一种面向不完全标记的文本数据流自适应分类方法

张玉红 陈 伟 胡学钢
(合肥工业大学计算机与信息学院 合肥 230009)

摘 要 现实生活中网络监控、网络评论以及微博等应用领域涌现了大量文本数据流,这些数据的不完全标记和频繁概念漂移给已有的数据流分类方法带来了挑战。为此,面向不完全标记的文本数据流提出了一种自适应的数据流分类算法。该算法以一个标记数据块作为起始数据块,对未标记数据块首先提取标记数据块与未标记数据块之间的特征集,并利用特征在两个数据块间的相似度进行概念漂移检测,最后计算未标记数据中特征的极性并对数据进行预测。实验表明了算法在分类精度上的优越性,尤其在标记信息较少和概念漂移较为频繁时。

关键词 不完全标记,自适应,数据流,概念漂移
中图法分类号 TP181 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.12.032

Self-adaptation Classification for Incomplete Labeled Text Data Stream

ZHANG Yu-hong CHEN Wei HU Xue-gang
(School of Computer and Information, Hefei University of Technology, Hefei 230009, China)

Abstract In the real-world applications, a large number of text data stream are emerging, such as network monitoring, network comments and microblogs. However, these data have incomplete labels and frequent concept drifts, which have brought many challenges to existing classification methods of data stream. Thus we proposed a self-adaptation classification algorithm for incomplete labeled text data stream in this paper. The proposed algorithm uses a labeled data chunk as the starting one, and extracts features between the labeled data chunk and the unlabeled data chunk. Meanwhile, for unlabeled data chunks, it uses the similarity of features between two data chunks to test concept drift. Finally, the polarity of features of the unlabeled data chunks is calculated to predict the instances. The experimental results show our algorithm can improve the classification accuracy, especially in the data cases with less label information and more concepts drifts.

Keywords Incomplete labeled, Self-adaptation, Data stream, Concept drift

1 引言

数据流是网络监控、检测网络评论及微博等应用领域常见的数据形态,具有数据量大、价值密度低、异构和概念漂移等特点,已成为了研究热点。针对标记数据流分类方法,学者们提出了很多的算法,如 VFDT[1](Very Fast Decision Tree),UFFT[2](Ultra Fast Forest Tree)和数据流集成分类方法[3]等。

然而,相对于数据获取的便利性和廉价性,标记信息的获取却十分昂贵,学者们对部分标记的数据流开展了研究。文献[4]对有标记示例集进行可重复取样(bootstrap sampling)以获得3个有标记训练集并分别产生一个分类器,分类器获得的新标记示例由其余两个分类器协作提供,算法不需要冗余的数据集,在批处理方面具有很好的应用。文献[5]提出了一种集成分类器和聚类算法,该算法对标记数据构建分类器,对未标记数据构建聚类簇,并结合分类器和聚类集成新的集成模型;该算法利用聚类簇信息提高了预测精度,且算法效果受到所含标签信息比例的影响。

实际数据流应用中,标签信息的缺失和概念漂移往往是同时存在的,已有算法大多利用标签信息进行概念漂移检测,当概念漂移相对频繁时,可获取的标签信息则相对较少,从而影响算法的效果。为此,本文针对不完全标记且概念漂移相对较为频繁的文本数据流,提出了一种自适应数据流分类算法。该算法以一个标记数据块作为起始数据块,若当前为标记数据块,则直接更新分类器,否则利用上一个标记数据块和当前未标记数据块之间特征的相似度进行概念漂移的判断并对未标记数据块进行预测。

2 相关工作

本节主要介绍经典的标记数据流分类方法和不完全标记数据流分类方法。

到稿日期:2015-10-10 返修日期:2016-01-07 本文受教育部创新团队(IRT13059),国家自然科学基金(61305063,61273292),博士点项目基金(20130111110011)资助。

张玉红(1979—),女,副教授,主要研究方向为数据挖掘、机器学习,E-mail:zhangyh@hfut.edu.cn;陈 伟(1990—),硕士生,主要研究方向为数据流分类、迁移学习,E-mail:1020071601@qq.com;胡学钢(1961—),男,教授,博士生导师,主要研究方向为数据挖掘、智能计算,E-mail:jsjxhuxg@hfut.edu.cn。

2.1 标记数据流分类方法

标记数据流分类方法多基于增量式分类模型，其代表算法有 ASHoeffdingTree[6]，CVFDT[7]，UFFT[2] 和数据流集成分类算法[3]。

VFDT 是一种基于 Hoeffding 不等式建立决策树的算法，它通过不断地将叶节点替换为决策节点生成。CVFDT 在 VFDT 的基础上增加了通过扫描决策树内部节点的错误率来检测概念漂移的过程。文献[8]在 VFDT 的基础上，修改 Hoeffding 不等式，使用一种新的基于分类错误率的方法来建立决策树。UFFT 算法是基于二叉决策树集成模型的增量式算法，即按照每对数据流类别可能的组合构建二叉决策树。UFFTE[9]是基于 UFFT 模型的增量式算法，其不同点在于决策树在每个决策节点都构建一个 Naïve Bayes 分类器，利用分类器的错误率来检测漂移。文献[10]介绍了一种基于实例学习的数据流分类算法，该算法首先利用原始的有标签数据建立分类器，然后保留一个隐含的概念描述，通过加入有用的实例并删除无用的实例来更新分类器，利用 KNN 算法对新到来的实例进行预测。文献[11]介绍了一种基于序列正则化(Sequential regularization)机制的概念漂移检测方法，该算法从叶子节点回溯当前遍历的决策树。在当前决策路径中，对比每个节点的数据分布与派生的叶子节点整体分布，如果检测到分布变化，叶子节点的统计信息被回溯到发生漂移的节点，且剪枝节点所在的子树。

上述算法要求数据流是完全标记的，不适用于标签大量缺失的数据流。

2.2 不完全标记数据流分类方法

文献[4]是面向不完全标记数据流的一种半监督算法，通过对标记数据集进行可重复取样获得 3 个有标记训练集，并分别构建分类器，然后利用其中两个分类器的协作为另一个分类器提供新标记示例，最后基于 3 个分类器的集成对未标记示例进行预测。文献[5]结合集成分类和聚类方法，提出一种不完全标记数据流的分类算法，其对标记数据构建分类器，对未标记数据用 k-means 构建聚类簇，再对已有的标记数据构建聚类簇，根据标记聚类簇与未标记聚类簇之间的簇相似度给未标记数据打上标签，并训练新分类器加入集成分类器中。文献[12]介绍了一种利用少量标签信息的不完全标记数据流分类算法，首选利用 k-means 对标记实例构建 k 个簇，然后计算出未标记实例与每个簇之间的距离，利用距离最近的簇的标签信息给未标记实例打上标签，同时利用最新的标记实例对分类器进行模型更新。以上算法并不直接进行概念漂移检测。文献[13]提出一种用于概念漂移学习的概念跟踪算法，当存在无标签示例时，采用增量式方式来构建概念层次结构以识别示例的类别。文献[14]采用主动学习方法选择对概念变化敏感的标记示例来训练分类器，所提算法只需少量的标记文档就能处理文本过滤环境下的概念漂移问题。文献[15]提出一种新的不完全标记数据流分类算法，将新类别检测机制融入到传统的分类器中，并设计了一个最大可允许等待时间 Tc 作为时间约束，实现了在数据流的真实类标签到来之前的新类别示例的自动检测。

上述算法在标签信息较多和概念漂移较少的数据流上具有一定的效果，而当数据流中含有的标签信息较少、概念漂移较为频繁时，其分类精度受到一定影响。

3 自适应在线数据流分类算法

本文面向不完全标记的文本数据流，提出了一种自适应在线数据流分类算法 SAOC(Self-Adaptation Online Classification)，该算法提取标记数据与未标记数据间的共有特征，并对未标记数据进行预测，在保证算法时间效率的情况下，提高了算法的精度。

3.1 问题定义

假设不完全标记数据流 S 共有 $m+n$ 个数据块，表示为 $(D_{L1}, D_{U1}, \cdots, D_{Li}, D_{Uj}, D_{Lm}, \cdots, D_{Un})$，其中 D_{L_i} 表示标记数据块，D_{Uj} 表示未标记数据块。一般情况下，$m \ll n$。本文用到的变量的表示及其意义说明如表 1 所列，算法流程图如图 1 所示。

表 1 变量的表示及意义

字母名称	代表的意思
D_{Li}	第 i 个标记数据块
D_{Uj}	第 j 个未标记数据块
$F=\{f_k\}$	D_{Li} 和 D_{Uj} 共有特征集合 f_k 表示集合中的第 k 个特征
P_F^L, P_F^U	F 在 D_{Li} 和 D_{Uj} 上的极性
F_L, F_U	D_{Li}, D_{Uj} 的专有特征集合
P_L, P_U	F_L, F_U 的极性
$d_{f_k}^L, d_{f_k}^U$	f_k 在 D_{Li}, D_{Uj} 上的方向因子
M_F^L, M_F^U	F 在 D_{Li}, D_{Uj} 上的共现矩阵
M_F^{FL}, M_F^{FU}	F_L, F_U 与 F 的共现矩阵

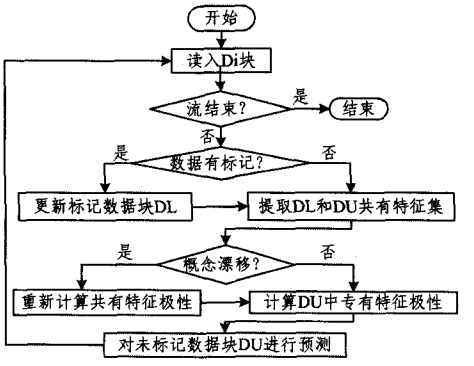


图 1 算法流程图

算法主要包括 3 个步骤：1)基于滑动窗口机制将数据读入；2)若当前为标记数据块，则更新标记数据块 D_{Li}，否则首先提取 D_{Li} 和 D_{Uj} 的共有特征集，再进行概念漂移检测及 D_{Uj} 中共有特征极性的计算，最后对 D_{Uj} 中的专有特征极性进行计算；3)利用 D_{Uj} 中的特征极性对未标记块进行预测并返回至步骤 1)，直到数据流结束。用最新的标记数据块与未标记数据块提取共有特征集。步骤 2)为算法的核心步骤，下面将进行重点介绍。

3.2 共有特征提取

本节的目的是构建两个数据块 D_{Li} 和 D_{Uj} 之间的共有特征集，选取两个数据块中具有较高极性的公共特征作为共有特征。

对于标记数据块 D_{Li}，可利用经典的策略如 MI(Mutual Information)或 OR(Odds Ratio)来选择极性高的特征，但这些策略不适用于未标记数据块 D_{Uj}，而基于词频选取特征的方法[16,17]则容易选择出一些极性较低或对分类无意义的特征。

本文首先利用 MI 或 OR 选择标记数据块 D_{Li} 中区分力高的特征，记为 $F_S=\{f_*\}$，然后构建 F_S 在 D_{Uj} 上的特征共现矩阵 $M_{F_S}^U=\{m_{k,t}\}$。假设与正向特征共现频率高而与负向特征共现频率低的特征具有较高的正向极性；反之亦然。选择

在 D_{Li} 和 D_{Uj} 中都具有较高极性的特征作为共有特征集合 F，其余特征分别作为专有特征集合，记为 F_L 和 F_U。F 的具体选取方法如式(1)所示：

$$F=\{f_k \mid\mid \sum_{p_{f_{sk}}\geqslant 0} m_{k,l}-\sum_{p_{f_{sk}}<0} m_{k,l}\mid>\varepsilon\} \tag{1}$$

其中，$m_{k,l}$ 表示 D_{Uj} 中的特征与 F_S 的共现次数，$p_{f_{sk}}\geqslant 0$ 表示 f_{sk} 的极性为正，反之为负。式(1)中特征 f_{sk} 与正向特征共现次数和与负向特征共现次数的差值越大，表示特征极性越强。

3.3 未标记块共有特征极性的计算

本节的目的是计算共有特征集 F 在未标记数据块 D_{Uj} 中的极性。尽管 F 在标记数据块 D_{Li} 中的极性是已知的，但是在发生概念漂移的情况下，这些极性不适用于 D_{Uj}。为此首先需要判断是否发生概念漂移。在不发生概念漂移时令 $P_F^U=P_F^L$，而概念漂移时需重新计算 P_F^U。

根据共有特征中相似特征的比例来判断是否漂移。引入相似因子以便更好地表示，如式(2)所示：

$$f_{ad}=\frac{\sum_{1}^{|F|} d_{f_k}^L \oplus d_{f_k}^U}{|F|} \tag{2}$$

其中，$d_{f_k}^L=\sum_{p_{f_l}\geqslant 0} m_{k,l}-\sum_{p_{f_l}<0} m_{k,l}$ 表示 f_k 在 D_{Li} 中的方向因子，$p_{f_l}\geqslant 0$ 表示 F 中的正向特征集合。$d_{f_k}^L\oplus d_{f_k}^U$ 表示特征 f_k 在 D_{Li} 和 D_{Uj} 中的方向因子的异构值，若该值为+1，表示特征 f_k 在 D_{Li} 和 D_{Uj} 中相似；若该值为−1，则表示特征 f_k 在 D_{Li} 和 D_{Uj} 中不相似。f_{ad} 表示 F 中相似特征在共有特征中所占的比例，当 f_{ad} 大于阈值 α，则认为标记数据块与非标记数据块的特征分布不相似，即发生概念漂移，否则没有发生漂移，如式(3)所示：

$$P_F^U=\begin{cases} P_F^L, & f_{ad}\leqslant\alpha \\ P_F^L * R, & f_{ad}>\alpha \end{cases} \tag{3}$$

当发生概念漂移时，通过计算 F 在标记数据块和未标记数据块中的分布距离来计算其极性。R 是一个分布距离矩阵：$R=\{r_{k,l}\}$，如式(4)所示：

$$r_{k,l}=(d_{f_k}^L\oplus d_{f_l}^U) * KL(M_{f_k}^L \mid M_{f_l}^U) \tag{4}$$

$KL(M_{f_k}^L \mid M_{f_l}^U)$ 表示 $f_k(f_k\in D_{Li})$ 和 $f_l(f_l\in D_{Uj})$ 的 KL (Kullback-Leibler)距离[18]，如式(5)所示：

$$KL(M_{f_k}^L \mid M_{f_l}^U)=\sum_{f_i}\sum_{p_{f_i}\geqslant 0} m_{f_k,f_i}^U * \log\frac{\sum_{p_{f_i}\geqslant 0} m_{f_k,f_i}^U}{\sum_{p_{f_i}\geqslant 0} m_{f_k,f_i}^L} \tag{5}$$

其中，$\sum_{p_{f_i}\geqslant 0} m_{f_k,f_i}^L$ 表示 F 中的特征 f_k 与 D_{Li} 中所有正向特征 f_i 的共现概率。

3.4 未标记数据专有特征的极性计算

本节目的是计算未标记数据块中专有特征的极性 P_U，已知的信息有 P_L 和 D_{Uj} 中的 P_F^U。可以利用 D_{Uj} 中专有特征与共有特征的共现关系，使与之共现的专有特征极性 P_L 从 D_{Li} 传递到 D_{Uj} 中，来共同计算 D_{Uj} 中专有特征的极性 P_U。

例如，在 D_{Li} 中的专有特征 read 与共有特征 good 共现，而在 D_{Uj} 中 good 又与专有特征 sharp 共现，因此利用共有特征 good 作为桥梁，可将 read 的极性从 D_{Li} 传递到 D_{Uj} 中的 sharp 上。另一方面，在 D_{Uj} 中，sharp 可直接利用 good 在 D_{Uj} 中的极性 P_F^U。

未标记数据块 D_{Uj} 中专有特征极性的具体表示方法如式(6)所示：

$$P_U=\beta P_F^U * M_F^{FU}+(1-\beta)P_L * M_F^{FL} \tag{6}$$

式(6)中有两项，第一项表示利用未标记数据块 D_{Uj} 自身的数据分布信息，第二项表示利用标记数据块 D_{Li} 中专有特征的信息，β 为权重参数，其取值范围为[0,1]。

需要说明的是，本文中始终使用最近的标签数据块对未标签数据块 D_{Li+1} 进行预测。

4 实验结果与分析

本节首先给出实验数据集；其次给出对比算法及参数设置；最后对比标记信息对算法精度的影响、算法对概念漂移的适应性及算法的时间效率。

4.1 实验数据集和对比算法

共采用了两个数据集，RevE[19]和 RevC。RevE 是英文亚马逊产品评论数据，包括 Books(B)，Dvd(D)，Electronics(E)和 Kitchen(K) 4 个领域。RevC 是作者从中文亚马逊爬取的数据，包括老人机(O)、相机(C)、智能机(I)、电脑(N)、酒店(H) 5 个领域。两个数据集中的评论基于用户的评价分数标记为−1(负向评论)或+1(正向评论)。

采用随机抽取标记块和未标记块的方式来模拟数据流，每个块包含 750 个正向评论和 750 个负向评论。由于抽取的随机性，数据流中的概念漂移次数相对频繁。

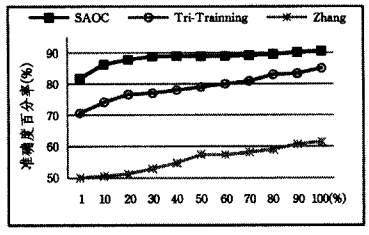
将所提算法与完全标记数据流分类算法及不完全标记数据流分类算法进行了对比。完全标记数据流分类算法有 VFDT[1]和 ASHoeffdingTree[6]，不完全标记数据流分类算法有 Tri-Training[4]和 Zhang[5]。

4.2 参数设置

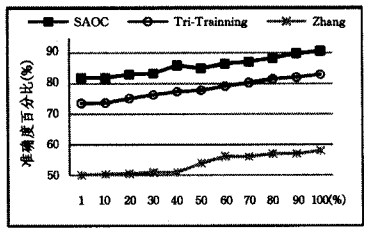
在进行数据预处理时，对 RevE 把所有词转换为小写，并且去除停用词。另外，对 RevE 和 RevC 还过滤掉文档词频小于 3 的特征。在标记数据块中，特征按极性大小降序排列，选择正负绝对值大的前 2000 个特征。式(1)中的参数 ε 设置为 2，式(3)中的参数 α 设置为 0.06，式(6)中的参数 β 设置为 0.5，标记数据块的滑动窗口块大小设置为 1500，共有特征集 F 的大小设置为 150。

4.3 标记信息对算法精度的影响

本节主要考察标记数据信息的含量对算法精度的影响，由于 VFDT 和 ASHoeffdingTree 只适用于完全标记的数据，因此本节只将 SAOC 与不完全标记数据流算法进行对比，结果如图 2 所示。



(a)RevC 数据集



(b)RevE 数据集

图 2 算法随标记信息比例变化的曲线图

由图 2 可知，所提算法的精度要高于 Tri-Training 和 Zhang 的。在数据集 RevC 上，与 Tri-Training 和 Zhang 相比，SAOC 的平均精度分别提高了 10%和 33%；当标记信息比例为[1%，40%]时，SAOC 的平均精度分别提高了 13%和 35%；当标记信息比例为[50%，100%]时，SAOC 的平均精度分别提高了 7%和 30%。由此可见，SAOC 在标记信息大量缺失的情况下较其他算法具有一定的优势。分析其原因如下：Tri-Training 和 Zhang 没有足够的标签信息来重新训练分类器，而原来的分类器由于发生概念漂移而不适应当前的数据块，从而导致其分类精度降低；而 SAOC 在发生概念漂移时，能利用原有标记数据块中适应部分的标签信息和不适应部分的标签信息（即共有特征和专有特征两部分的标签信息）进行预测，因此具有的较好的分类效果。

4.4 算法对概念漂移的适应性

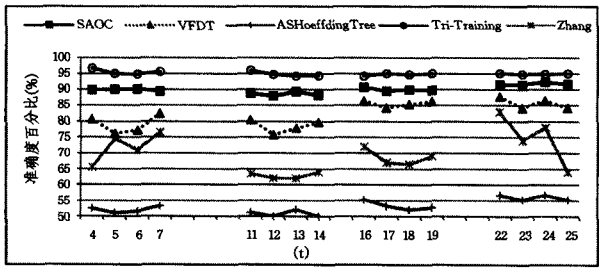
本节主要考察当数据发生概念漂移时算法的适应性。

对未标记信息为 50%的数据流算法在各时间点的精度统计如表 2 所列。表 2 中，RevE 数据集没有发生概念漂移时，Tri-Training 效果最好，平均精度为 95.41%，SAOC 次于 Tri-Training，平均精度为 90.04%。图 3(a)中，在 t11－t14 时间段，数据没有发生概念漂移，此时 Tri-Training 的平均精度为 94.8%，SAOC 的平均精度为 88.57%。Tri-Training 的平均精度最高的原因是在训练分类器时，当数据分布相似时，所使用的集成分类器将新的文本加入到分类器中，必然会使分类模型中标记信息涵盖越来越全，促使精度大幅度提高。

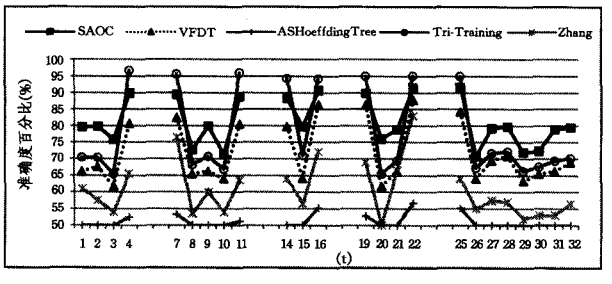
表 2　RevC 和 RevE 数据集上各时间点下各算法的平均精度

漂移时间点		各时间点平均精度(%)				
RevC	t1－t3，t8，t9，t14－t16，t21－t26	SAOC 82.87	VFDT 70.92	ASHoeffdingTree 54.04	Tri-Trianing 72.95	Zhang 51.1
RevE	t1－t3，t8－t10，t15，t20，t21，t26－t32	77	66.89	50	70.93	51
未漂移的时间点						
RevC	t4－t7，t10－t13，t17－t20	90.32	77.17	54.72	92.77	69.6
RevE	t4－t7，t11－t14，t16－t19，t22－t25	90.04	82.27	52.99	95.41	66.1

图 3(b)中，t8 时间点发生概念漂移时，SAOC 的精度为 72.63%，Tri-Training 的精度为 68.15%。当数据在时间点 t26－t32 间持续发生概念漂移时，SAOC 和 Tri-Training 的平均精度分别为 76.03%和 69.2%，由此可见，在数据持续发生概念漂移时 SAOC 的适应性要优于其余算法。表 2 中的结果同样表明了 SAOC 在频繁概念漂移时的优越性。分析其原因如下：当发生频繁概念漂移时，Tri-Training 和 Zhang 难以同时适用于多个概念，而 SAOC 利用同一标记数据块，分别对不同数据块的共有特征极性进行动态调整，从而能对频繁概念漂移具有较好的适应性。



(a)RevE-未漂移时间点



(b)RevE-漂移时间点

图 3　算法在 RevE 上漂移时间点和未漂移时间点的精度变化曲线

结束语　针对不完全标记且包含频繁概念漂移的文本数据流分类问题，本文提出了一种自适应分类算法，以一个标记数据块作为起始数据块，利用标记数据块和未标记数据块之间特征的相似度进行概念漂移的判断并对未标记数据块进行预测。实验表明在标签大量缺失及概念漂移较为频繁的文本数据流上，所提算法的分类性能优于其他的基本算法。在不平衡数据流环境下如何有效地进行特征选择是下一步的研究重点。

参 考 文 献

[1] Domingos P, Hulten G. Mining high-speed data streams [C]// Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2000. New York, NY, USA: ACM, 2000: 71-80

[2] Gama J, Medas P, Rocha R. Forest Trees for On-line Data[C]// Proceedings of the 2004 ACM Symposium on Applied Computing, 2004. New York, NY, USA: ACM, 2004: 632-636

[3] Wang H, Fan W, Yu P S, et al. Mining concept-drifting data streams using ensemble classifiers [C] // Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2003. New York, NY, USA: ACM, 2003: 226-235

[4] Zhou Z H, Li M. Tri-training: Exploiting unlabeled data using three classifiers[J]. IEEE Transactions on Knowledge and Data Engineering, 2005, 17(11): 1529-1541

[5] Zhang P, Zhu X, Tan J, et al. Classifier and cluster ensembles for mining concept Drifting data streams[C]//Proceedings of IEEE International Conference on Data Mining, 2010. Washington, DC, USA: IEEE Computer Society, 2010: 1175-1180

[6] Hoeffding W. Probability inequalities for sums of bounded random variables[J]. Journal of the American Statistical Association, 1963, 58(301): 13-30

[7] Hulten G, Spencer L, Domingos P. Mining time-changing data streams[C]//Proceedings of the Seventh ACM SIGKDD Inter-

(下转第 194 页)

nications Conference (GLOBECOM 2010). IEEE, 2010: 1-5

[4] Cheng Xue-qi, Jin Xiao-long, Wang Yuan-zhuo, et al. Survey on Big Data System and Analytic Technology[J]. Journal of Software, 2014(9): 1889-1908(in Chinese)
程学旗,靳小龙,王元卓,等. 大数据系统和分析技术综述[J]. 软件学报,2014(9):1889-1908

[5] Meng Xiao-feng, Li Yong, Zhu Jian-hua, et al. Social Computing in the Era of Big Data: Opportunities and Challenges[J]. Journal of Computer Research and Development, 2013, 50(12): 2483-2491(in Chinese)
孟小峰,李勇,祝建华,等. 社会计算:大数据时代的机遇与挑战[J]. 计算机研究与发展,2013,50(12):2483-2491

[6] Shi Jin-gang, Bao Yu-bin, Leng Fang-ling, et al. Study on Log-Based Change Data Capture and Handling Mechanism in Real-Time Data Warehouse[C]// Proceedings of 2008 International Conference on Computer Sience and Software Engineering. Wuhan, 2008: 478-481

[7] Li Shi-jun, Yu Jun-qing, Ou Wei-jie. Web Information Extraction Based on HTML Pattern Algebra[J]. Journal of Computer Research and Development, 2006, 43(9): 1644-1650(in Chinese)
李石君,于俊清,欧伟杰. 基于 HTML 模式代数的 Web 信息提取方法[J]. 计算机研究与发展,2006,43(9):1644-1650

[8] Building the Data Warehouse [M]. New York: John Wiley & Sons, 1996

[9] Korn F, Muthukrishnan S, Zhu Y. Checks and balances: Monitoring data quality problems in network traffic databases[C]// Proc of the 29th IntConf on Very Large Databases. San Francisco, USA, 2003: 536-547

[10] Xu K S, Kliger M, Hero A O I. Evolutionary spectral clustering with adaptive forgetting factor[C]// International Conference on Acoustics, Speech, and Signal Processing, 1988 (ICASSP-88). 2010: 2174-2177

[11] Wang Y, Liu S X, Feng J, et al. Mining Naturally Smooth Evolution of Clusters from Dynamic Data[C]// Proc. of SIAM Conf. on Data Mining. 2007: 125-134

[12] Li J, Li S. Evolutionary Hierarchical Dirichlet Process for Timeline Summarization[C]// Meeting of the Association for Computational Linguistic. 2013: 556-560

[13] Kim H D, Lee D H, Choe H, et al. The evolution of cluster network structure and firm growth: a study of industrial software clusters[J]. Scientometrics, 2014, 99(1): 77-95

[14] Hedeler C, Belhajjame K, Fernandes A A A, et al. Dimensions of Dataspaces[M]// Dataspace: The Final Frontier. Springer Berlin Heidelberg, 2009: 55-66

[15] Ci Xiang, Ma You-zhong, Meng Xiao-feng, et al. Method for Top-K Query on Big Data in Cloud[J]. Journal of Software, 2014, 25(4): 813-825(in Chinese)
慈祥,马友忠,孟小峰,等. 一种云环境下的大数据 Top-K 查询方法[J]. 软件学报,2014,25(4):813-825

[16] Peng Yuan-hao, PAN Jiu-hui. Study on Incremental Data Capturing Method Based on Log Analysis[J]. Computer Engineering, 2015, 6(6): 56-60(in Chinese)
彭远浩,潘久辉. 基于日志分析的增量数据捕获方法研究[J]. 计算机工程,2015,6(6):56-60

(上接第 182 页)

national Conference on Knowledge Discovery and Data Mining, 2001. New York, NY, USA: ACM, 2001: 97-106

[8] Rutkowski L, Jaworski M, Pietruczuk L, et al. A New Method for Data Stream Mining Based on the Misclassification Error [J]. IEEE Transactions on Neural Networks and Learning Systems, 2015, 26(5): 1048-1059

[9] Gama J. Learning Decision Trees from Dynamic Data Streams [J]. Journal of Universal Computer Science, 2005, 11(8): 1353-1366

[10] Mena Torres D, Aguilar Ruiz J S. A similarity-based approach for data stream classification[J]. Expert Systems with Applications, 2014, 41(9): 4224-4234

[11] Gama J, Fernandes R, Rocha R. Decision Trees for Mining Data Streams[J]. Intelligent Data Analysis, 2006, 10(1): 23-45

[12] Andromeda T, Marsono M N, Ru L H. Online Data Stream Learning and Classification with Limited Labels[C]// Proceeding of International Conference on Electrical Engineering, Computer Science and Informatics, 2014. Yogyakarta, Indonesia: Indonesia journals, 2014: 161-164

[13] Widyantoro D H. Exploiting Unlabeled Data in Concept Drift Learning[J]. Jurnal Informatika, 2007, 8(1): 54-62

[14] Lindstrom P, Delany S J, B M Namee. Handling Concept Drift in a Text Data Stream Constrained by High Labelling Cost[C]// Proceedings of the 23rd International Florida Artificial Intelligence Research Society Conference, 2010. Florida, USA: AAAI, 2010: 32-37

[15] Masud M M, Gao J, Khan L, et al. Classification and Novel Class Detection in Concept-Drifting Data Streams under Time Constraints[J]. IEEE Transactions on Knowledge and Data Engineering, 2011, 23(6): 859-874

[16] Xiao M, Guo Y. Semi-Supervised Kernel Matching for Domain Adaptation[C]// Proceedings of the 26th AAAI Conference on Artificial Intelligence, 2012. North America: AAAI, 2012: 1183-1189

[17] Kobayashi N, Inui K, Matsumoto Y. Extracting Aspect-Evaluation and Aspect-of Relations in Opinion Mining[C]// Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, 2007. Prague: Association for Computational Linguistics, 2007: 1065-1074

[18] Li L H, Jin X M, Long M S. Topic Correlation Analysis for Cross-Domain Text Classification[C]// Proceedings of the 26th AAAI Conference on Artificial Intelligence, 2012. North America: AAAI, 2012: 998-1004

[19] Blitzer J, McDonald R, Pereira F. Domain adaptation with structural correspondence learning[C]// Proceedings of the Conference on Empirical Methods in Natural Language, 2006. Stroudsburg, PA, USA: Association for Computational Linguistics, 2006: 120-128