基于流数据分析的 ATM 欺诈检测

Yelleti Vivek[1], Vadlamani Ravi[1*], Abhay Anand Mane[2], Laveti Ramesh Naidu[2]

[1]人工智能与机器学习中心 银行技术发展与研究院，Castle Hills 路 #1，Masab Tank，海得拉巴-500076，印度[2] 高级计算发展中心，#1 Old Madras 路，班加罗尔，卡纳塔克邦 560038，印度
yvivek@idrbt.ac.in ; vravi@idrbt.ac.in; abhaym@cdac.in; rameshl@cdac.in

摘要

赢得客户的信任与信心，是金融机构和组织发展与成功的核心。近年来，金融行业受到大量欺诈活动的显著影响。此外，由于生成了海量数据集，底层框架必须具备可扩展性并满足实时需求，这一点至关重要。为解决这一问题，本研究分别提出了静态和流式环境下的 ATM 欺诈检测方法。在静态场景中，我们研究了一种基于 Spark 构建的并行可扩展机器学习算法，用于 ATM 欺诈检测，并使用多种机器学习模型进行训练，包括朴素贝叶斯（NB）、逻辑回归（LR）、支持向量机（SVM）、决策树（DT）、随机森林（RF）、梯度提升树（GBT）和多层感知机（MLP）。我们还采用了多种平衡技术，例如合成少数类过采样技术（SMOTE）及其变体，以及生成对抗网络（GAN），以应对数据集中少数类样本稀缺的问题。此外，我们还提出了适用于流式环境的基于流数据的 ATM 欺诈检测方法。我们基于滑动窗口的方法会收集指定时间间隔内发生的 ATM 交易，并利用这些交易训练多个机器学习模型，包括朴素贝叶斯（NB）、随机森林（RF）、决策树（DT）和 K 近邻（KNN）。我们根据这些模型较低的模型复杂度和较快的响应时间选取了它们。在两种环境中，随机森林（RF）均表现最佳。在静态场景中，RF 的平均曲线下面积达到 0.975；在流式环境中，其平均曲线下面积达到 0.910。实证结果还证明，RF 在统计意义上显著优于次优模型。

关键词： 欺诈检测；Spark；流处理；不平衡，V-GAN

1. 引言

客户普遍将自动柜员机（ATM）作为日常广泛接受的支付方式之一。过去二十年间，随着金融机构努力满足客户需求并提供先进的服务管理，ATM 的需求激增。然而，这种迅猛增长引起了诈骗者的注意，他们以此为目标并造成了重大经济损失。这给客户造成了严重损失，并进一步影响了“银行与客户”关系。采取适当措施便可避免这种情况，从而赢得客户的信任，增强客户对金融机构和组织的信心。

* 通讯作者

金融欺诈有多种形式，包括信用卡欺诈、洗钱、借记卡欺诈等。一旦发生此类金融犯罪，金融机构便会蒙受重大损失。因此，任何金融机构或组织都必须谨慎应对。各种金融欺诈造成的负面影响引起了利益相关者的关注。尽管目前尚无普遍认可的金融欺诈定义，《牛津英语词典》[1]将欺诈定义为“意图通过不正当或犯罪手段欺骗他人，以谋取经济或个人利益。”Phua 等人 [2]将欺诈描述为“滥用营利性组织的系统，但不一定会导致直接的法律后果”。Wang 等人 [14] 将其定义为“故意实施违反法律、规则或政策的行为，以获取未经授权的经济利益。”目前，金融机构的繁荣发展以及客户关系管理都与这些金融工具息息相关。

组织依据对海量数据（即“大数据”）的分析来制定政策并做出关键决策。尤其是这些金融欺诈数据集，属于大数据范式的范畴。五个 V——数据量、速度、多样性、真实性和价值[11] 是大数据的主要特征。目前有多种大数据处理工具，例如 Hadoop 和 Spark，可帮助组织收集和管理海量数据。近年来，Spark 凭借其统一的数据分析引擎而广受欢迎，该引擎能够处理海量数据集。Spark 提供多种抽象，包括数据框（DF）和弹性分布式数据集（RDD），它们具有不可变、支持持久化、可缓存等特性，并划分为多个数据分区，从而支持并行和分布式操作。此外，Spark 还支持 Scala、Python、Java 和 R 等多种编程语言。Spark 还通过 SparkSQL 扩展了对 SQL 的支持，借助 SparkMLlib 开发可扩展的机器学习算法，并以离散化流（DStreams）作为抽象来进行流式分析。DStream 是一段时间内收集的 RDD 集合，可实现批流处理。Spark 能够轻松集成 Apache Kafka、Kinesis 或 TCP 套接字等数据摄取工具，以及 Hadoop、Hive 等其他大数据处理工具，从而管理连续的数据流。

在前文讨论的各类金融欺诈中，ATM 欺诈检测尤其引起了我们的关注。我们可以检查 ATM 交易历史，以检测欺诈交易。我们认为，以下情形可能有助于识别欺诈交易：(i) 借记卡或信用卡被盗或遗失，(ii) 异常的交易历史，(iii) 在异常地点发起的交易，以及 (iv) 涉及大额交易的交易，这些情况可能会增加交易成为欺诈交易的概率。在有限的时间内实时开展此类分析相当困难。因此，有必要运用各种机器学习技术来实现这一过程的自动化。ATM 欺诈检测本质上是一项二元分类任务，其中欺诈交易被视为正类，非欺诈交易则被视为负类。

鉴于在大数据范式下开展欺诈检测的重要性，本研究采用可扩展的机器学习算法和高效的解决方案方法，在静态和流式环境下研究了大数据范式中的 ATM 欺诈检测。为此，我们开展了以下研究：

- 首先，在静态场景中，我们基于从印度收集的历史自动取款机交易数据开发了机器学习模型。为了构建可扩展的机器学习模型，我们使用了 Spark MLlib。
- 其次，在流式环境中，我们使用 DStreams 定期捕获数据，并采用基于滑动窗口的欺诈检测方法。

[1] 《牛津简明英语词典》，第十版，出版商，1999年

本研究分别应对了静态和流式环境中的以下挑战：(i) 处理欺诈检测数据集中的不平衡问题；(ii) 开发可扩展的机器学习模型，以满足大数据范式的要求；(iii) 在流式环境中，延迟应尽可能低，并且需要及时提供分析结果，同时不影响性能；(iv) 流式处理方法应支持大量数据流；(v) 在大数据范式下实现整个欺诈检测流程的自动化。

在本研究中，我们分析了来自印度的 ATM 交易数据，以进行欺诈检测。如前所述，ATM 欺诈检测问题对日常生活有重大影响，这促使我们致力于提供有效且高效的解决方案。本研究采用二元分类任务进行 ATM 欺诈检测。为此，我们使用了多种机器学习技术：朴素贝叶斯、逻辑回归、支持向量机、决策树、随机森林、梯度提升树和多层感知机。所收集的 ATM 交易数据具有高度不平衡的特点。我们采用以下技术处理数据不平衡问题：SMOTE 及其变体（SMOTE-ENN、SMOTE-Tomek）和 ADASYN。此外，鉴于生成对抗网络变体（即 V-GAN 和 W-GAN）近期的进展与成果，我们也采用了这些方法。

所提出工作的主要贡献如下：

- 提出了基于 Spark 的韧性 ATM欺诈检测框架。
- 在静态场景下探索并比较了用于 ATM 欺诈检测的机器学习算法。
- 研究了使用生成对抗网络变体（V-GAN 和 W-GAN）处理 ATM 欺诈检测中数据不平衡问题的方法。
- 提出了基于滑动窗口方法的流式分析 ATM 欺诈检测方案，从而处理延迟问题并实现实时决策。
- 通过采用多种数据平衡技术处理不平衡问题。

本文其余部分安排如下：第 2 节介绍文献综述。第 3 节讨论所提出的框架，该框架针对静态和流式环境下的 ATM交易数据采用二元建模。第 4 节介绍机器学习的背景理论以及所采用的平衡技术。第 5 节描述数据集。第 6 节讨论各种技术的结果。最后，第 7 节对本文进行总结。

2. 文献综述

本节将介绍相关研究，这些研究旨在缓解银行与金融行业中的欺诈检测问题，涉及静态和流式环境下的信用卡欺诈检测、财务报表等领域。

数据挖掘算法在事后欺诈检测中发挥着至关重要的作用 [1]。这些算法包括逻辑回归、多层感知机、决策树等。采用半监督和无监督技术等前沿方法挖掘数据流中的数据，也可以实现近乎实时的欺诈检测 [2,3]。在线分类、在线聚类、异常值检测等都是一些最先进的方法。关于信用卡欺诈检测的研究已有相当数量 [1,4-5]，然而，ATM/借记卡欺诈检测却鲜少成为研究课题。据我们所知，只有一篇研究论文探讨了 ATM 交易欺诈，并提出了一种专门用于金融数据流主动欺诈管理的语言 [3]。根据 ATM 卡的使用方式——在 ATM、销售点 (POS) 设备或网上银行使用——可分为三种情况。还可能发生其他类型的欺诈；由于由此产生的数据类型从结构化到半结构化再到非结构化不等，相应的检测方法也各不相同。Wei等人 [15] 观察了其中涉及的各种特征和挑战

在信用卡欺诈检测领域。Chan等人 [6]率先在分布式环境中开发了可扩展的信用卡欺诈检测系统。作者阐明了在开发解决方案时考虑可扩展性的重要性。

Chen 等人 [7] 提出了一种二元支持向量机，即支持向量机与遗传算法（遗传算法）的混合方法。在此方法中，利用遗传算法选出最优的支持向量组合。此外，还采用自组织映射来估计输入数据的分布。为实现实时分析，Dorronsoro 等人 [8] 提出了一种基于神经分类器的信用卡欺诈检测在线系统。为检测欺诈，作者提出了费舍尔判别分析的一种非线性变体。该策略能够从合法流量中分离出大量欺诈活动。Quah 等人 [9] 提出了一种实时欺诈检测系统，该系统利用自组织映射识别在线欺诈性信用卡交易和消费者行为。Sanchez 等人 [10] 应用关联规则挖掘，寻找异常行为模式，以协助识别欺诈交易。作者在交易数据库上展示了所提出技术的效果。Srivastava 等人 [11] 设计了一种基于隐马尔可夫模型的信用卡欺诈检测方法，包括以下步骤：（i）首先，利用聚类技术将具有相同消费模式的客户归为一组。（ii）随后，运行隐马尔可夫模型以识别能够捕捉欺诈交易的关键模式和序列。Zaslavsky 和 Strizhak [12] 将自组织映射用于信用卡欺诈检测。Wang 等人 [14] 提出了基于信息技术的范式，用于应对多种欺诈行为：（i）系统攻击和（ii）非系统攻击。作者在台湾金融欺诈数据集上研究了其框架的有效性。

信用卡交易数据通常将可扩展性视为必须在实时环境中解决的主要问题之一。Syeda等人 [13] 开发了一种采用共享内存架构的并行粒化神经网络，以解决上述问题。该网络应用于Visa卡交易，以识别欺诈交易。Wei等人 [15] 使用交易向量识别并存储客户行为，以应对欺诈数据集中的不平衡问题。作者提出了一种对比挖掘器，通过挖掘对比模式并识别非法行为，高效提取关键信息。由于在线活动大幅增加，一些欺诈者正攻击最常用的网站。文献中提出了许多旨在识别此类攻击的方法 [31]。Arya等人 [16] 采用了一种受神经科学启发的脉冲神经网络，并将其应用于多个分类问题。

Ravisankar 等人 [27] 提出了一种欺诈检测框架，通过分析财务报表来识别关键财务比率。借助这一框架，可以评估公司是否背负了过多债务或持有过多库存。财务数据中的缺失值往往会影响模型的性能。Fraquad 等人 [29] 提出了一种改进的主动学习框架，用于客户流失预测和保险欺诈检测，并使用支持向量机提取如果—那么规则。这是一种分为三个阶段的方法：(i) 使用支持向量机进行递归特征消除；(ii) 使用主动学习生成合成数据；(iii) 使用决策树和朴素贝叶斯生成规则。Vasu 和 Ravi [30] 提出了一种欠采样技术，以应对银行与金融领域许多应用中的不平衡问题。该方法使用 K 反向最近邻从多数类中移除离群值，然后使用分类器预测欺诈交易。Sundarkumar 和 Ravi [23] 分析了 CSMINING小组提供的数据集。作者在从 API调用中提取特征后，采用互信息作为特征选择标准。该数据集本身存在类别不平衡，因此，作者在多种机器学习技术中采用了过采样技术。

客户有时会怀疑自己是否正在正规网站上进行交易。因此，确定网站是否遭受网络攻击至关重要。Dhanalakshmi 等人 [34]提出了一种识别网络钓鱼网站的策略。为了判断网站是否合法，他们的方法会检查服务器的 IP 地址、域名和互联网域名，以确认网站的合法性。Lakshmi 和 Vijaya [33] 提出了一种用于网络钓鱼检测的监督学习机制。作者从 200 个网站 URL 中提取特征，这证明了该方法的有效性。PSO 训练的分类关联规则挖掘（PSOCARM）[18], 是一种基于规则的分类器，可判断是否检测到网络钓鱼。此外，Tayal 和 Ravi [19] 将上述问题表述为组合优化问题，并据此开发了模糊规则关联规则挖掘方法。这种基于模糊规则的挖掘器使用粒子群优化进行训练，并应用于事务数据库以提取规则，从而有助于检测钓鱼攻击。Pradeep 等人 [22] 还提出了一种基于萤火虫算法和阈值接受算法的基于规则的分类器，用于对欺诈性财务报表进行分类。

恶意软件检测面临诸多困难，需要强大、先进的安全系统。恶意软件攻击者通常利用应用程序接口（API）调用窃取信用卡信息、重要个人数据等。为获取这些信息，Sundarkumar 等人 [20] 采用了这些 API 调用信息和主题建模方法。他们的研究结果表明，决策树和支持向量机的表现优于其他方法，其中决策树在可解释性方面排名第一。Sheen 等人的另一项研究 [32], 利用从可执行文件中提取的特征和分类器集成来识别恶意软件。作者的主要目标是采用尽可能少的分类器来使用和声搜索算法。另一方面，欺诈检测也可以通过单类分类（OCC）方法来解决。Kamaruddin 和 Ravi [17] 在 Spark 环境下实现了上述混合架构，即 PSO-AANN 架构，使其适用于大型数据集。该架构的表现优于现有方法，例如单类支持向量机等。针对欺诈检测中的稀有类问题，Sundarkumar 和 Ravi [21], 提出了一种欠采样方法，将 K 反向最近邻与单类支持向量机结合使用，以移除多数类中的冗余样本。作者在汽车保险欺诈检测数据集和客户信用卡流失预测数据集上验证了该方法的有效性。

现在，我们将回顾利用实时分析解决欺诈检测问题的相关研究。Thennakoon 等人 [46] 利用机器学习模型和应用处理接口（API）机制标记欺诈交易。Arya 和 Sastry [49] 提出了一种基于预测建模的信用卡欺诈检测方法，并将其命名为用于在实时数据流中检测欺诈交易的深度集成学习。作者试图解决过拟合问题，这是以往研究的主要关注点之一。他们采用深度学习网络集成来识别信用卡欺诈。Mittal 和 Tyagi [48] 讨论了各种计算技术在实时信用卡欺诈检测中的优缺点。Abakiram 等人 [50] 提出了一种基于深度学习的实时信用卡欺诈检测模型。作者采用深度自编码器对欺诈交易进行分类。Rajeswari 和 Babu [45] 提出了一个实时信用卡欺诈检测系统。作者提出了一个框架，其中模型基于历史数据开发并运行，据此实时识别欺诈交易。Carcillo 等人 [47] 提出了一种使用 Spark 进行信用卡欺诈检测的可扩展框架。作者将 Kafka、Cassandra 等数据摄取工具集成到其框架中，并模拟以数据流形式传入的信用卡欺诈交易。他们采用了三种不同的机制来识别欺诈交易。

3. 所提方法

本节首先介绍静态场景下的所提方法，然后继续讨论流式环境。

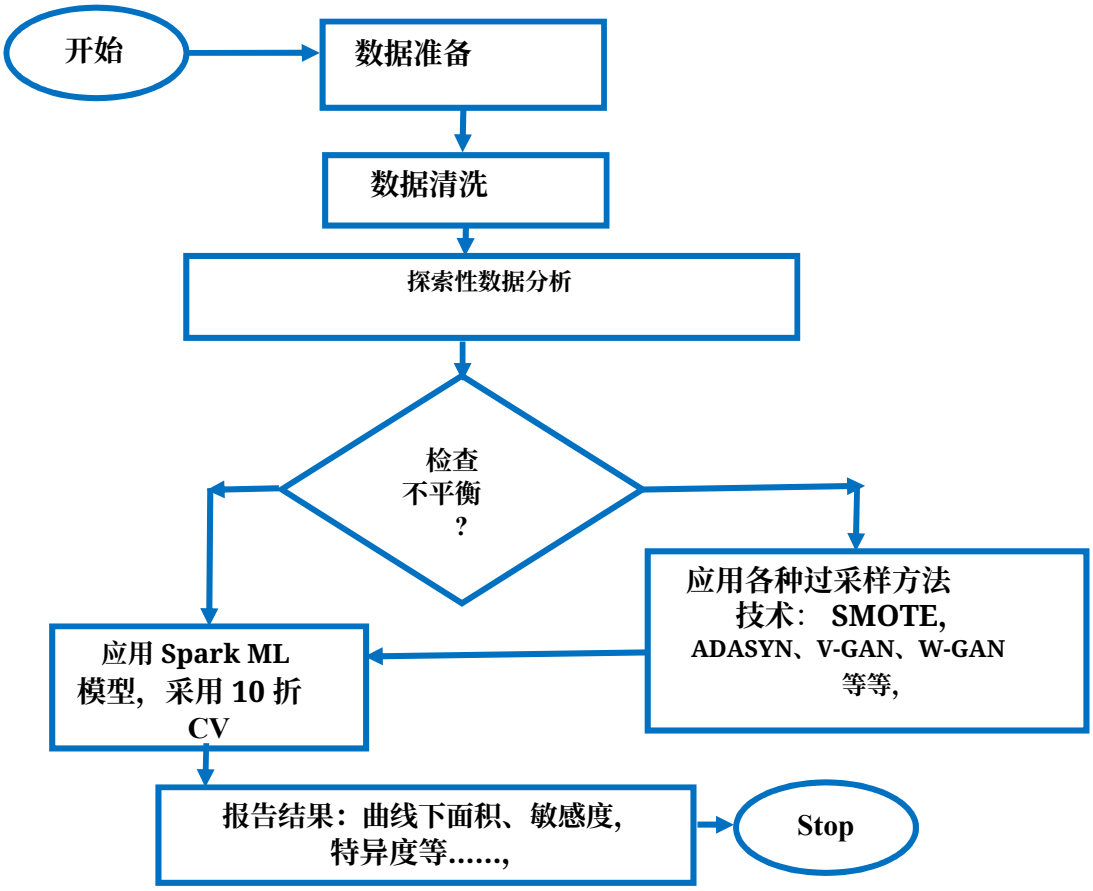

图 1. 静态环境下所提方法的流程图

3.1 Spark 下的二元分类方法

所提方法如图 1 所示，框图如图 2 所示，可用于静态环境，此时模型基于历史数据进行训练。该过程从数据收集步骤开始。在本研究中，我们收集了来自印度的 ATM交易数据。在印度，所有银行都维护着用于监控 ATM 每日交易的软件。收集的数据集包含通过三种不同模式进行的交易，即：(i) ATM交易数据，(ii) 销售点 (POS) 交易，以及 (iii) 互联网交易。其中包含与客户相关的关键私密信息。为保护数据完整性和客户隐私，我们对特征名称进行了掩码处理。

由于“垃圾数据输入，垃圾分析输出”这句话非常适用于机器学习，因此，为了提高数据的丰富度，我们采用以下数据清洗步骤：(i) 删除重复交易，因为相同的信息被保存了两次，而这确实不会增加任何

价值；(ii) 删除数据集中的不完整数据，这些数据可能包含因人为错误或其他原因而损坏的特征；(iii) 过滤掉几乎全为空值的特征。我们知道，缺失数据需要通过插补技术处理，但如果某个特征的空值超过 90%，这些插补技术就会彻底失效。

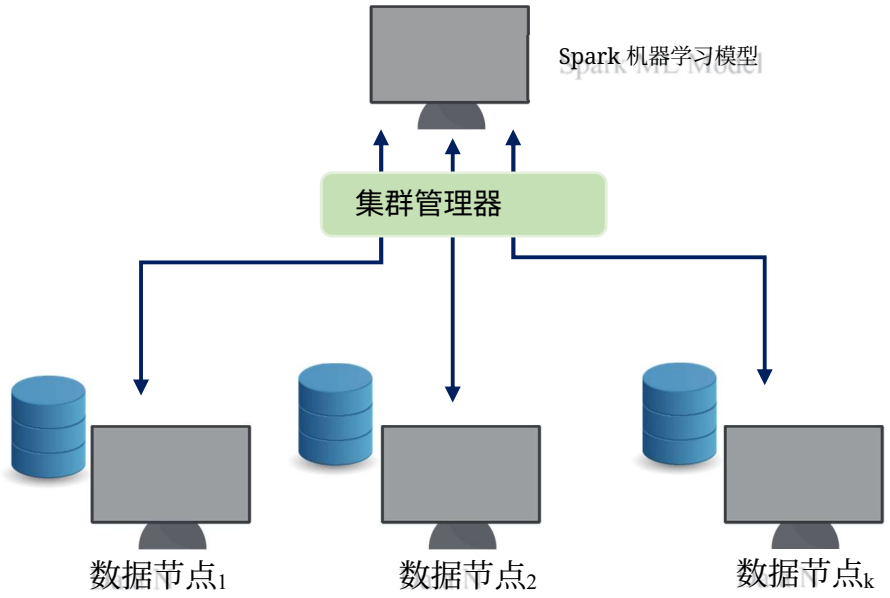

图 2. 静态场景下 ATM 欺诈检测框图

因此，我们删除了空值较多的特征。随后，可以分析数据以获取重要见解，并借助生成的统计摘要检验假设。所有特征均已归一化。现在可以看出，该数据集属于不平衡类型。对此，我们采用 ADASYN、SMOTE 等多种平衡技术进行处理。数据集按 80:20 的比例划分为训练集和测试集。在此过程中采用分层随机抽样，以确保训练集和测试集中正类与负类的比例相同。后续章节将详细讨论所有这些平衡技术。需要注意的是，进行过采样时，测试数据集必须保持不变。完成上述操作后，数据集现已存储在弹性分布式数据集（RDD）中（详见第 4.1.1 节）。

现在进入建模阶段，我们将在此阶段构建朴素贝叶斯、逻辑回归等机器学习模型。所有这些模型都在 Spark 环境中构建，并使用了 Spark MLlib。具体做法是构建一个由多个阶段组成的流水线。每个阶段都会使用前一阶段生成的数据。Spark 中的流水线遵循有向无环图（DAG），各阶段按拓扑序排列。我们使用 Spark MLlib 中提供的特征索引器处理特征，然后将整个数据点转换为一个列表，这是 Spark MLlib 模型所必需的。接着，使用训练数据集训练模型，并在测试数据集上进行测试。上述过程同样适用于其余 Spark 模型。在构建模型时，我们采用了网格搜索超参数调优技术。此外，我们还采用了 10 折交叉验证，并对其性能进行了比较。我们对曲线下面积（AUC）、敏感度和特异度分数等指标进行了展示和深入分析。

算法 1： 滑动 窗口方法：流式环境下的 ATM 欺诈检测

```
1：收集数据集
2：初始化 ws ← 窗口大小
3：初始化 sl ← 滑动间隔
4：创建无界表 ← ϕ
5：启动流处理上下文(Time)
6：当输入流 == 为真 时
7:        无界表 ← 追加输入流
8：如果窗口 == 完整 则
9:              训练数据 ← 窗口[:ws-1]
10:             测试数据 ← 窗口[ws]
11:             使用训练数据训练模型
12:             使用测试数据验证模型
13:             曲线下面积 ← 在测试数据上计算曲线下面积得分
14:             敏感度 ← 在测试数据上计算敏感度
15:             特异度← 在测试数据上计算特异度
16:             报告曲线下面积、敏感度和敏感度
17: 结束 if
18: 按照 sl 滑动窗口，遍历无界表
19: 结束 while
20: 停止 流处理上下文
```

3.2 流式数据的滑动窗口

第 3.1 节讨论的方法非常适用于静态环境。然而，处理需要按指定时间间隔处理的数据流时，则需要采用新的机制来应对这一需求。为满足这些挑战和需求，我们提出了一种受滑动窗口启发的流式处理方法，用于检测 ATM 欺诈数据。该算法见算法 1，框图如图 3 所示。

Spark 提供 DStream（第 4.1.2 节将详细讨论），它是一个由多个 RDD 组成的数组，每个 RDD 都包含在特定时间间隔内收集的交易信息。DStream 要处理的所有操作都在流式环境中执行。因此，需要在流程开始时初始化流式环境，并在操作完成后将其关闭。所有变量、模型、广播变量等都仅限于此流式环境。值得一提的是，Spark 提供批流处理。因此，操作针对以批次形式收集的数据执行。借助数据摄取工具，流数据会被处理并传入 HDFS 集群。在我们的场景中，我们没有使用任何数据摄取工具。不过，我们使用 Spark 提供的 QStreams（第 4.1.3 节将详细讨论）模拟了数据摄取。我们创建了一个无界数据帧，其中的数据流以多个批次的形式不断到达。

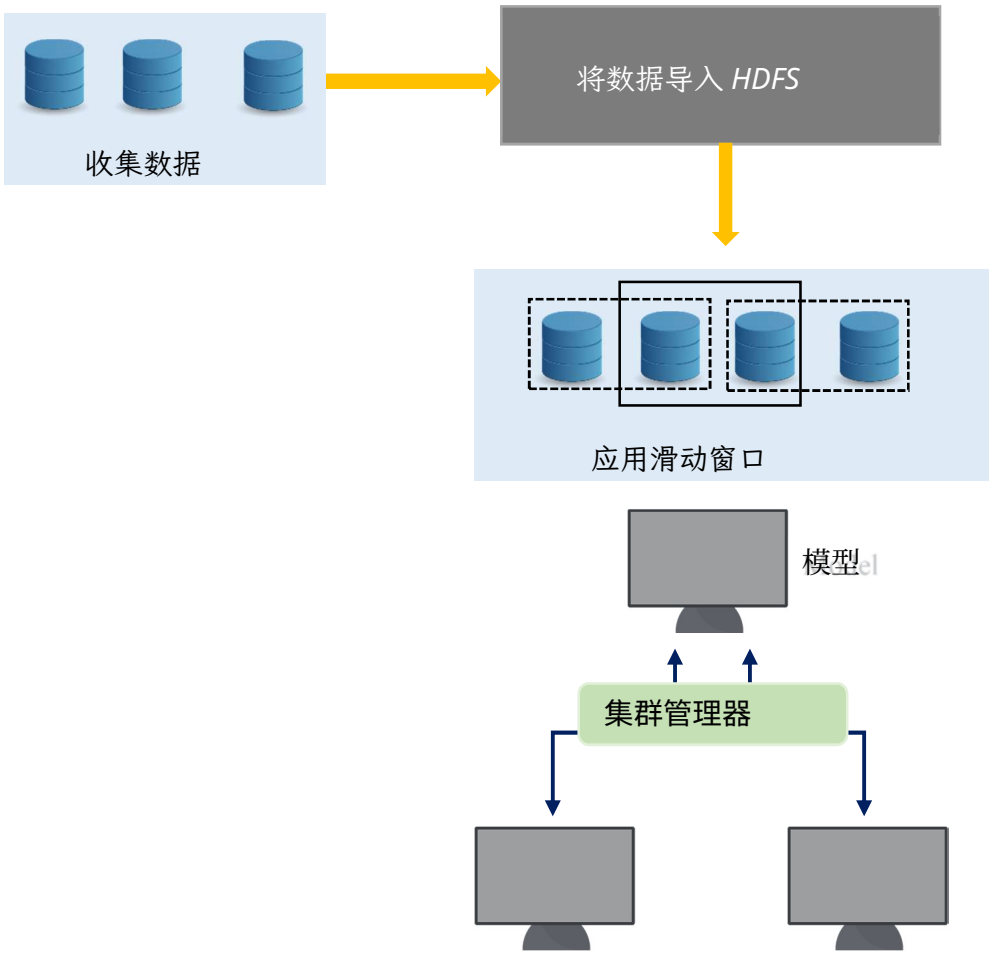

图 3. 流式环境下 ATM 欺诈检测框图

我们采用了滑动窗口方法。该方法的设计如下：交易数据流按指定的时间间隔以批次形式收集。在特定时间间隔内迄今收集的数据存放在一个 RDD 中。收集的 RDD 数量取决于窗口大小。窗口过大会增加延迟。设滑动窗口长度为 *ws*，则使用前 '*ws-1*' 个批次进行训练，并使用滑动窗口中的最后一个批次进行测试。如果尚未收集到所需数量的批次，系统就会等待，直到收集完整。正如前文所述，窗口越大，需要处理的历史数据就越多，也会进一步增加延迟。考虑到这些因素，我们将窗口大小设为 2。这样一来，我们只使用最新的数据。因此，底层机器学习模型使用窗口中的第一个 RDD 进行训练，再使用第二个 RDD 进行测试。我们将步长（即窗口的移动速度）设为 1。因此，第二个窗口中的第二个 RDD 会成为下一个窗口中的第一个 RDD。只要有新数据可用并被摄取到集群中，这一过程就会持续进行。

4. 所采用技术概述

本节简要讨论当前研究中采用的整体机器学习技术、各种平衡技术以及单类分类技术。

4.1. Spark 抽象

4.1.1 弹性分布式数据集

Spark 提供了一种称为弹性分布式数据集（RDD）的抽象。它是分散在整个集群中的数据集分区集合，并且具有不可变性。借助这一特性，Spark 可以执行并行处理。发生故障时，它具备有助于恢复数据的容错特征。它主要支持两种不同的操作：（i）转换操作和（ii）行动操作。转换操作是一组用于将数据集从一种形式转换为另一种形式的操作。这些操作会作用于 RDD 的每个元素。行动操作是聚合操作，因此其结果会被收集并返回给驱动程序。Spark RDD 支持惰性求值。为了在必要时应用更改，它会先将这些更改记录下来。

4.1.2 离散化流

图 4 展示了离散化流（DStream），这是 Spark Streaming 提供的另一种抽象。在流式环境中，信息持续不断地流动。因此，DStream 包含一系列持续生成的 RDD。这里，每个 RDD 都由特定时间段内收集的信息构成。这些 DStream RDD 支持应用所有操作，包括转换操作和行动操作。与 RDD 一样，DStream 是一种不可变的分布式流，支持并行操作。应用于 DStream 的操作实际上都会应用于这些 RDD。

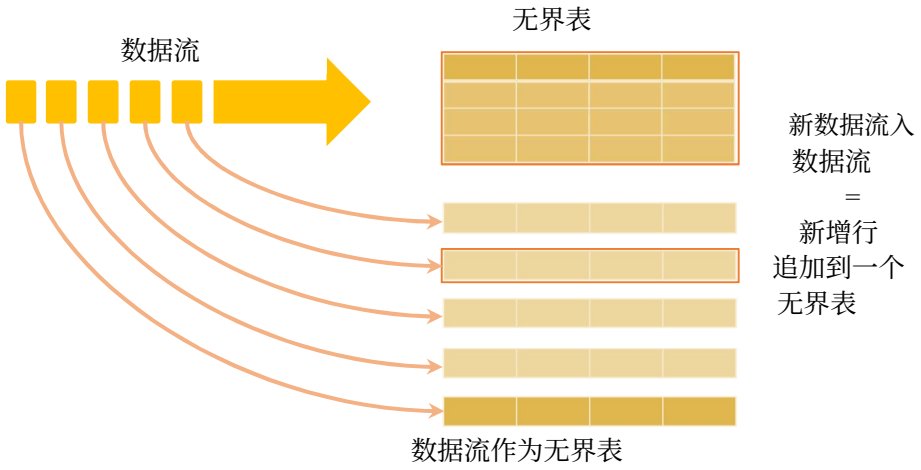

图 4. DStreams

4.1.3 队列流与窗口操作

在现实世界中，数据集是在一定时间范围内收集的。这些收集到的数据流会使用 Spark 按批次进行处理。因此，底层的 RDD 队列被视为一批 RDD。由此，所有操作都在这批 RDD 上执行。为支持这一点，Spark 提供了一个名为队列流的应用程序接口，允许用户根据一组 RDD 创建 DStreams。它有助于从一段时间内积累的 RDD 队列中生成输入流。生成的这类数据流可以按顺序处理，也可以同时处理。

此外，Spark 还支持应用窗口操作，这有助于在数据的滑动窗口上执行转换操作。窗口中包含的所有 RDD 都会合并，并在窗口沿 DStream 滑动时对其执行相应操作。需要注意的是，所有这些窗口操作都基于以下两个重要因素：

(i) **窗口长度：**用于定义窗口的持续时间；(ii) **滑动间隔：**执行窗口操作的时间间隔。

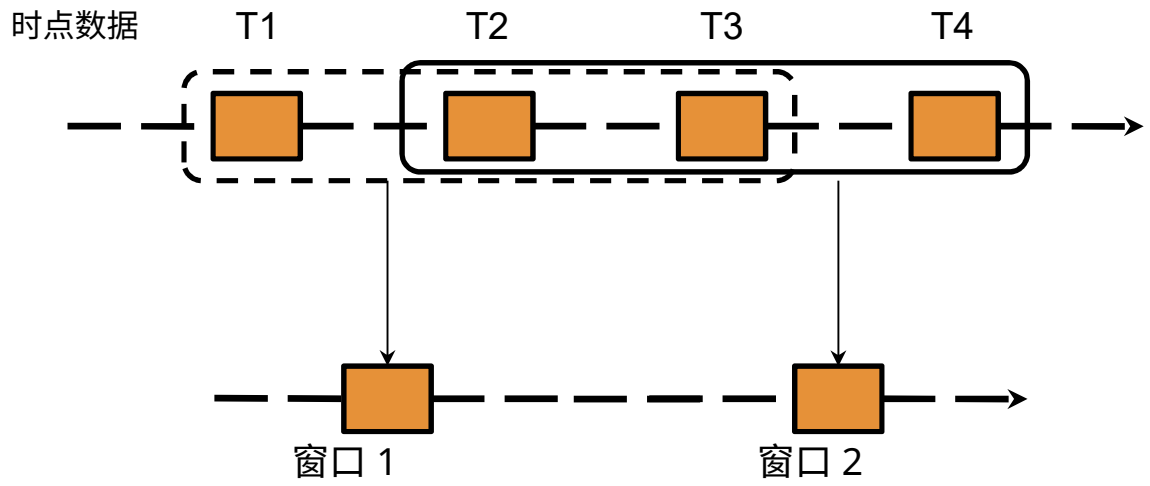

图 5. DStreams 上的滑动窗口操作（窗口长度大小 = 3，滑动间隔=1）

4.2 机器学习技术

4.2.1 朴素贝叶斯

朴素贝叶斯（NB）分类器是基于贝叶斯定理的概率模型，用于执行分类任务。它们速度快且易于实现。朴素贝叶斯基于以下假设：(i) 所有特征彼此独立；(ii) 所有特征对决策变量/类别变量的贡献相同。这些假设限制了朴素贝叶斯在现实场景中的应用，因为这些假设往往有一项不成立，从而影响其性能。

4.2.2 逻辑回归

逻辑回归（LR）是用于二元分类任务的最受欢迎的监督分类器之一。它也称为逻辑模型，用于估计事件发生的概率，例如欺诈或合法、流失或未流失等。

4.2.3 支持向量机

支持向量机（SVM）是一种由 Vapnik 提出的监督式机器学习算法 [24], , 用于处理各种分类和回归任务。SVM 的主要目标是通过最大化间隔，找到最优超平面，作为对 n 维数据进行分类的决策边界。这在形式上称为结构风险最小化。SVM 使用多种核函数，包括线性核、径向基函数（RBF）等；核函数有助于转换训练数据，使 SVM 能够学习非线性决策面。

4.2.4 决策树

决策树 [39] 是解决包括分类和回归在内的众多机器学习问题的最有效方法之一。决策树根据训练数据构建具有内部节点和叶节点的树结构。每个叶节点代表一个类别标签，而每个内部节点则代表分类规则。处理缺失值、自动进行特征选择，以及处理类别特征和数值特征，都是决策树的优势。根节点分裂基于多种因素，包括基尼系数、熵等；决策树还会生成一组易于理解的规则。因此，决策树被视为白盒模型。

4.2.5 随机森林

Ho [25], 提出的随机森林是一种集成机器学习技术。它可用于解决分类和回归问题。随机森林的训练过程如下：使用自助采样技术从数据集中选取一个随机子集。每棵决策树都独立构建。在测试阶段，将测试样本输入每棵决策树，而每棵决策树可能会给出不同的预测结果。最终决策通过多数投票得出。有效处理高维数据集是随机森林的一大特点。

4.2.6 梯度提升树

GBT 也是一种集成机器学习技术，由多棵决策树组成。GBT 擅长处理各种分类任务和回归任务。不过，RF 和 GBT 的主要区别如下：(i) 与 RF 不同，GBT 逐棵构建决策树，采用加法方式逐步生成。此外，每棵新树都会修正此前树所产生的错误。这种方法称为“提升”。(ii) RF 是通过汇总已构建的决策树来做出决策，而 GBT 会更早地做出决策。由于 GBT 容易过拟合，因此必须非常谨慎地调整超参数。

4.2.7 多层感知机

多层感知机 [26] 是最受欢迎的神经网络之一，可将输入特征集映射为类别变量集。它常被称为“通用逼近器”，可用于解决分类和回归问题。多层感知机由三个不同的层组成：输入层、隐藏层和输出层。隐藏层用于识别给定数据集中存在的非线性模式。权重通过标准反向传播算法进行估计。

4.3 平衡技术

In t本节将讨论当前研究中采用的所有平衡技术 .

4.3.1 SMOTE 及其变体

SMOTE

最受欢迎的重采样方法之一是 SMOTE [35]。它会在复制少数类样本的同时，创建新的合成样本。首先，它会从少数类中随机选择一个样本，然后使用 K近邻（K近邻）创建一个新的合成样本。重复此过程，直到生成所需比例的样本。

SMOTE-ENN

该方法由 Batista 等人开发，[38], 并使用编辑近邻（ENN）消除部分多数类样本点，从而丰富数据集。SMOTE-ENN 分两个阶段完成：第一阶段由 SMOTE 生成样本，第二阶段由 ENN 移除违背其指导原则的样本。根据 ENN，当某个多数类样本及其 K近邻中的多数类别与该样本不一致时，才将该多数类样本及其 K近邻从数据集中移除。

SMOTE-Tomek

该方法由 Batista 等人开发，[38], 并使用作为欠采样技术的 Tomek 链来丰富数据集。Tomek 链由一对分属两个不同类别、但彼此互为 K近邻的点组成。SMOTE-Tomek 分两个阶段完成：第一阶段由 SMOTE 生成样本，第二阶段移除 Tomek 链。

4.3.2 ADASYN

前面讨论的 SMOTE 及其变体在生成合成样本时没有考虑少数类的密度。因此，ADASYN [28] 通过引入这一原则，针对低密度区域生成大量样本，而在高密度区域生成较少或不生成样本。

4.3.3 生成对抗网络

GAN 最初由 Ian Good Fellow 等人提出 [36]。此后，重采样一直是 GAN 的多种应用之一。GAN 的主要目标是生成与训练数据的概率分布高度相似的数据。在本研究中，我们采用了两种不同的 GAN 变体，以如下方式实现过采样：

V-GAN

GAN [36] 的主要组成部分分别是判别器和生成器模型。生成器会创建伪造样本并试图欺骗判别器，而判别器则试图识别并分类这些伪造样本。因此，在 GAN 训练过程中会形成一个极大极小优化问题。生成器和判别器的权重依次进行优化。原始生成对抗网络是这种简单 GAN 变体的另一种称呼。

W-GAN

瓦瑟斯坦距离／推土机距离损失函数是 W-GAN 使用的损失函数[37], ，它采用了不同的损失函数。瓦瑟斯坦距离用于比较两个分布的相似性。V-GAN 和 W-GAN 的其余训练过程相同。

5 数据集描述

本研究收集了来自印度的 ATM 交易数据。这些 ATM 交易由印度银行维护的各种软件进行跟踪。该数据集包含数日内通过三个不同渠道完成的总计一百万笔交易，包括 ATM交易数据、销售点（POS）交易数据和互联网交易数据。这些数据包含此类交易的重要信息，例如支付方式、客户信息、交易金额等。为保护数据隐私和完整性，同时遵守银行法规，我们对特征进行了重命名。我们针对静态和流式环境采用了所有数据预处理技术，以清理数据。数据集经过整理后，共有 7,46,724 笔交易。由于删除了若干唯一值和空值特征，特征数量减少至 10 个。非欺诈类和欺诈类的数据比例为 87.80:12.20。因此，这是一个不平衡数据集。以下所有实验均采用 10折交叉验证（CV）进行。

5.1 评估指标

我们采用曲线下面积（AUC）来衡量所用分类器的稳健性。除了曲线下面积，我们还采用敏感度和特异度作为二元分类的其他评估指标。

曲线下面积（AUC）它已被证

明是处理不平衡数据集时的一项稳健指标，且为特异度和敏感度的平均值。曲线下面积的数学表示见公式（2）。

$$AUC = \frac{(Sensitivity + Specificity)}{2} \tag{2}$$

敏感度

它也被称为真正率（TPR），是被正确预测为正类的正样本数占所有正样本数的比例。其数学表示见公式(3)。

$$Sensitivity = \frac{TP}{TP + FN} \quad (3)$$

其中，TP 表示真正例， and FN is a false negative.

特异度也称为

真负率（TNR）。它是被正确预测为负类的负样本数占所有负样本数的比例。其数学表示见公式（4）。

$$Specificity = \frac{TN}{TN + FP} \quad (4)$$

其中 TN 表示真阴性 , and FP is a false positive.

表2：所有技术的超参数

模型	**超参数**
LR	‘正则化参数’：[0.1, 0.01, 0.001] ‘最大迭代次数’：[10,20,30,40,50,100,500]
SVM	‘正则化参数’：[0.1, 0.01] ‘最大迭代次数’：[30,40,50,100]
DT	‘划分标准’：{‘基尼系数’,’熵’} ‘最大深度’：[5,10, 15, 20]
RF	‘最大深度’：[5, 10, 15, 20] ‘基学习器’：[10,20,30,40,50]
GBT	‘最大深度’：[5, 10, 15, 20] ‘基学习器’：[10,20,30,40,50]
MLP	‘求解器’：{‘有限内存BFGS’，‘梯度下降’} ‘最大迭代次数’：[50, 100, 200, 300]
V-GAN	**判别器：** 隐藏层 1：128 个神经元，激活函数：’带泄漏的 ReLU’ 隐藏层 2：64 个神经元，激活函数：’带泄漏的 ReLU’ 隐藏层 3：32 个神经元，激活函数：’带泄漏的 ReLU’ 隐藏层 4：8 个神经元，激活函数：’带泄漏的 ReLU’ 训练轮次=10,000
W-GAN	**判别器：** 隐藏层 1：256 个神经元，激活函数：’带泄漏的 ReLU’ 隐藏层 2：128 个神经元，激活函数：’带泄漏的 ReLU’ 隐藏层 3：64 个神经元，激活函数：’带泄漏的 ReLU’ 隐藏层 4：32 个神经元，激活函数：’带泄漏的 ReLU’ 训练轮次=10,000

6 结果与讨论

在本节中，我们分别展示了静态和流式环境下的结果。首先，我们将讨论静态场景下的结果，然后讨论流式环境下的结果。

6.1 静态场景结果

在本研究中，我们采用了表2中列出的超参数。这些超参数均是在经过严格实验后确定的。如第3节所述，所有实验均采用相同的数据清洗和数据预处理技术。我们观察到，从印度收集的ATM交易数据集存在严重不平衡，比例为87.80 : 12.20。因此，我们采用了多种平衡技术，如 SMOTE、SMOTE-ENN、V-GAN 等。值得注意的是，所有实验均采用10折交叉验证（CV）。对于不平衡分类数据集，曲线下面积已被证明是一项稳健的度量指标，这一点在信用卡欺诈检测等以往案例研究中得到了充分体现。因此，在选择最佳模型时，我们更加重视曲线下面积。相关结果见表 3。此外，我们还分别在表4和表5中讨论了敏感度和特异度分数。

在所有机器学习模型中，使用 V-GAN 平衡技术对数据集进行平衡后，多层感知机在曲线下面积方面表现最佳（参见表 3）。梯度提升树和随机森林分别位居第二和第三。值得注意的是，随机森林和梯度提升树获得了相近的曲线下面积，其敏感度和特异度分数也反映了这一点。作为决策者，在选择最佳模型时，除了数值上的曲线下面积，还应考虑模型的复杂度。在这种情况下，随机森林表现最佳，因为与多层感知机相比，它的可学习参数较少；与梯度提升树相比，它的复杂度较低。此外，与另外两个表现最佳的模型相比，随机森林的平均曲线下面积仅有极小差异（曲线下面积相差 0.2%）。总体而言，与其他机器学习模型相比，所有基于树的模型表现都相对较好。在平衡技术方面，V-GAN 在取得最佳曲线下面积方面明显优于其他平衡技术。

选择最佳模型时，还可以考虑另一个平局判定因素。众所周知，在基于树的模型中，决策树是可解释性较强的模型。随机森林和梯度提升树属于集成模型，因此在可解释性方面，它们是复杂度较高、排名第二的模型。但仔细观察会发现，其曲线下面积比随机森林和梯度提升树低2%，这对于欺诈检测等关键应用而言是相当不可接受的。因此，尽管可解释性较差，随机森林和梯度提升树的排名仍高于决策树。由于数值表现更优且模型复杂度较低，随机森林被评为最佳模型。

表 3：10 折交叉验证下各种机器学习模型获得的平均曲线下面积

模型	平衡技术						
	不平衡	SMOTE	SMOTE-Tomek	SMOTE-ENN	ADASYN	V-GAN	W-GAN
NB	0.721	0.775	0.775	0.777	0.692	0.897	0.798
LR	0.709	0.803	0.803	0.805	0.690	0.893	0.869
SVM	0.807	0.806	0.805	0.807	0.694	0.890	0.853
DT	0.755	0.823	0.823	0.824	0.775	0.953	0.904
RF	0.759	0.837	0.837	0.838	0.791	**0.975**	0.920
GBT	0.759	0.837	0.837	0.838	0.791	0.975	0.920
MLP	0.757	0.837	0.831	0.831	0.721	**0.977**	0.904

敏感度分数方面也观察到了类似的结果（参见表4）。众所周知，敏感度也是一个关键参数，它会影响被正确识别为欺诈交易的交易数量。多层感知机凭借较高的平均敏感性得分，表现优于其他模型。然而，将模型复杂度和数值表现优势作为平局判定因素时，随机森林在平均敏感度方面脱颖而出，成为最佳模型。

另一方面，特异度分数也呈现出类似的情况（参见表5），它反映了被正确识别为非欺诈交易的数量。特异度也是另一个关键因素，因为忠诚客户的交易不应被标记为欺诈交易。观察发现，支持向量机的敏感度分数达到 1.0，但其特异度分数明显低于基于树的模型。考虑平局时的判定因素后，我们得出结论：随机森林最终成为静态场景下的最佳模型。

表4：各种机器学习技术在 10 折交叉验证中获得的平均敏感性得分

模型	平衡技术						
	不平衡	SMOTE	SMOTE-Tomek	SMOTE-ENN	ADASYN	V-GAN	W-GAN
NB	0.546	0.779	0.778	0.780	0.699	0.871	0.674
LR	0.540	0.782	0.781	0.783	0.660	0.787	0.745
SVM	0.736	0.775	0.774	0.775	0.632	0.781	0.707
DT	0.607	0.789	0.787	0.789	0.738	0.930	0.824
RF	0.618	0.773	0.773	0.776	0.769	0.961	0.850
GBT	0.618	0.773	0.773	0.776	0.769	0.961	0.850
MLP	0.613	0.781	0.781	0.781	0.675	**0.969**	0.825

表 5：各种机器学习技术在 10 折交叉验证中获得的平均特异度分数

模型	平衡技术						
	不平衡	SMOTE	SMOTE-Tomek	SMOTE-ENN	ADASYN	V-GAN	W-GAN
NB	0.895	0.772	0.773	0.774	0.685	0.924	0.922
LR	0.878	0.825	0.825	0.826	0.720	0.999	0.992
SVM	0.877	0.837	0.836	0.838	0.757	**1.0**	0.999
DT	0.902	0.857	0.859	0.859	0.813	0.975	0.991
RF	0.901	0.901	0.902	0.901	0.812	0.989	0.991
GBT	0.901	0.901	0.902	0.901	0.812	0.989	0.991
MLP	0.902	0.882	0.882	0.882	0.767	0.986	0.982

6.2 结果的统计检验

进行双尾 t 检验分析，以检验模型 A 在数值上优于模型 B 究竟只是巧合，还是确实源于其自身的优势。因此，我们将表现最佳的模型与其余模型进行了成对 t 检验分析。该 t 检验分析基于各模型通过 10 折交叉验证获得的曲线下面积进行，结果见表 3。t 检验结果见表 6。

零假设为：H_0：两种算法在统计上相等。备择假设为：H_1：两种算法在统计上不相等。

p 值决定是否接受或拒绝零假设。显著性水平设为 5%，自由度为 18（10+10-2）。因此，若要拒绝零假设，p 值应小于 5%。表 6 推断，当 p 值小于 0.05 时，零

假设被拒绝，备择假设被接受。这是对随机森林和决策树进行 t 检验后得出的结论。因此，在静态场景下，就曲线下面积而言，随机森林在统计学上比决策树更显著。然而，随机森林和梯度提升树在统计学上被证明是相等的。不过，就复杂度而言，随机森林低于梯度提升树，因此随机森林被认为是最佳模型。

表 6：配对 t 检验结果

模型	参数	
	t 统计量	p 值
随机森林 vs 梯度提升树	0.0	1.0
随机森林 vs 决策树	33.39	1.21×10^{-17}

6.3 流式环境结果

现在，我们将讨论模型在流式环境中的性能。正如我们之前所讨论的，ATM 交易数据集高度不平衡。因此，在一个窗口内收集的交易中可能没有任何欺诈交易。此外，仅提供一个类别相关数据点时，无法构建二分类模型。因此，在我们的方法中，针对流式环境下一个窗口内收集的负类和正类交易比例，我们作出了一项假设。因此，在每个数据流中，我们都会将欺诈交易（正类样本）与收集到的 ATM 交易一同纳入。因此，在流式环境中，我们仅考虑平衡数据集进行评估。此外，延迟也是评估流式环境中模型时的另一个关键参数。因此，模型应尽可能简单。为此，我们考虑了简单模型和基于树的模型，并比较了它们的性能。需要注意的是，所有模型均采用滑动窗口方式进行评估，并在本节中计算和讨论曲线下面积、敏感度和特异度等指标。采用平衡技术后，数据集现在包含 100 万笔交易。每个窗口包含 1000 笔交易，因此窗口数量为 1000。

表7：各种机器学习技术在滑动窗口上的平均曲线下面积得分 **s**

模型	平衡技术					
	SMOTE	SMOTE-Tomek	SMOTE-ENN	ADASYN	V-GAN	W-GAN
LR	0.811	0.811	0.811	0.691	0.842	0.875
KNN	0.854	0.855	0.855	0.774	0.870	0.896
DT	0.889	0.889	0.889	0.810	0.908	0.909
RF	0.896	0.898	0.898	0.821	**0.910**	0.911

此外，我们也将曲线下面积视为选择最佳模型的合适指标，结果见表7。我们在图6至图11中展示了所得的曲线下面积，以说明所采用的机器学习模型的异常值行为。我们还在图12至图15中展示了不同窗口上的曲线下面积，以概述各种模型的性能。这确实能够呈现机器学习模型的性能。总体而言，无论采用何种平衡技术，随机森林模型的表现都是最佳的，决策树位居第二。此外，图10显示，随机森林模型的异常值行为少得多，因此在识别正类时更加稳定。然而，决策树表现出较高的变异性，因此尽管具有可解释性，仍位居第二。敏感度和特异度分数也呈现出相同的情况。值得注意的是，随机森林不仅在平均曲线下面积、敏感度和特异度分数方面表现出色，而且变异性也较小，这进一步证明了其

稳健性。关于其他平衡技术，也观察到了相同的情况（参见图6至图11）。在本设置中，实证结果表明，V-GAN是一种有效的平衡技术，其次是SMOTE-ENN。

表 8：各种机器学习技术在滑动窗口上的平均敏感性得分

模型	平衡技术					
	SMOTE	SMOTE-Tomek	SMOTE-ENN	ADASYN	V-GAN	W-GAN
LR	0.806	0.806	0.806	0.667	0.848	0.914
KNN	0.860	0.861	0.861	0.746	0.828	0.859
DT	0.895	0.895	0.895	0.779	0.865	0.867
RF	0.899	0.895	0.895	0.793	**0.871**	0.873

Ta表 9：通过滑动窗口获得的各种机器学习技术的平均特异度分数 **s**

模型	平衡技术					
	SMOTE	SMOTE-Tomek	SMOTE-ENN	ADASYN	V-GAN	W-GAN
LR	0.815	0.816	0.816	0.715	0.836	0.836
KNN	0.849	0.850	0.850	0.801	0.913	0.933
DT	0.882	0.882	0.882	0.842	**0.950**	0.952
RF	0.893	0.892	0.892	0.810	0.948	0.949

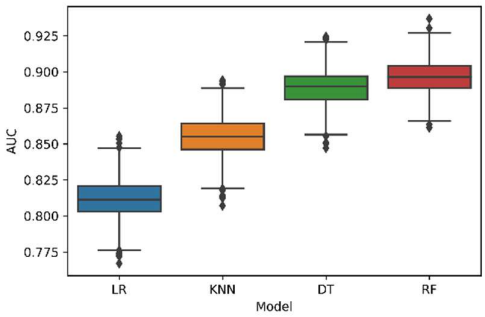

图6. 在流式环境中采用 SMOTE 时获得的曲线下面积

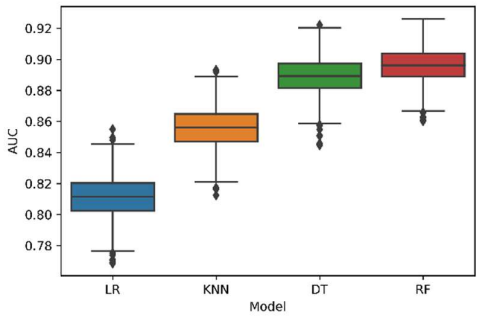

图7. 在流式环境中采用 SMOTE-Tomek 时获得的曲线下面积

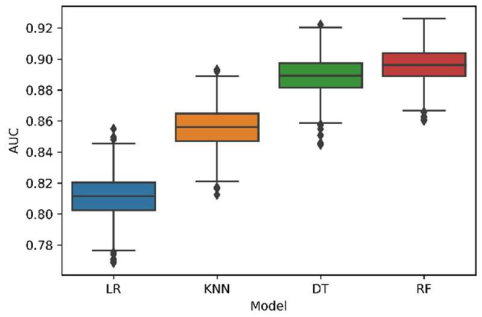

图8. 在流式环境中使用 SMOTE-ENN 时获得的曲线下面积

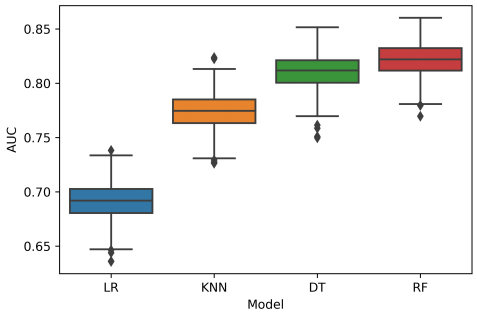

图9. 在流式环境中使用 ADASYN 时获得的曲线下面积

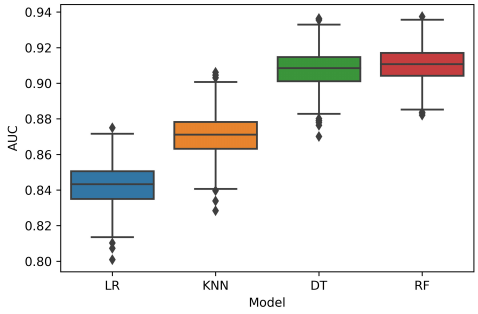

图10. 在流式环境中使用 V-GAN 时获得的曲线下面积

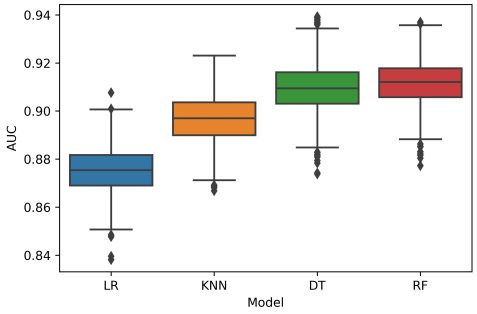

图11. 在流式环境中使用 W-GAN 时获得的曲线下面积

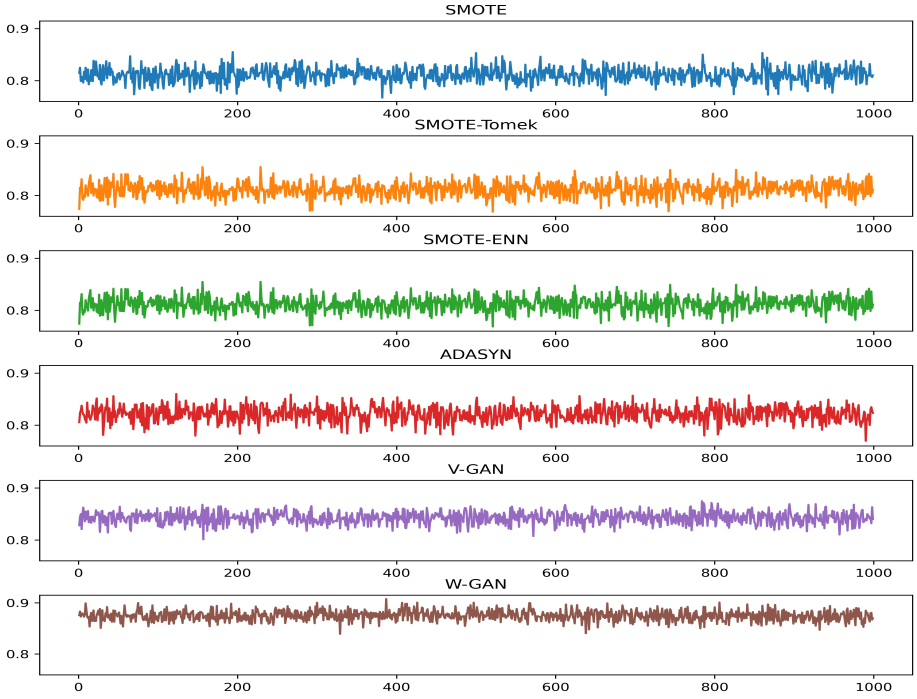

图12. 逻辑回归模型在多个窗口上的曲线下面积

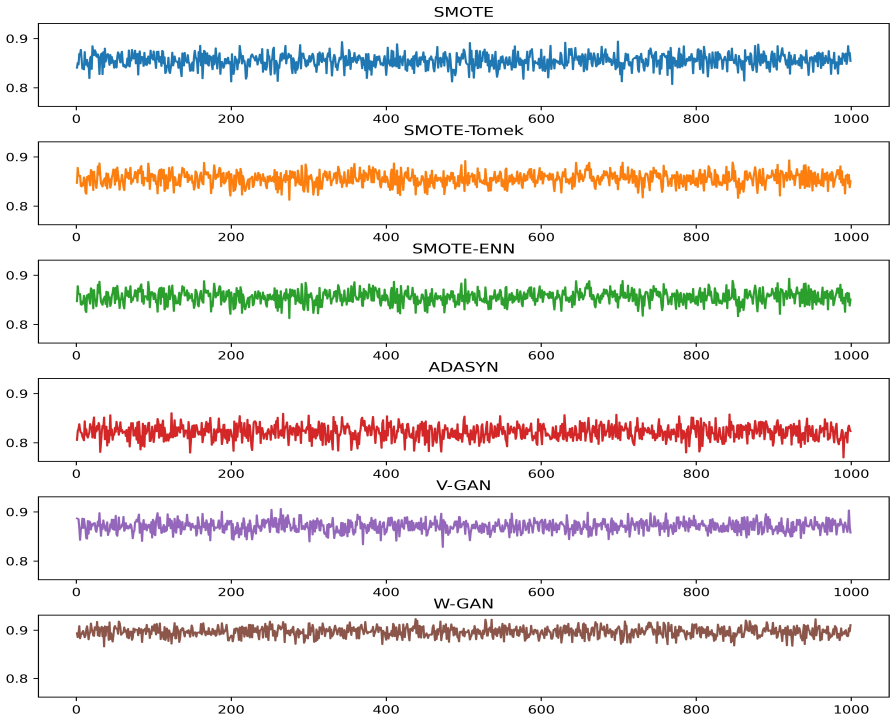

图 13. K近邻模型在多个窗口上的曲线下面积

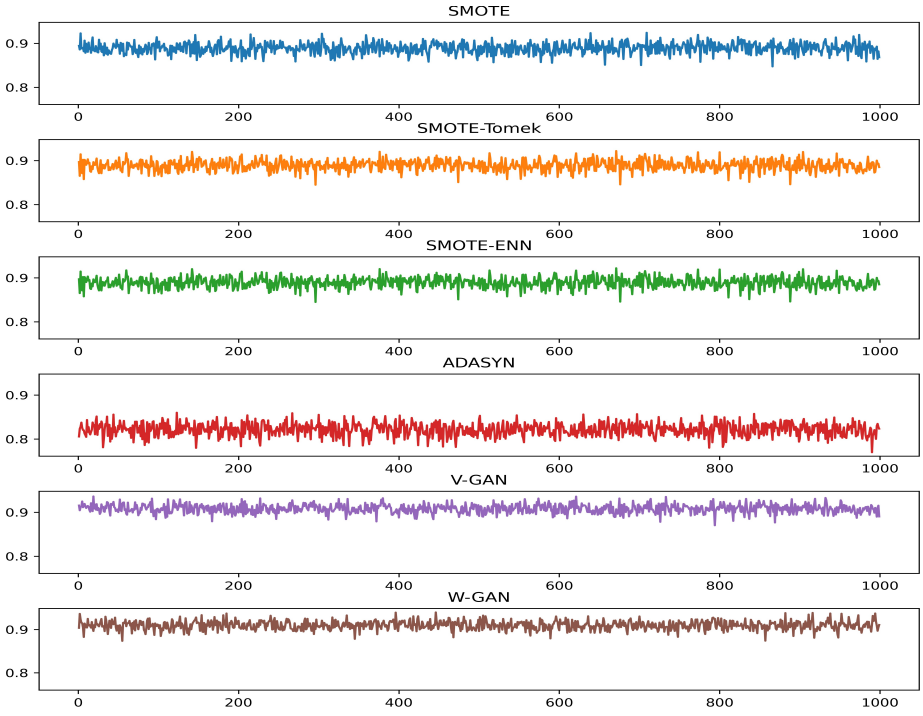

图 14. 决策树模型在多个窗口上的曲线下面积

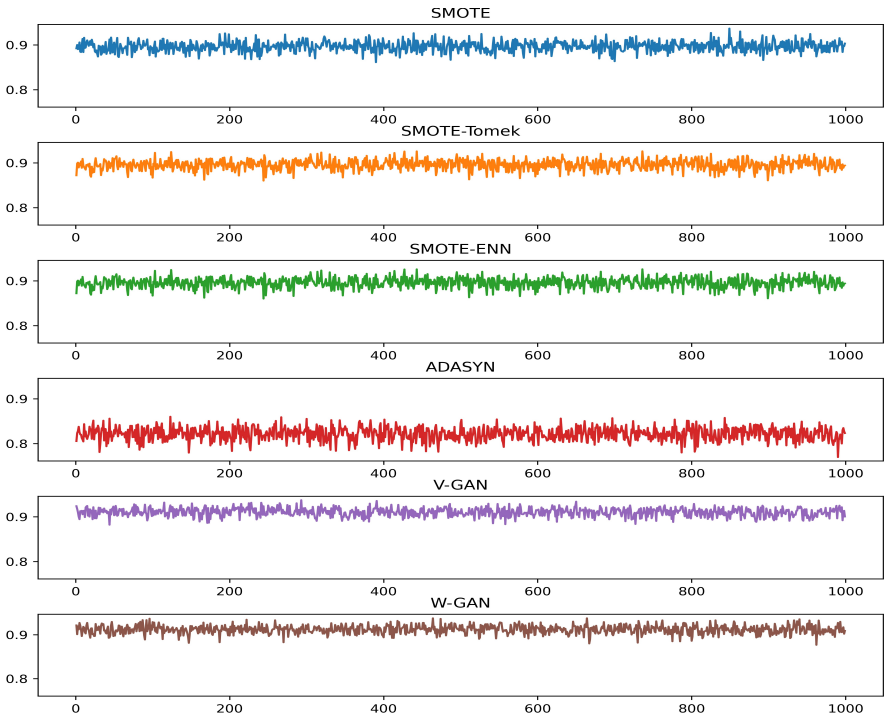

图 15. 随机森林模型在多个窗口上获得的曲线下面积

6.4 结果的统计检验

Wilcoxon检验是一种非参数检验，用于判断模型 A 优于模型 B 的性能是否纯属巧合，还是确实源于其优越性。因此，我们根据多个窗口上获得的曲线下面积，对排名前两位的模型进行了 Wilcoxon检验分析。Wilcoxon检验结果见表 10。

零假设为，H_0：两种算法在统计上相等。备择假设为，H_1：两种算法在统计上不相等。

p 值决定是否接受或拒绝零假设。显著性水平设为 5%。因此，若要拒绝零假设，p 值应小于 5%。表 10 表明，在流式环境中，随机森林在统计上的显著性高于次优模型决策树。

表 10：配对 Wilcoxon检验结果

模型	参数	
	Wilcoxon检验-统计量	p 值
决策树 vs 随机森林	204017.0	$5.34x10^{-07}$

7. 结论与局限性

在本研究中，我们开发了可扩展的机器学习模型，用于识别从印度收集的 ATM 交易数据集中的 ATM欺诈交易。我们设计了两种不同的机制，分别适用于流式环境和静态环境。观察发现，收集的数据集高度不平衡。因此，我们采用了多种采样技术，即 SMOTE、

ADASYN 和 GAN。在所有采样技术中，V-GAN 的表现显著优于其他模型。此外，我们在静态和流式环境中采用了多种机器学习技术。在静态场景中，随机森林表现优于其他模型，曲线下面积达到 0.973，其次是曲线下面积相近的梯度提升树。在流式环境中，随机森林排名第一。此外，我们进行了统计检验，结果表明随机森林具有统计显著性。

我们提出的滑动窗口方法有助于降低正确识别 ATM 欺诈交易的风险。不过，我们发现该方法存在一个局限性：每个窗口都假设已收到一定数量的正样本信息。为降低这一风险，在训练模型时，我们需要将一些历史正样本与同一时间戳采集的数据一并提供。否则，任何机器学习技术都无法正确识别欺诈样本。另一种做法是使用单类分类（OCC）模型，而非二分类模型。

参考文献

[1] E.W.T. Ngai, Y. Hu, Yong, Y.H. Wong, Y. Chen, X. Sun. 数据挖掘技术在金融欺诈检测中的应用：分类框架与文献学术综述，《决策支持系统》，50，（2011），559–569。[2] C. Phua, V. Lee, K. Smith, R. Gayler. 基于数据挖掘的欺诈检测研究综合调查，《人工智能评论》（2005）1–14。[3] M.E. Edge, P.R.F. Sampaio. FFML 的设计：一种用于金融数据流主动欺诈管理的基于规则的策略建模语言，《应用专家系统》39，（2012），9966–9985。[4] G.J. Krishna, V. Ravi。 将进化计算应用于客户关系管理：综述，《人工智能工程应用》，（2016），56，30-59。[5] B. Shravan Kumar, V. Ravi. 文本挖掘在金融领域中的应用综述，《基于知识的系统》，114，（20[illegible]。[6] [illegible]数据挖[illegible]4。[7] R. Chen, T. Chen, C. Lin. 一种提高信用卡欺诈检测率[illegible]Dori[illegible], C.S. Cruz. 信用卡业务中的神经网络欺诈检测，《IEEE神经网络汇刊》8，4，（1997），827–834。[9] J.T.S. Quah, M. Sriganesh. 使用计算智能进行实时信用卡欺诈检测，《应用专家系统》35，4，（2008），1721–173。[10] D. Sánchez, M.A. Vila, L. Cerda, J.M. Serrano. 应用于信用卡欺诈检测的关联规则，《应用专家系统》36，2，（2009），3630–3640。[11] A. Srivastava, A. Kundu, A.K. Majumdar. 使用隐马尔可夫模型进行信用卡欺诈检测，《IEEE可靠与安全计算汇刊》5，1，（2008），37–48。[12] V. Zaslavsky, A. Strizhak. 使用自组织映射进行信用卡欺诈检测，《信息与安全》18，（2006），48–63。[13] M. Syeda, Y. Zhang, Y. Pan. 用于快速信用卡欺诈检测的并行粒神经网络，2002，《IEEE国际模糊系统会议》1，（2002），572–577。[14] J. Wang, Y. Liao, T. Tsai, G. Hung. 台湾基于技术的金融欺诈：问题与应对方法，《IEEE系统、人类与控制论会议》，10月，（2006），1120–1124。[15] J. Wei Wei, Li, L. Cao, Y. Ou, J. Chen. 在极度不平衡数据上有效检测复杂的网上银行欺诈，《万维网》，（2012），DOI 10.1007/s11280-012-0178-0。[16] A.S. Arya, V. Ravi, V. Tejasviram, N. Sengupta, N. Kasabov. 使用演化脉冲神经网络进行网络欺诈检测，ICIIS，鲁尔基印度理工学院，印度，2-3，（2016），2016年12月。

[17] S. Kamaruddin, V. Ravi. 使用大数据分析进行信用卡欺诈检测：基于 PSOAANN 的单类分类，载于国际信息学与分析会议（ICIA-16）论文集。计算机协会。纽约，NY，美国，33，（2016），1-8。[18] K. Tayal, V. Ravi. 粒子群优化训练的类关联规则挖掘：在网络钓鱼检测中的应用，载于国际信息学与分析会议（ICIA-16）论文集。计算机协会。纽约，NY，美国，13，（2016），1-8。[19] K. Tayal, V. Ravi. 使用二进制粒子群优化进行模糊关联规则挖掘：在网络欺诈分析中的应用，《IEEE计算智能与计算研究国际会议》（ICCIC2015），（2015），第318-322页。[20] G.G. Sundarkumar, V. Ravi, I. Nwogu, V. Govindaraju. 通过应用程序接口调用、主题模型和机器学习检测恶意软件”，《2015年IEEE自动化科学与工程国际会议》（CASE），2015，（2015），第1212-1217页。[21] G.G. Sundarkumar, V. Ravi. 一种用于挖掘银行业和保险领域不平衡数据集的新型混合欠采样方法，《人工智能工程应用》，37，（2015），368–377。[22] G. Pradeep, V. Ravi, N. Kaushik, B.L. Deekshatulu, [illegible] Sugamaran, P., Das, S.（编）《群体、进化与模因计算》。SEMCCO 2014。《计算机科学讲义》，（2014），8947 [23] G.G. Sundar Kumar, V. Ravi. IEEE计算智能与计算研究国际会议，2[illegible]，（20[illegible]，第1-6页。[24] C. Cortes, v. Vapnik. 支持向量网络。《机器学习》20，（1995），27[illegible]，[illegible] 届国际文档分析与识别会议论文集，1，（1995），278-282，doi:10.[illegible] Rum[illegible]卡：（编）“开行分布式处理：认知微观结构探索》，1。麻省理工学院出[illegible] Rao, [illegible]挖掘技术进行财务报表欺诈检测与特征选择”《决策支持系统》，50，2，（2011），491-500。[28] H. He, Y. Bai, E.A. Garcia, S. Li. ADASYN：用于不平衡学习的自适应合成采样方法，载于《IEEE神经网络国际联合会议》（IEEE计算智能世界大会），（2008），1322-1328。[29] M.A.H. Farquad, V. Ravi, S. Bapi Raju. 使用支持向量机进行银行与金融领域的分析型客户关系管理：一种基于主动学习的改进规则提取方法，《国际电子客户关系管理杂志》，2012，6:1，（2012），48-73。[30] M. Vasu, V. Ravi. 一种用于挖掘不平衡数据集的混合欠采样方法：在银行业和保险领域中的应用，《国际数据挖掘、建模与管理杂志》，2011，3:1，（2010）[illegible]，[illegible]，[illegible]，

[43] H. Kriegel, A. Zimek 等。高维数据中的基于角度的离群点检测。载于 *第14届ACM SIGKDD*知识发现与数据挖掘国际会议论文集，（2008年），444–452。ACM，2008年。[44] J. Hardin, M.R. David。在多聚类场景中使用最小协方差行列式估计量进行异常值检测。计算统计与数据分析，44（4），（2004年），625–638。[45] U. Rajeshwari, B. S. Babu，使用流式分析进行实时信用卡欺诈检测，2016年，第2届应用与理论计算及通信技术国际会议（iCATccT），印度班加罗尔，2016年，第439-444页，doi：10.1109/ICATCCT.2016.7912039。[46] A. Thennakoon, C. Bhagyani, S. Premadasa, S. Mihiranga 和 N. Kuruwitaarachchi，“使用机器学习进行实时信用卡欺诈检测”，2019年，第9届云计算、数据科学与工程国际会议（Confluence），印度诺伊达，2019年，第488-493页，doi：10.1109/CONFLUENCE.2019.8776942。[47] F. Carcillo, A. Dal Pozzolo, Y. Le Borgne, O. Caelen, Y. Mazzer, G. Bontempi，SCARFF：基于 Spark 的流式信用卡欺诈检测可扩展框架。2018年，信息融合，*41*，182-194。https://doi.org/10.1016/j.inffus.2017.09.005[48] S. Mittal, S. Tyagi。实时信用卡欺诈检测计算技术。载于：Gupta, B., Perez, G., Agrawal, D., Gupta, D.（编）（2020年）。计算机网络与网络安全手册。Springer，Cham。https://doi.org/10.1007/978-3-030-22277-2_26[49] M. Arya, H. Sastry，DEAL——用于实时分析的信用卡欺诈检测使用卡欺诈检测。[illegible] Smart Science [illegible]
https://doi.org/10.1080/23080477.2020.1783481[50] Y. Abakarim, M. Lahby, A. Attioui。基于深度学习的高效实时信用卡欺诈检测模型。载于第12届智能系统：理论与应用国际会议论文集（SITA'18）。（2018年）。计算机协会，30，1–7。https://doi.org/10.1145/3289402.3289530