅助新到概念进行分类和学习，在检测到概念漂移时选择权值较大的历史分类器作为辅助分类器。文益民等[7]注意到 CDOL 算法选取距离当前概念最近的历史分类器作为辅助分类器时可能会产生“负迁移”。基于此，他们提出了基于在线迁移学习的重现概念漂移数据流分类算法——RCOTL。在发生概念漂移时，RCOTL 使用“负相似度”选取最合适的历史分类器辅助新到概念进行学习。RCOTL 与 CDOL 类似，仅能使用单个历史分类器进行迁移学习，无法从根本上避免 CDOL 的不足。

针对 CDOL 算法的不足，本文受 HomOTLMS 算法[26]的启发，提出了一种能利用多个历史分类器实施迁移学习的概念漂移分类算法——CMOL。CMOL 采取了一种分类器池更新规则——替换分类器池中权值最大的分类器或在分类器池中加入新分类器，以保证分类器池中的分类器尽可能属于不同的概念，以使用多个历史分类器实施多源在线迁移学习。

2 CMOL 算法

CMOL 算法可分为两个阶段，即发生概念漂移前的学习与分类阶段和发生概念漂移后的模型调整阶段。

2.1 CMOL 算法的学习与分类阶段

CMOL 算法每获取到数据流中的一个待分类样本 x_t，就使用分类器池中的分类器和目标分类器加权集成以对 x_t 进行分类。集成分类器如式(1)所示：

$$\tilde{y}=\mathrm{sign}\left(\sum_{i=1}^{n}p_t^i f^{Si}(x_t)+p_t^{\mathrm{T}}f_t^{\mathrm{T}}(x_t)\right) \tag{1}$$

其中，x_t 为数据流中的第 t 个样本，p_t^i 表示在对样本 x_t 进行分类时分类器池中第 i 个分类器的权重，p_t^{T} 表示在获取到样本 x_t 时目标分类器 f_t^{T} 的权重。

使用集成分类器对数据流中获取到的样本 x_t 分类完成之后，获取到样本 x_t 的真实标签。CMOL 根据分类器池中的每个分类器对样本 x_t 分类的正确与否来调节其对应的权重，分类器池中各个分类器权重的调节方式如式(2)和式(3)所示：

$$z_t^i=I(\mathrm{sign}(y_t f^{Si}(x_t))<0) \tag{2}$$

$$p_{t+1}^i=p_t^i\beta^{z_t^i} \tag{3}$$

其中，$I(x)$为指示函数，$I(x)=\begin{cases}1, & x=\mathrm{true}\\ 0, & x=\mathrm{false}\end{cases}$，$\mathrm{sign}(x)$为符号函数。目标分类器的权重则按照式(4)和式(5)的方式进行调节。

$$z_t^{\mathrm{T}}=I(\mathrm{sign}(y_t f_t^{\mathrm{T}}(x_t))<0) \tag{4}$$

$$p_{t+1}^{\mathrm{T}}=p_t^{\mathrm{T}}\beta^{z_t^{\mathrm{T}}} \tag{5}$$

2.2 CMOL 算法的模型调整阶段

当获取到一定量的样本之后，需要进行概念漂移检测，检测到概念漂移后进入模型调整阶段。本文使用 OWA 算法进行概念漂移检测。OWA 算法每隔一定量的样本就使用准确率来判断是否发生了概念漂移。OWA 算法的具体描述详见文献[6]。设 t 时刻的分类器池为 CP^t。

当检测到概念漂移发生后，需要及时调整当前分类模型。模型的具体调整步骤为：1)将使用检测到概念漂移之前的样本训练的目标分类器 f_t^{T} 按照一定的规则加入到分类器池 CP^t 中。在加入到分类器池时判断分类器池容量是否达到上限。若未达到容量上限，则将 f_t^{T} 直接加入分类器池；若达到上限，则首先移除分类器池中权值最大的分类器，之后将使用检测到概念漂移之前的样本训练的目标分类器 f_t^{T} 加入到分类器池。2)将分类器池 CP^t 中分类器的权重进行重置。3)在检测到概念漂移后的数据流上再新建一个目标分类器 f_{t+1}^{T}，其权值的设置见算法 1。调整完分类模型之后，模型将进入学习阶段。

在进行分类器池调整时，主要有两种情况需要分析：1)分类器池的容量大于整个数据流所包含的概念数量；2)分类器池的容量小于整个数据流所包含的概念数量。

1)当分类器池的容量大于整个数据流中所包含的概念数量时，就能保证分类器池中至少有两个分类器对应于同一个概念。当发生概念漂移时，若新到概念为重现概念，则分类器池中必然有一个分类器对新到概念具有最高的分类准确率，该历史分类器会具有最高的权重，其对应的概念即为新到概念，此时替换掉权值最大的分类器不会使得分类器池中所包含的概念数量发生变化。若新到概念为新概念，且权值最大的分类器对应的概念在分类器池中有两个分类器，则替换掉权值最大的分类器，分类器池中概念的数量会增加一个，即增加了分类器池中分类器的“多样性”。利用分类器池中的分类器进行迁移学习时，对于任何一个目标域，找到相似性更高的源领域的可能性会更高。若新到概念为新概念，且权值最大的分类器对应的概念在分类器池中只有一个分类器，则替换掉权值最大的分类器，分类器池中分类器对应的概念数量不会减少。

2)当分类池的容量小于整个数据流中所包含的概念数量时，与情况 1)类似，只是无法确保分类器池中至少有两个分类器属于同一概念，但还是能确保分类器池中分类器对应的概念数量不会减少。

综上所述，替换掉分类器池中权值最大的分类器能够尽可能地保证分类器池的“多样性”，还能够保证分类器池中包含尽可能多的知识，从而保证在实施迁移学习时，从分类器池中找到与新到概念分类器相似的分类器的可能性更大。

2.3 CMOL 算法的整体流程

CMOL 算法的伪代码如算法 1 所示。第 1 行表示初始化样本缓存区；第 3 行表示获取到数据流中的一个样本 x_t；第 4—8 行表示对获取到的样本 x_t 进行分类，其中第 5 行表示使用多个历史分类器与目标分类器加权对获取到的样本进行分类，第 7 行表示分类器池为空时仅使用目标分类器对获取到

的样本进行分类；第 9 行表示获取到样本 x_t 的真实标签；第 10 行表示将新到样本 x_t 缓存起来；第 11－16 行表示使用真实标签计算每个历史分类器对 x_t 的分类结果并更新其对应的权重；第 17－18 行表示利用真实标签计算目标分类器的分类结果并更新其对应的权重；第 19－22 行表示更新目标分类器 f_t^T，其中第 19 行表示计算目标分类器的分类损失，w_t 表示目标分类器的权值向量，第 20－22 行表示使用分类损失更新目标分类器；第 23 行表示是否到了检测概念漂移的时刻；第 24 行表示进行概念漂移检测；第 25－35 行表示检测到概念漂移后，根据指定规则删除掉分类器池中权值最大的分类器；第 36 行表示将目标分类器纳入分类器池；第 37 行表示清空样本缓存；第 38－40 行表示重新初始化分类器池中每个分类器的权重；第 41 行表示在检测到概念漂移后采集的样本上新建一个目标分类器；第 42 行表示初始化新建目标分类器的权重。

算法 1　CMOL 算法伪代码

输入：数据流 DS，分类器池 $CP^t=\{\}$，惩罚系数 C，权重衰减系数 $\beta\in(0,1)$，概念漂移检测窗口大小 p

输出：集成分类器：$\tilde{y}=\mathrm{sign}(\sum_{i=1}^{n}p_t^i f^{Si}(x_t)+p_t^T f_t^T(x_t))$

初始化：$f_1^T=(0,0,\cdots,0)$

1. PS=｛｝；
2. for t=1，2，…，DS. size() do
3.　receive instance：$x_t\in R^n$；
4.　if $CP^t!=\emptyset$
5.　predict：$\tilde{y}=\mathrm{sign}(\sum_{i=1}^{n}p_t^i f^{Si}(x_t)+p_t^T f_t^T(x_t))$；
6.　else
7.　predict：$\tilde{y}=f_t^T(x_t)$；
8.　end if
9.　receive the true label：$y_t\in\{-1,+1\}$；
10.　PS. add((x_t,y_t))；
11.　if $CP^t!=\emptyset$
12.　for each classifier f^{Si} in CP^t
13.　$z_t^i=I(\mathrm{sign}(y_t f^{Si}(x_t))<0)$；
14.　$p_{t+1}^i=p_t^i\beta^{z_t^i}$；
15.　end for
16.　end if
17.　$z_t^T=I(\mathrm{sign}(y_t f_t^T(x_t))<0)$；
18.　$p_{t+1}^T=p_t^T\beta^{z_t^T}$；
19.　suffer loss：$l_t=\max\{0,1-y_t(w_t x_t)\}$；
20.　if $l_t>0$ then
21.　$f_{t+1}^T=f_t^T+\tau_t y_t k(x_t,x_t)$　where $\tau_t=\min\{C,\frac{l_t}{k(x_t,x_t)}\}$；
22.　end if
23.　if mod(t，p)==0
24.　if OWA(PS)
25.　if CP^t. size()>=CP^t. MaxSize
26.　maxW=0.0；
27.　Del_i=0；
28.　for each classifier f^{Si} in CP^t
29.　if $p_t^i>$maxW；
30.　maxW=p_t^i；
31.　Del_i=i；
32.　end if
33.　end for
34.　CP^t. remove(Del_i)；
35.　end if
36.　CP^t. add(f_t^T)；
37.　PS=｛｝；
38.　for each classifier f^{Si} in CP^t
39.　$p_{t+1}^i=(1/(CP^t.\mathrm{size}()+1))$；
40.　end for
41.　$f_{t+1}^T=(0,0,\cdots,0)$；
42.　$p_{t+1}^T=(1/(CP^t.\mathrm{size}()+1))$；
43.　end if
44.　end if
45. end for

3　实验结果与分析

3.1　实验方法和实验数据

本文将 CMOL 算法与 PA 算法和 CDOL 算法进行对比以验证其有效性，CDOL 算法的描述详见文献[6]。在实验中，所有算法的基分类器均为 PA。本文将采用 Letter 数据集、RTG 数据集、waveform 数据集、SEA 数据集进行实验。Letter 数据集来自于 UCI 数据库，该数据集是真实数据集。RTG 数据集、waveform 数据集、SEA 数据集均来自于数据流开源学习框架 MOA[28]，该框架在数据流挖掘领域被广泛使用。

RTG 原始数据集共包含 40 维连续属性，4 个原始类别，共计 42000 个样本。在 RTG 数据集中，第 3 个原始类仅包含 2397 个样本。在 RTG 原始数据集中并未包含概念漂移，为了产生概念漂移，每次从每个原始类中分别随机抽取一定数量的样本，并为不同的原始类按照表 1 的要求重新标注正类和负类的类别标签。由于不同次抽取的属于同一个原始类的样本仅有类别标签发生了变化，即 $p(y|x)$ 发生了变化，从而产生了概念漂移。RTG 数据集生成的概念漂移数据集中共包含 4 个概念，为了使数据流中包含更多的概念，按照表 1 的顺序将每个新概念依次再重复产生两次（样本不重复），共产生 12 个概念。需要说明的是：第 12 个概念中的样本由前 11 个概念随机抽取后剩余的样本组成。在生成第 12 个概念时，由于第 3 个原始类包含的数据量较少，因此该概念包含每个原始类各 197 个样本，故第 12 个概念仅包含 788 个样本。其他概念中，每个概念均包含 800 个样本，包含每个原始类各 200 个样本，所有的样本均为随机选择。

表 1　RTG 数据集概念定义表

Table 1　Concept's definitions of RTG dataset

原始类别	1－800	801－1600	1601－2400	2401－3200
1	－	＋	＋	－
2	－	＋	－	＋
3	＋	－	＋	－
4	＋	－	－	＋

注：＋代表正类，－代表负类

waveform 原始数据集共包含 21 维连续属性，3 个原始类别，共计 5500 个样本，每一个类别由 3 个基波中的 2 个基波组合而成。waveform 原始数据集中并未包含概念漂移，为了产生概念漂移，waveform 数据按照 RTG 数据集类似的方式

打标签，waveform 按照表 2 所列的分布打标签，从而产生概念漂移。waveform 数据集生成的概念漂移数据集中共包含 3 个概念，为了使数据流中包含更多的概念，按照表 2 的顺序将每个新概念依次再重复两次(样本不重复)，共产生 9 个概念。需要说明的是：第 9 个概念中的样本由前 8 个概念随机抽取后剩余的样本组成，这导致了第 9 个概念共包含 700 个样本，其中包含第一个原始类 236 个样本，第 2 个原始类 189 个样本，第 3 个原始类 275 个样本。其他概念中，每个概念均包含 600 个样本，每个原始类各 200 个样本，所有的样本均为随机选择。

表 2 waveform 数据集概念定义表

Table 2 Concept's definitions of waveform dataset

原始类别	1－600	601－1200	1201－1800
1	－	＋	－
2	＋	－	－
3	＋	＋	＋

注：＋代表正类，－代表负类

SEA 数据集是突变式概念漂移数据集[27]，含两个类标签、三维连续属性，其中仅前两维属性是与类别相关的，第三维是无关属性的，每一维属性的取值范围均为[0，10]。该数据集含有 4 个生成函数，每个数据生成函数满足 $f_1+f_2\leqslant\theta$，其中 f_1 和 f_2 代表了 SEA 数据集的前两维，θ 是阈值，取值分别为 9，8，7 和 9.5。根据 θ 的不同取值可形成 4 个概念，每个 θ 值对应的概念包含 1000 个样本，按照上述 θ 值的顺序产生 4 个新概念。为了使数据流中包含更多的概念，对每个 θ 值按照上述顺序依次再重复两次，共产生 12 个概念，因此 SEA 数据集共包含 12000 个样本。由于 SEA 数据集中的 θ 在不同的概念中取值不同，并且 θ 不是一个连续值，因此 SEA 数据集是一个突变型的概念漂移数据集。

Letter 数据集共包含 16 维连续属性，26 个字母，20000 个样本。本文使用 Letter 数据集中的字母 ABCD 制作 ABCD 数据集，使用字母 EFGH 制作 EFGH 数据集，使用字母 NOPQ 制作 NOPQ 数据集。Letter 原始数据集中并未包含概念漂移，为了产生概念漂移，Letter 数据按照 RTG 数据集类似的方式重新标注正类和负类的类别标签，从而产生概念漂移。在各个字母产生的概念漂移数据集中，ABCD 数据集、EFGH 数据集和 NOPQ 数据集均包含 3 个新概念。为了使数据流中包含更多的概念并且考虑到数据量的关系，按照表 3 的顺序将每个新概念依次重复 1 次，之后再将第 1 个和第 2 个新概念再重复 1 次，因此共产生 8 个概念。在 ABCD 数据集、EFGH 数据集和 NOPQ 数据集中，除最后一个重复概念外，每个概念均包含 400 个样本，每个字母各 100 个样本，所有的样本均为随机选择。将每个字符中剩余的数据组成最后一个重复概念。ABCD 数据集、EFGH 数据集和 NOPQ 数据集的概念分布如表 3 所列。

由于不同数据集之间的分布不尽相同，因此本文在不同数据集上采用不同的核函数和惩罚系数。在 RTG 数据集上，概念漂移检测窗口的大小 $p=200$，PA 的高斯核系数 $\delta=0.5$，惩罚系数 $C=2.0$。在 SEA 数据集上，概念漂移检测窗口的大小 $p=200$，PA 的高斯核系数 $\delta=0.5$，惩罚系数 $C=0.5$。在 waveform 数据集上，概念漂移检测窗口的大小 $p=100$，PA 的高斯核系数 $\delta=1.0$，惩罚系数 $C=2.0$。在 Letter 数据集上，概念漂移检测窗口的大小 $p=100$，PA 的高斯核系数 $\delta=1.0$，惩罚系数 $C=2.0$。为了探讨分类器池的更新策略，本文考虑了另外两种其他方法，即 CMOL-I 和 CMOL-II。CMOL-I 表示在分类器池达到上限时，移除最先存储的分类器。CMOL-II 表示在分类器池达到上限时，对分类器池中的分类器按权值大小进行排序，并找到权值差异最小的两个分类器，并移除权值较小的分类器。CMOL，CMOL-I 和 CMOL-II 算法最多缓存 5 个历史分类器。

表 3 ABCD，EFGH 和 NOPQ 数据集概念定义表

Table 3 Concept's definitions of ABCD，EFGH and NOPQ

字母	1－400	401－800	801－1200
A(E)(N)	－	＋	＋
B(F)(O)	－	＋	－
C(G)(P)	＋	－	＋
D(H)(Q)	＋	－	－

注：＋代表正类，－代表负类

3.2 实验结果

实验中对上述所采用的数据集均随机生成 50 份数据，最终结果为 50 次实验的平均值。本文给出各种算法的累积准确率以及即时准确率变化图。各种算法在不同数据集上的累积准确率如表 4 所列，即时准确率变化图如图 1－图 6 所示。

表 4 在人工数据集上的累积准确率

Table 4 Cumulative accuracy on several synthetic datasets

准确率	PA	CDOL	CMOL	CMOL-I	CMOL-II
RTG	0.5040	0.5293	**0.5878**	0.5840	0.5821
waveform	0.6847	0.7070	**0.8098**	0.8068	0.8070
SEA	0.9205	0.9215	**0.9352**	0.9349	0.9343
ABCD	0.6066	0.6962	**0.8070**	0.8058	0.8062
EFGH	0.6039	0.6914	**0.7837**	0.7825	0.7831
NOPQ	0.6042	0.6941	**0.8006**	0.7996	0.7998

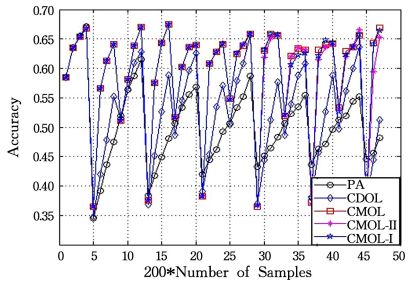

图 1 各种算法在 RTG 数据集上的即时准确率

Fig. 1 Instant accuracy of several algorithms on RTG dataset

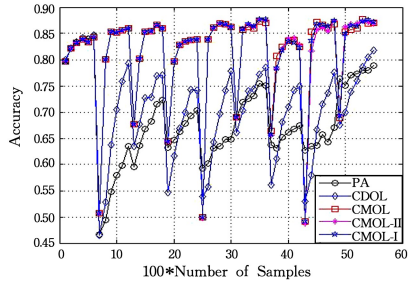

图 2 各种算法在 waveform 数据集上的即时准确率

Fig. 2 Instant accuracy of several algorithms on waveform dataset

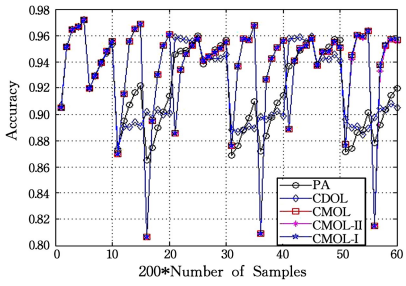

图 3　各种算法在 SEA 数据集上的即时准确率

Fig. 3　Instant accuracy of several algorithms on SEA dataset

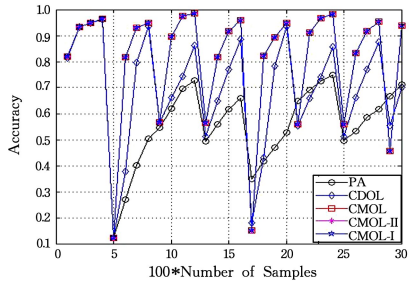

图 4　各种算法在 ABCD 数据集上的即时准确率

Fig. 4　Instant accuracy of several algorithms on ABCD dataset

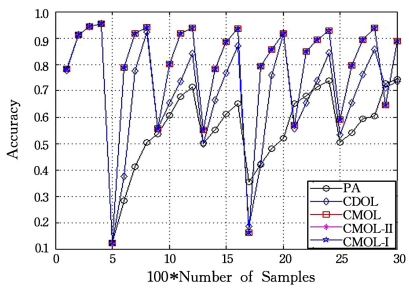

图 5　各种算法在 EFGH 数据集上的即时准确率

Fig. 5　Instant accuracy of several algorithms on EFGH dataset

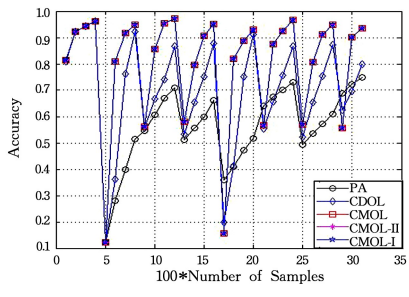

图 6　各种算法在 NOPQ 数据集上的即时准确率

Fig. 6　Instant accuracy of several algorithms on NOPQ dataset

从表 4 所列的累积准确率可以看出，与基于单源在线迁移学习的数据流分类算法 CDOL 相比，本文提出的 CMOL 算法能够体现出较好的分类准确率。在 SEA 数据集上，CDOL 算法并不能体现出迁移效果，表明在该数据集上使用单个历史概念无法有效地辅助新到概念进行学习。与 CDOL 算法进行对比，本文提出的 CMOL 算法能够体现出较高的分类准确率，表明本文提出的 CMOL 算法可以有效地弥补单个历史概念的不足。

从图 1－图 2 以及图 4－图 6 的即时准确率可以看出，CDOL 算法在检测到概念漂移时，相较于 PA 能够较快速地适应新到概念。相较于 CDOL 算法，本文提出的 CMOL 算法在概念漂移发生初期能够在短时间内恢复模型的分类准确率，从而使得模型更快速地适应新到概念。从图 4 的即时准确率可以看出，CDOL 算法在检测到概念漂移时，无法及时恢复分类模型的分类准确率。而与 CDOL 算法进行对比，本文提出的 CMOL 算法在检测到概念漂移时，能够迅速提升模型的分类准确率。

从表 4 中的累积准确率可以看出，本文所提出的 CMOL 算法在所有数据集上能够表现出较高的准确率，这说明了本文采用的分类器池调整规则的有效性。为进一步研究本文所采用的指标的有效性，本文使用 RTG 中的一份数据进行跟踪观察，发现 CMOL 算法不会降低分类器池的“多样性”，而在使用 CMOL-I 算法和 CMOL-II 算法时，会导致分类器池的“多样性”降低。表 5－表 7 中的实验结果也表明了这一点。在表 5－表 7 中，每一列表示分类器池中分类器对应的概念序号。

从表 5 中可以看出，CMOL 在每次发生概念漂移时，替换掉权值最大的历史分类器，能够使得分类器池中尽可能多地包含不同的概念。从表 6 中可以看出，CMOL-I 在每次发生概念漂移时替换掉最老的历史分类器，不能够确保分类器池中尽可能多地包含不同的概念。从表 7 中可以看出，CMOL-II 采取的分类器池调整规则，会导致一个概念所对应的分类器会在分类器池中多次出现，无法保证分类器池中包含尽可能多的概念，说明权值并不能准确代表分类器所对应的概念之间的关系。若两个分类器的权值最接近，且权值较大者不是分类器池中的权值最大者，此时，如果新到概念为重现概念，那么分类器池中权值最大的分类器所对应的概念与新到概念应为同一概念。按照 CMOL-II 的分类器池调整规则，使用新到概念分类器替换掉权值最接近的两个分类器中权值较小的分类器，会导致分类器池中至少有两个分类器均对应于新到概念，从而使分类器池中所包含的概念数量减少。比如从 3400－3600 到 3600－4200，3600－4200 到 4200－4800，5000－5800 到 5800－6000，均满足以上条件，此时若采用 CMOL-II 的分类器池调整规则，分类器池存储的概念数量则会减少。

表 5　CMOL 算法运行时分类器池中的概念变化情况

Table 5　Changes of concept in classifier pools during CMOL algorithm running

3400－3600	3600－4200	4200－4800	4800－5000	5000－5800	5800－6000	6000－6600	6600－6800	6800－7400	7400－7600	7600－8200	8200－9000	9000－
1	2	2	3	4	4	4	1	2	2	3	4	4
2	3	3	4	4	4	1	2	3	3	4	4	4
3	4	4	4	1	1	2	3	4	4	4	1	1
4	4	4	1	2	2	3	4	4	4	1	2	2
4	1	1	2	3	3	4	4	1	1	2	2	3

表 6 CMOL-I 运行时分类器池中的概念变化情况

Table 6 Changes of concept in classifier pools during CMOL-I algorithm running

3400—3600	3600—4200	4200—4800	4800—5000	5000—5800	5800—6000	6000—6600	6600—6800	6800—7400	7400—7600	7600—8200	8200—9000	9000—
1	2	3	4	4	1	1	2	3	3	4	4	1
2	3	4	4	1	1	2	3	3	4	4	1	1
3	4	4	1	1	2	3	3	4	4	1	1	2
4	4	1	1	2	3	3	4	4	1	1	2	2
4	1	1	2	3	3	4	4	1	1	2	2	3

表 7 CMOL-II 运行时分类器池中的概念变化情况

Table 7 Changes of concept in classifier pools during CMOL-II algorithm running

3400—3600	3600—4200	4200—4800	4800—5000	5000—5800	5800—6000	6000—6600	6600—6800	6800—7400	7400—7600	7600—8200	8200—9000	9000—
1	1	1	4	4	1	1	1	1	1	1	1	1
2	3	4	4	1	1	1	1	1	1	1	1	1
3	4	4	1	1	2	2	2	4	4	4	1	1
4	4	1	1	2	3	3	4	4	1	1	2	2
4	1	1	2	3	3	4	4	1	1	2	2	3

为了更直观地理解，对上述情形进行举例说明，具体的例子如图 7—图 8 所示。调整前分类器池中各分类器已经按照权值从大到小排列，如$\{C_1,\cdots,C_i,C_j,\cdots,C_m\}$，分类器池中每个分类器均对应于不同的概念。同时，设 C_i 与 C_j 是分类器池中权值差异最小的两个分类器。

分类器池

$\{C_1,\cdots,C_i,C_n,\cdots,C_m\}$

图 7 采用 CMOL-II 调整后分类器池中的分类器

Fig. 7 Classifiers in classifier pool adjusted by CMOL-II

分类器池

$\{C_n,\cdots,C_i,C_j,\cdots,C_m\}$

图 8 采用 CMOL 调整后分类器池中的分类器

Fig. 8 Classifiers in classifier pool adjusted by CMOL

图 7 中，C_n 是新到概念对应的分类器。设新到概念为重现概念，则 C_1 与 C_n 是对应于同一概念的两个分类器。CMOL-II 在调整分类器池时，使用分类器 C_n 替换 C_j，替换后分类器池中存储的分类器如图 7 所示。图 7 中，由于 C_1 与 C_n 对应于同一概念，因此此时分类器池包含的概念数量减 1。CMOL 在调整分类器池时，使用分类器 C_n 替换 C_1，替换后分类器池中存储的分类器如图 8 所示。图 8 中，分类器池中的所有分类器均对应于不同的概念，因此此时分类器池包含的概念数量没有发生变化。因此，与 CMOL 相比，CMOL-II 所采取的分类器池调整规则不能够确保分类器池中尽可能多地包含不同的概念。

3.3 分类器池大小对 CMOL 算法的影响

为了验证分类器池的大小对 CMOL 算法的影响，本文将分类器池的大小依次设置为 3，5，7，9，将 Letter 数据集的分类器池的大小依次设置为 3，4，5，6，7。实验结果表明，不管分类器池的大小如何变化，CMOL 算法可以在多数情况下达到最高的准确率，说明了 CMOL 所采用的分类池调整策略的有效性。

从图 9 和图 10 中可以看出，各种算法随着分类器池变大，分类准确率逐渐提升。这可能是因为数据块不能够很好地表达其原本的数据分布，所以当分类器池中分类器数量不断增加时，分类器池中总能加入新的信息，故此时算法的准确率随着分类器池规模的变大逐渐提升。从图 11—图 14 中可以看出，当分类器池的大小增大到一定程度时，各种算法的分类准确率不再发生变化。这可能是因为数据块能够很好地表达其原本的数据分布，当分类器池达到一定的规模时，即使分类器池中分类器的数量继续不断增加，分类器池中也无法加入新的信息。因此，当分类器池的规模达到一定程度时，再增加分类器池的规模，分类准确率也无法得到提升。

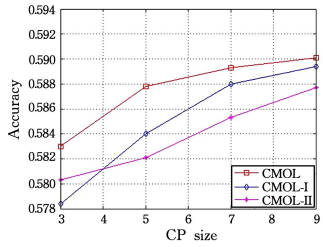

图 9 RTG 数据集上分类器池大小对分类准确率的影响

Fig. 9 Influence of classifier pool size on classification accuracy on RTG dataset

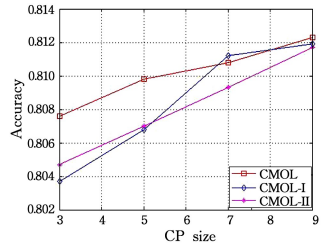

图 10 waveform 数据集上分类器池大小对分类准确率的影响

Fig. 10 Influence of classifier pool size on classification accuracy on waveform dataset

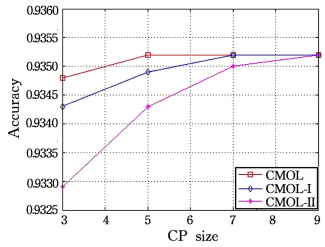

图 11 SEA 数据集上分类器池大小对分类准确率的影响

Fig. 11 Influence of classifier pool size on classification accuracy on SEA dataset

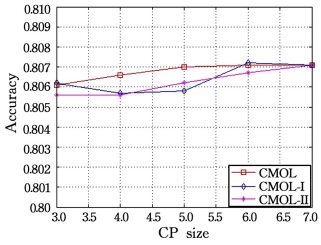

图 12　ABCD 数据集上分类器池大小对分类准确率的影响

Fig. 12　Influence of classifier pool size on classification accuracy on ABCD dataset

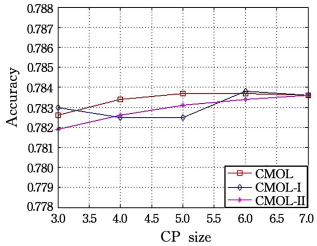

图 13　EFGH 数据集上分类器池大小对分类准确率的影响

Fig. 13　Influence of classifier pool size on classification accuracy on EFGH dataset

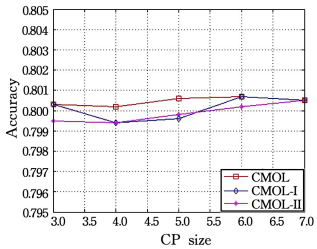

图 14　NOPQ 数据集上分类器池大小对分类准确率的影响

Fig. 14　Influence of classifier pool size on classification accuracy on NOPQ dataset

为了进一步分析是否是由于数据块的表达能力对分类准确率产生了影响，本文使用 RTG，waveform 和 SEA 数据集在不同大小的数据块上观察分类准确率的变化。实验结果如图 15－图 17 所示。

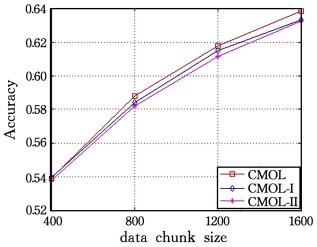

图 15　RTG 数据集上数据块大小对分类准确率的影响

Fig. 15　Influence of data block size on classification accuracy on RTG dataset

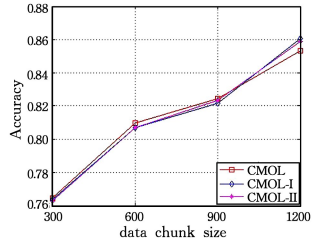

图 16　waveform 数据集上数据块大小对分类准确率的影响

Fig. 16　Influence of data block size on classification accuracy on waveform dataset

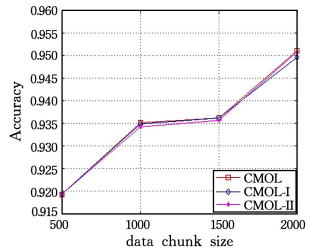

图 17　SEA 数据集上数据块大小对分类准确率的影响

Fig. 17　Influence of data block size on classification accuracy on SEA dataset

从图 15 可以看到，分类准确率的变化为 54%→58%→62%→64%；从图 16 可以看到分类准确率的变化为 76%→81%→82%→86%。也就是说，随着数据块的增大，各算法的分类准确率有着较大幅度的上升，分别从 54%提升到 64%左右以及从 76%提升到 86%左右，这说明数据块较大时，数据块能较好地表示数据原本的分布。而从图 17 可以看到，分类准确率的变化为 92%→93.5%→93.5%→95%，这说明数据块的增大只带来了分类准确率的微小提升，这就说明数据块能较好地表示数据原本的分布。因此，不同的数据集上，数据块的大小对分类准确率的影响不一样，从而验证了分类器池的大小对 CMOL 算法分类准确率的影响。

结束语　本文针对现有的概念漂移分类算法在遭遇概念漂移后分类器的分类准确率恢复较慢的不足，结合多源在线迁移学习提出了一种能够使用多个历史分类器实施多源迁移学习的数据流分类算法——CMOL。CMOL 算法根据存储的每个历史分类器对新到概念样本的分类性能动态调节每个历史分类器对应的权重，在检测到概念漂移时按照一定的规则更新分类器池，使得分类器池能尽可能包含更多的概念。实验结果进一步表明，与基于单源迁移的 CDOL 算法相比，CMOL 能够有效地使用多个历史分类器辅助新到概念进行学习，提升了在新到概念上的即时准确率。由于概念漂移检测的滞后性，每个分类器通常不是由属于同一个概念的样本训练得到的，这将导致分类准确率的下降，本文下一步将研究此问题，以减少概念漂移检测的滞后性产生的影响。

参 考 文 献

[1]　SCHLIMMER J C，GRANGER R H. Incremental Learning from Noisy Data[J]. Machine Learning，1986，1(3)：317-354.

[2]　HULTEN G，SPENCER L，DOMINGOS P. Mining time-changing data streams[C]// Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York：ACM，2001：97-106.

[3]　KOLTER J Z，MALOOF M A. Dynamic weighted majority：a new ensemble method for tracking concept drift[C]// Proceedings of the IEEE Conference on Data Mining. Piscataway：IEEE，2003：123-130.

[4]　JR P M G，BARROS R S M D. RCD：A recurring concept drift framework[J]. Pattern Recognition Letters，2013，34(9)：1018-1025.

[5]　LI P，WU X，HU X. Mining recurring concept drifts with limited labeled streaming data[C]// Proceedings of the 2nd Asian Con-

ference on Machine Learning. New York: ACM, 2010: 241-252.

[6] ZHAO P, HOI S C H, WANG J, et al. Online Transfer Learning [J]. Journal of Artificial Intelligence, 2014, 216(16): 76-102.

[7] WEN Y M, TANG S Q, FENG C, et al. Online Transfer Learning for Mining Recurring Concept in Data Stream Classification [J]. Journal of Computer Research and Development, 2016, 53(8): 1781-1791. (in Chinese)
文益民, 唐诗淇, 冯超, 等. 基于在线迁移学习的重现概念漂移数据流分类[J]. 计算机研究与发展, 2016, 53(8): 1781-1791.

[8] WEN Y M, QIANG B H, FAN Z G. A survey of the classification of data streams with concept drift[J]. CAAI Transactions on Intelligent Systems, 2013, 8(2): 95-104. (in Chinese)
文益民, 强保华, 范志刚. 概念漂移数据流分类研究综述[J]. 智能系统学报, 2013, 8(2): 95-104.

[9] ZLIOBAITE I, PECHENIZKIY M, GAMA J. An overview of concept drift applications[J]. Studies in Big Data, 2016, 16(1): 91-114.

[10] KRAWCZYK B, MINKU L L, GAMA J, et al. Ensemble learning for data stream analysis: A survey[J]. Information Fusion, 2017, 37(C): 132-156.

[11] GAMA J, ZLIOBAITE I, BIFET A, et al. A survey on concept drift adaptation[J]. ACM Computing Surveys (CSUR), 2014, 46(4): 1-37.

[12] CASTILLO G, GAMA J, BREDA A M. Adaptive bayes for a student modeling prediction task based on learning styles[C]// Proceedings of the International Conference on User Modeling. BerLin: Springer, 2003: 328-332.

[13] KUKAR M. Drifting Concepts as Hidden Factors in Clinical Studies[M]// Artificial Intelligence in Medicine. Berlin: Springer, 2003: 28-35.

[14] ZHUANG F Z, LUO P, HE Q, et al. Survey on transfer learning research[J]. Journal of Software, 2015, 26(1): 26-39. (in Chinese)
庄福振, 罗平, 何清, 等. 迁移学习研究进展[J]. 软件学报, 2015, 26(1): 26-39.

[15] LU L L, ZHANG Y P, TAN H Y, et al. Research on classification algorithm and concept drift based on big data[J]. Journal of Frontiers of Computer Science & Technology, 2016, 10(12): 1683-1692. (in Chinese)
陆莉莉, 张永潘, 谈海宇, 等. 大数据分类挖掘算法及其概念漂移应用研究[J]. 计算机科学与探索, 2016, 10(12): 1683-1692.

[16] LI Y, ZHANG Y H, HU X G, et al. Classification Algorithm for Data Stream Based on Mixture Models of C4.5 and NB[J]. Computer Science, 2010, 37(12): 138-142. (in Chinese)
李燕, 张玉红, 胡学钢, 等. 基于 C4.5 和 NB 混合模型的数据流分类算法[J]. 计算机科学, 2010, 37(12): 138-142.

[17] VINAYAGA SUNDARAM B, AARIHI R J, SARANYA P A. Efficient Gaussian Decision Tree method for Concept drift data stream[C]// Proceedings of the International Conference on Signal Processing, Communication and Networking. Piscataway: IEEE, 2015: 1-5.

[18] STREET W N. A streaming ensemble algorithm (SEA) for large-scale classification[C]// Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York: ACM, 2001: 377-382.

[19] RAMAMURTHY S, BHATNAGAR R. Tracking Recurrent Concept Drift in Streaming Data Using Ensemble Classifiers[C]// Proceedings of the International Conference on Machine Learning and Applications. IEEE: NJ, 2007: 404-409.

[20] BRZEZINSKI D, STEFANOWSKI J. Reacting to different types of concept drift: The Accuracy Updated Ensemble algorithm[J]. IEEE Transactions on Neural Networks & Learning Systems, 2014, 25(1): 81-94.

[21] SUN Y, TANG K, ZHU Z, et al. Concept Drift Adaptation by Exploiting Historical Knowledge [J]. IEEE Transactions on Neural Networks & Learning Systems, 2017, PP(99): 1-11.

[22] XIN Y, GUO G D, CHEN L F, et al. IKnnM-DHecoc: A Method for Handling the Problem of Concept Drift[J]. Journal of Computer Research and Development, 2011, 48(4): 592-601. (in Chinese)
辛轶, 郭躬德, 陈黎飞, 等. IKnnM-DHecoc: 一种解决概念漂移问题的方法[J]. 计算机研究与发展, 2011, 48(4): 592-601.

[23] WEISS K, KHOSHGOFTAAR T M, WANG D D. A survey of transfer learning[J]. Journal of Big Data, 2016, 3(1): 9.

[24] SUN S, SHI H, WU Y. A survey of multi-source domain adaptation[J]. Journal of Information Fusion, 2015, 24(C): 84-92.

[25] PAN S J, YANG Q. A Survey on Transfer Learning[J]. IEEE Transactions on Knowledge And Data Engineering, 2010, 22(10): 1345-1359.

[26] WU Q, WU H, ZHOU X, et al. Online transfer learning with multiple homogeneous or heterogeneous sources [J]. IEEE Transactions on Knowledge and Data Engineering, 2017, 29(7): 1494-1507.

[27] TANG S Q, WEN Y M, QIN Y X, et al. Online Transfer Learning from Multiple Sources Based on Local Classification Accuracy[J]. Journal of Software, 2017, 28(11): 2940-2960. (in Chinese)
唐诗淇, 文益民, 秦一休, 等. 一种基于局部分类精度的多源在线迁移学习算法[J]. 软件学报, 2017, 28(11): 2940-2960.

[28] BIFET A, HOLMES G, KIRKBY R, et al. MOA: Massive Online Analysis[J]. Journal of Machine Learning Research, 2010, 11(2): 1601-1604.