PCCP

ARTICLE TYPE

Cite this: DOI: 00.0000/xxxxxxxxxx

利用深度学习生成模型分析微流控器件中的液滴聚并

Kewei Zhu[a], Sibo Cheng*[b], Nina Kovalchuk[c], Mark Simmons[c], Yi-Ke Guo[b], Omar K. Matar[d], Rossella Arcucci[e]

Received Date
Accepted Date

DOI: 00.0000/xxxxxxxxxx

arXiv:2305.00261v1 [cs.CE] 29 Apr 2023

基于工艺参数预测液滴聚并，对于化学工程中的实验设计至关重要。然而，预测模型可能会受到训练数据不足的影响，更重要的是，还可能面临标签不平衡问题。本研究提出使用深度学习生成模型，通过利用生成的合成数据训练预测模型来解决这一瓶颈。我们为带标签的表格数据开发了一种名为双空间条件变分自编码器（DSCVAE）的新型生成模型。通过在潜在空间和原始空间中同时引入标签约束，与标准条件变分自编码器（CVAE）相比，DSCVAE 能够生成一致且逼真的样本。我们利用合成数据增强了两种预测模型，即随机森林和梯度提升分类器，并在真实实验数据上评估了它们的性能。数值结果表明，使用合成数据可显著提高预测准确率，且所提出的 DSCVAE 明显优于标准 CVAE。本研究为分类问题中处理不平衡数据提供了更深入的见解，尤其是在化学工程领域。

1 引言

液滴聚并在工业中具有重要意义，因为它决定了乳液的稳定性。人们投入了大量精力，致力于理解和预测不同流体动力学条件下的聚并现象；许多研究则根据具体应用，力求将聚并的可能性降至最低或提高到最大。工业上通常需要避免聚并的情形包括：预测并延长泡沫／乳液的保质期（温和的流体动力学条件），或确保产品在运输过程中的稳定性（更具挑战性的流体动力学条件，例如摇晃）。另一方面，在分离过程中，聚并则是有利的，例如去除污染油品的水滴。

微流控为在严格控制的流体动力学条件下研究液滴聚并提供了独特的机会，并可在必要时追踪每个形成的液滴对的去向。微流控平台的使用使研究人员能够以极少的材料用量和能耗探索数千次聚并事件。液滴聚并在微流控中可用于触发或终止液滴内的化学反应，或用于细胞筛选。根据研究目的，会采用不同设计的微流控腔室 [1–4]。

鉴于机器学习（ML）技术已在化学和化学工程领域取得广泛成功，应用范围涵盖从量子化学研究到分子反应动力学，我们认为，有效的机器学习模型也能够预测微流控装置中液滴聚并的概率，从而显著减少微流控设计优化所耗费的时间和成本，包括能源成本，以及光刻过程中使用的昂贵且具有危害性的材料成本。

对于依赖聚并的微流控液滴反应或细胞筛选，液滴聚并率必须接近 100%。在此类情形下，优先考虑的是聚并液滴的组成，而调整连续相性质的可能性通常非常有限。因此，基于几何条件和流体动力学条件准确预测液滴聚并，对于有效的实验设计至关重要。本研究的实验数据集记录了两个液滴在微流控聚并腔中相互作用时发生“聚并”或“非聚并”的结果，具体取决于液滴尺寸和流速等工艺参数。

在此，机器学习可用于优化聚并腔的参数以及微流控器件的整体几何结构，还可优化流动特性，以最大限度地提高液滴聚并概率。从设计上看，不同的机器学习方法

[a] 英国约克大学计算机科学系 [b] 英国伦敦帝国理工学院计算系数据科学研究所 [c] 英国伯明翰大学化学工程学院 [d] 英国伦敦帝国理工学院化学工程系 [e] 英国伦敦帝国理工学院地球科学与工程系

* 通讯作者：*sibo.cheng@imperial.ac.uk*

能够处理化学领域中的各种信息，例如描述现象的图像、记录过程的视频、用于说明的文本以及记录数值的表格数据。例如，研究人员已利用 Lasso 回归和随机森林（RF），根据实验数据预测液滴聚并[8]。此外，神经网络的应用有助于对碰撞合并过程进行参数化[9]，并且研究人员已开发基于深度学习（DL）模型的代理模型，根据实验记录视频预测液滴相互作用的动态过程[10]。在本例中，采用能够准确预测聚并情况的有效机器学习方案，并结合微流控技术，可显著减少配方优化过程中的材料和能源消耗。

尽管机器学习（ML）[11]已在广泛的应用领域取得成功，但人们普遍注意到，不平衡的训练数据会导致预测/分类性能不佳。[12] 化学物质与疾病关系评估[13]以及药物诱导损伤分类[14–17] 中都能找到这样的例子。为解决数据不平衡问题，研究者开发了多种先进的基于树的算法。[18]这些方法主要通过构建由部分数据训练的子模型，来降低数据不平衡导致过拟合的风险。然而，在数据极度不平衡的情况下，由于标签平衡的限制，每个子模型所使用的训练数据集规模有限。

生成模型作为一种数据增强方法应运而生。[19]基于此，它们被用于解决数据不平衡问题。生成对抗网络（GAN）[20]是一种常用的人工数据生成方法：生成器负责生成数据，随后由判别器判断合成数据是否与真实数据相似。尽管生成对抗网络在解决数据不平衡问题方面取得了显著进展，[21], 但其隐式后验分布难以优化，且需要大规模数据才能收敛。[22]另一种生成合成数据的方法是变分自编码器（VAE）。[23]它采用显式后验分布，但无法生成具有类别级条件的数据。为使变分自编码器能够生成特定类别的数据，研究者开发了条件变分自编码器（CVAE），[24] 并在原始空间中添加了一个分类器。得益于这一约束，CVAE 能够同时在原始空间中显式地、在潜在空间中隐式地学习条件表征。借助条件表征，CVAE 能够分别且大量地生成合成数据，从而减轻不平衡数据问题的影响。[25–27]然而，CVAE 的潜在空间仍缺少条件约束。因此，它无法显式学习潜在条件表征，导致更多信息被存储在解码器中，而非潜在空间。[28]Chagot 等人近期的研究 [29] 通过生成合成数据来缓解不平衡数据问题，但潜在条件表征方面的挑战依然存在。

为了提升生成模型的性能，一些研究开始关注潜在空间[30,31]。GET-3D[32] 在三维建模领域并不只约束输出，还提出在潜在空间中增加额外约束，以提高输出的一致性。然而，由于目标任务的特点，GET-3D生成的所有输出都不对应特定标签。例如，GET-3D能够生成逼真的三维汽车，但并没有涉及针对特定标签的约束。GET-3D只是引导生成器生成相似的数据，而不关注它们之间的类别差异。

在本研究中，我们旨在运用机器学习方法解决基于实验的分类问题。本研究基于表格数据开展，这类数据的样本量相对较小，且结果（即“聚并”或“非聚并”）对应的条件值分布非常相似。通过预测微流控器件中的液滴聚并，我们探索了一种解决表格数据不平衡难题的新方法。为完成预测任务，我们采用随机森林和 XGBoost 机器学习方法，因为它们具有可解释性，且擅长处理小样本数据集。为解决数据不平衡问题并提升预测性能，我们采用基于变分自编码器（VAE）的生成模型来获得更多合成数据。在这一阶段，我们提出了双空间条件变分自编码器（DSCVAE），该模型由标准变分自编码器以及分别作用于潜在空间和原始空间的两个显式约束构成。其创新的潜在空间约束引导潜在空间学习条件表征，使 DSCVAE 的潜在空间包含更多信息。利用 DSCVAE 生成更多训练数据后，预测模型的准确率分别提高了 7.5% 和 8.5%。通过基于 Shapley Addictive Explanations（SHAP）值的敏感性分析发现，这些合成数据对预测模型的贡献与微流控实验结果更加一致。

本文其余部分安排如下。第 2 节介绍实验流程以及实验所得的初始数据。第 3 节介绍本任务涉及的机器学习方法，包括两种用于预测聚并结果的基于树的预测算法，以及我们用于生成合成数据的新型 DSCVAE 模型。第 4 节展示数值实验的结果。我们首先介绍实现细节，随后追踪生成模型的训练过程，讨论预测器的性能，并通过 SHAP 分析初始数据集和生成数据集对预测器的影响。

2 实验与数据集

2.1 实验

聚并实验在采用标准软光刻[33]工艺制成的聚二甲基硅氧烷（PDMS）矩形通道微流控装置中进行。水滴（由 Aquatron A 4000 D 蒸馏器制备的双蒸水，Stuart）在硅油（5 cSt, Aldrich）连续相中通过流动聚焦[34]形成，随后在两个对称的交叉结处相遇，并进入图 1 所示的聚并腔。液滴的移动和聚并过程由 Photron SA-5 高速摄像机记录；该摄像机连接至 Nikon Eclipse Ti2-U 倒置显微镜，拍摄速度为每秒 1000 帧，空间分辨率为 2 μm/像素。使用 ImageJ[35] 处理图像，以测定液滴尺寸。

腔室有两个入口或输入通道（图1中的顶部和底部通道）以及两个出口或出口通道（图1中的左侧和右侧通道），宽度为 315 ± 5 μm，深度为 140 ±5 μm。图1b所示的腔室尺寸**L**范围为 901 ± 11μm。腔室内及其相邻

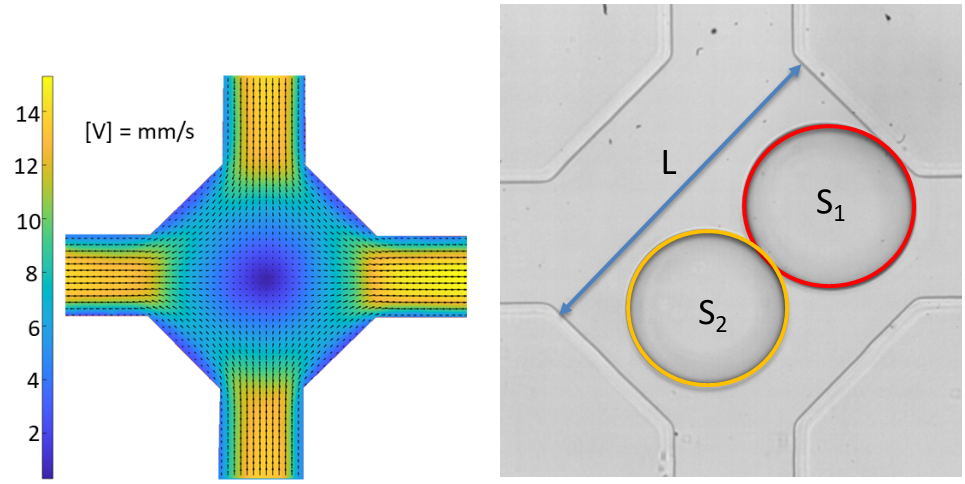


图 1 微流控器件中的液滴聚并实验

通道中的流场如图1a所示，图中以mm/s为单位显示速度大小。该流场采用幽灵粒子测速法进行测量，这是一种非侵入式技术，以小于衍射极限的颗粒散射光产生的散斑图案作为示踪物[36,37]。如图1a所示，由于流动对称性，腔室中心存在一个驻点。液滴以不同的时间间隔进入腔室，这些时间延迟是由注射泵（Al-4000, World Precision Instruments）供给的连续相和分散相的流速波动、通道尺寸不可避免的偏差，以及液滴存在导致的通道阻力波动所引起的。

在理想情况下，液滴沿输入通道的对称轴移动，首先到达腔室的液滴会被困在驻点处，直到第二个液滴到达。实际上，流动波动以及第二个液滴对腔室内流场的影响，会使液滴在驻点周围的不同位置相遇。相遇后，液滴形成液滴对，即开始共同移动。如果液滴对的轴线与输入通道的轴线重合（该轴线也是腔室的一条对称轴），液滴对就会被连续相产生的压缩流困住。在这种情况下，聚并概率为 100%，连续相和分散相的组成只会影响液滴对形成到聚并之间的时间间隔。由于波动始终存在，且实际器件不可避免地存在微小的不对称性，液滴对会开始旋转或平移至某个出口通道；图 1b 展示了一个从初始位置旋转并移动后的液滴对。当液滴对的轴线与出口通道的轴线重合时，其旋转会使其从压缩流区域转移到伸长流区域。伸长流会导致液滴分离，从而发生非聚并事件。

一般而言，液滴相遇的结果取决于连续相从分隔液滴的液膜中排出的时间尺度，以及液滴对从压缩流区域过渡到伸长流区域的时间尺度之间的相对关系：如果液滴对完全不发生旋转，聚并概率为 100%；如果旋转得非常快，聚并概率则为零，即聚并概率会随着流速降低而增加。然而，非常缓慢的流动与高处理量的要求不相容，而高处理量是衡量微流控器件效率的指标。此外，在一定的流速范围内，由于惯性效应，排液时间会随着连续相流速的增加而缩短[38] 从而减弱，甚至可能逆转聚并概率对流速的依赖关系。

因此，有必要进行过程优化，以确定在保持足够高处理量的同时，使聚并概率保持较高的条件。优化参数包括$\mathbf{x}_{\text{flow}}$（总流速）；$\mathbf{x}^{\text{drop1}}$ 和$\mathbf{x}^{\text{drop2}}$（液滴直径）；以及$\mathbf{x}_{\text{dt}}$（液滴之间的时间间隔）。总流速由连续相和分散相的流速之和计算得出，注射泵的输送准确率为 ±1%。液滴尺寸根据其在观测平面中的面积确定，即图1b中的**S1** 和**S2**，最大误差为 ±1%。液滴直径按腔室宽度进行归一化。液滴之间的时间间隔$\mathbf{x}_{\text{dt}}$ 的不确定度为2 ms。

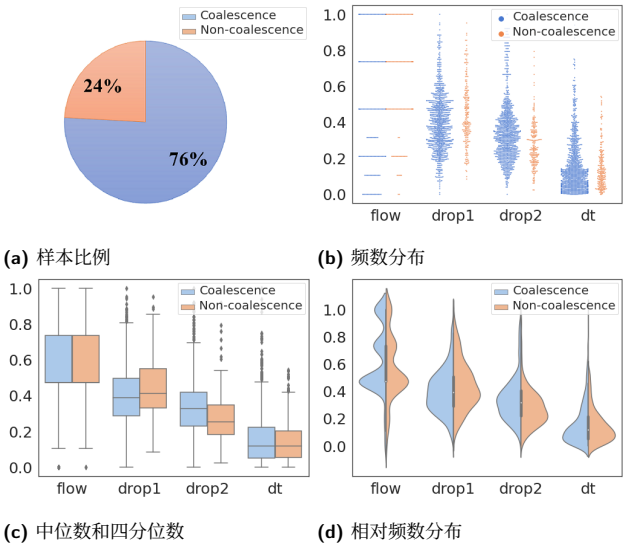
2.2 数据集

实验数据集共包含1531个样本，其中有一个标签**y**，受取值范围各异的四个特征$\mathbf{X}^{\text{features}} = [\mathbf{x}_{\text{flow}}, \mathbf{x}^{\text{drop1}}, \mathbf{x}^{\text{drop2}}, \mathbf{x}_{\text{dt}}]$影响。标签 **y** 有两个类别：“聚并”和“非聚并”。为使特征具有可比性，按照以下方法对其进行最小-最大归一化

$$\mathbf{x}_{\text{scaled}} = \frac{\mathbf{x} - \mathbf{x}_{\text{min}}}{\mathbf{x}_{\text{max}} - \mathbf{x}_{\text{min}}}, \text{ where } \mathbf{x} \in \{\mathbf{x}_{\text{flow}}, \mathbf{x}_{\text{drop1}}, \mathbf{x}_{\text{drop2}}, \mathbf{x}_{\text{dt}}\} \quad (1)$$

其中，最大值和最小值分别为$\mathbf{x}_{\text{max}}$ 和$\mathbf{x}_{\text{min}}$，而归一化结果为$\mathbf{x}$和$\mathbf{x}_{scaled}$。

二分类的统计特征详见图2。图2a显示了不平衡百分比



(a) 样本比例 (b) 频数分布

(c) 中位数和四分位数 (d) 相对频数分布

图2 实验数据集特征

：“聚并”和“非聚并”样本的比例，其中“聚并”占76%，而“非聚并”仅占24%。图2b显示了两个标签下四个特征的数值分布。图2c显示了$\mathbf{X}^{\text{features}}$的中位数、四分位数和须线。这两类标签在每个特征下的这些数值并无显著差异。图2d显示$\mathbf{X}^{\text{features}}$ 不服从标准正态分布，并再次证实不同标签下的分布相似。总之，$\mathbf{X}^{\text{features}}$ 在数量上对立的两个类别之间存在不平衡，但分布相同。换言之，训练数据集中“聚并”样本显著多于“非聚并”样本，而且仅通过观察输入分布很难区分这两类数据。这使预测任务更具挑战性。

机器学习模型必须使用未参与训练的数据进行评估。此外，还需要验证集来调整预测模型的超参数。由于实际中不存在占主导地位的实验结果，因此数据集被划分为平衡测试集和平衡验证集（此处“平衡”指“聚并”和“非聚并”的数量相同），而非按比例划分。其余样本用作训练集。此处的不平衡比率（IR）等于多数类结果（“聚并”）的数量除以少数类结果（“非聚并”）的数量。因此，整个数据集被划分为三个相互独立的集合，即平衡训练数据集、验证数据集和测试数据集，供后续流程使用（见表1）。值得一提的是，本文所有对比模型均在该平衡训练集上进行训练，以避免过拟合。

数据集划分后，不平衡比率（IR）从整个实验集的 3.15 增至训练数据集的 4.62，这使分类任务变得更加困难。

表1 数据集划分

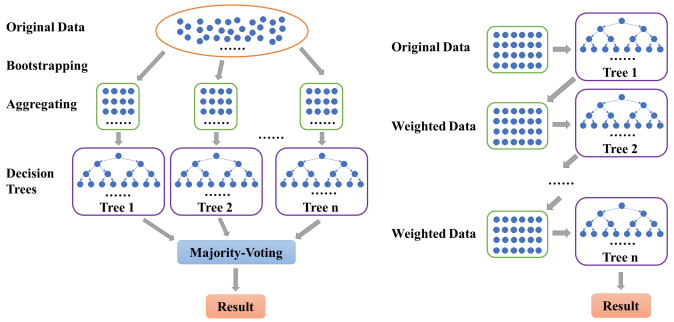
	聚并	非聚并	IR	总计
总数据集	1162	369	3.15	1531
训练数据集	1012	219	4.62	1231
*平衡训练数据集	219	219	1	438
验证集	50	50	1	100
测试数据集	100	100	1	200

3 方法

本节介绍两种先进的基于树的模型——随机森林（RF）和XGBoost，用于预测聚并结果。此外，我们设计了一种变分自编码器（VAE）的变体 DSCVAE，通过生成合成数据来解决上述数据不足和不平衡问题。

3.1 预测模型

基于树的模型属于广泛用于评估表格数据的算法类别，对于小型和中等规模的数据集，其表现优于神经网络[39]。因此，本研究采用这类模型处理小规模数据。决策树（DT）是一种经典的机器学习算法，并具有天然的可解释性。它已用于机器学习模型的分类和回归任务[40,41]。为避免过拟合，应设置一些参数（如最大深度$\mathbf{d}_{\max}$），以增强模型的鲁棒性[42]。然而，细微的变化可能会导致树结构截然不同[43]，，因此采用了若干结合集成学习方法的变体进行预测，例如随机森林（RF，见第 3.1.1 节）和XGBoost（见第 3.1.2 节）。前者和后者分别采用“装袋法”和“提升法”算法。



(a) 随机森林（RF） (b) XGBoost

图 3 预测模型流程图

3.1.1 随机森林

RF 是一种得益于“装袋法”技术、广泛应用于不平衡数据的模型[44] （图3a）。这里，“Bagging”一词是“Bootstrap aggregating”（自助聚合）的缩写，即通过自助采样以并行方式训练多个模型[45,46]。RF 从训练集中有放回地选取不规则比例的样本，并随机选择一些特征用于训练；子模型的数量为 $\mathbf{n}_{\text{estimators}}$。由于子数据集不同，各子模型的结果也会有所差异。随后，通过对分类树进行多数投票来得出预测结果——

这种方法的优点在于，它不仅解决了过拟合问题，还有助于总体上减少预测误差[47]。

3.1.2 XGBoost

不同于子模型并行扩展的“装袋法”，“提升法”是一种串行结构。它已被用于梯度提升决策树（GBDT），以降低树模型的方差和偏差[48]。新树基于前一棵树构建，以减少与目标之间的残差。XGBoost 是 GBDT 的一种高效实现（图3b），基于梯度提升算法[49]。它还具有较强的抗过拟合能力。其重要参数还包括$\mathbf{d}_{\max}$ 和 $\mathbf{n}_{\text{estimators}}$，与随机森林类似。

3.1.3 评估指标

为了从不同角度评估预测性能，并赋予数据特征以可解释性，本研究选用了几种常用方法。

为了介绍二分类的模型性能评估指标，我们采用以下评估指标来评价分类性能，包括从实际角度出发的准确率、精确率（公式2），从预测角度出发的召回率（公式3）、F1分数（公式4）以及混淆矩阵（表2）。

- **混淆矩阵**包含四个类别。这里，真正例（TP）和真负例（TN）表示正确分类的标签；假正例（FP）表示错误预测为正类的数量，而假负例（FN）则表示错误预测为负类的数量。

表 2 混淆矩阵示例

实际\预测	负类	正类
负类	**TN**	**FP**
正类	**FN**	**TP**

- **精确率** 或阳性预测值：

$$\text{Precision} = \frac{\text{TP}}{\text{TP}+\text{FP}} \tag{2}$$

- **召回率**或真正率：

$$\text{Recall} = \frac{\text{TP}}{\text{TP}+\text{FN}} \tag{3}$$

- **F1分数**是精确率和召回率的调和平均数：

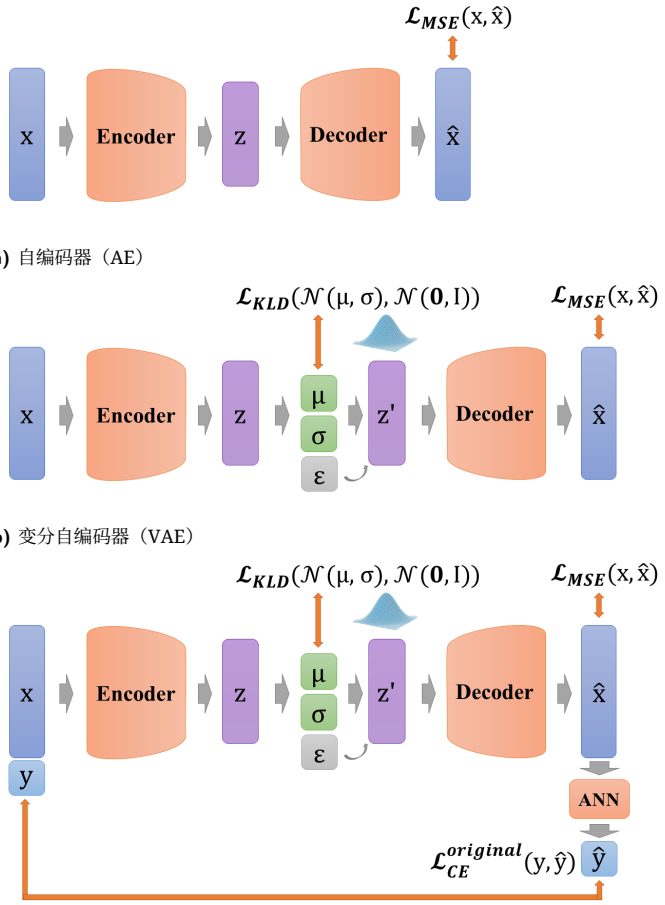
$$\text{F1}_{score} = \frac{2}{\frac{1}{\text{Precision}}+\frac{1}{\text{Recall}}} = 2\times\frac{\text{Precision}\times\text{TPR}}{\text{PPV}+\text{Recall}} \tag{4}$$

SHAP值（SHAP）基于博弈论中的经典夏普利值[50,51]。本研究通过考虑每个特征如何影响树模型，使用其解释特征对预测结果的影响。

3.2 生成模型

两类标签之间的不平衡（图2a、2b）导致树模型的预测产生偏向。此外，平衡数据（即删除多余样本）会减少多数类样本的数量，进而减少样本总数，这可能导致训练数据集不足以供预测模型使用。因此，应通过生成模型增加样本数量。

本研究采用生成模型，即变分自编码器，通过分别生成具有不同标签的数据来解决数据不平衡问题。随后，这些数据被用作预测模型的输入，以提升其性能。



(a) 自编码器（AE）

(b) 变分自编码器（VAE）

(c) 条件变分自编码器（CVAE）

图 4 现有的变分自编码器结构

3.2.1 变分自编码器（VAE）

自编码器（AE）是一种属于无监督学习的前馈神经网络[52]。典型的自编码器采用对称架构，如图4a所示。输入$\mathbf{x}$ 经过编码得到潜变量$\mathbf{z}$，再由其解码得到合成输出 $\hat{\mathbf{x}}$。因此，编码器E 和解码器D 可表示为

$$\mathbf{z} = E(\mathbf{x}) \quad \text{and} \quad \hat{\mathbf{x}} = D(\mathbf{z}) \tag{5}$$

该网络通过最小化 $\hat{\mathbf{x}}$ 与$\mathbf{x}$之间的重构误差$L(\mathbf{x},\hat{\mathbf{x}})$进行训练，通常采用均方误差（MSE）：

$$\mathscr{L}_{MSE}(\mathbf{x},\hat{\mathbf{x}})=\frac{1}{N}\sum_{i=0}^{N}(\mathbf{x}-\hat{\mathbf{x}}_i)^2 \tag{6}$$

该损失体现了数据压缩策略的效率，用于比较模型输出与输入之间的差异。因此，自编码器传统上用于降维[53,54]和特征提取[55]，而非数据生成。

变分自编码器源自自编码器，并开始关注潜在空间中$\mathbf{z}$ 的分布（图4b）[23],，这赋予了变分自编码器生成能力。与自编码器不同，变分自编码器会构建潜在空间中$\mathbf{z}$ 的分布，并从该分布中重新采样$\mathbf{z}'$ 。因此，变分自编码器的整个过程可表示为

$$\mathbf{z}=E(\mathbf{x}) \quad \text{and} \quad \hat{\mathbf{x}}=D(\mathbf{z}') \tag{7}$$

潜在分布$q(\mathbf{z}|\mathbf{x})$经训练后服从后验分布$p(\mathbf{z}|\mathbf{x})$，通常假设该分布为高斯分布。通过 Kullback–Leibler散度（KLD）损失衡量$q(\mathbf{z}|\mathbf{x})$与$p(\mathbf{z}|\mathbf{x})$之间的近似程度，如下所示：

$$\mathscr{L}_{KLD}(q(\mathbf{z}|\mathbf{x})\|p(\mathbf{z}|\mathbf{x}))=-\frac{1}{2}\sum\left(1+\log(\sigma^2)-\mu^2-\sigma^2\right) \tag{8}$$

KLD损失用于正则化潜在空间中隐藏变量$q(\mathbf{z}|\mathbf{x})$的分布，使其收敛至高斯分布$p(\mathbf{z}|\mathbf{x})$。因此，变分自编码器赋予模型可解释性。

重采样的潜变量$\mathbf{z}'$ 可用正态分布来描述，其均值为 μ 、方差为 σ ，记作 $\mathscr{N}(\mu,\sigma^2)$，其中 ε 从高斯分布中采样得到：

$$\mathbf{z}'\approx\mu+\varepsilon\cdot\sigma,\ \text{where}\ \varepsilon\sim\mathscr{N}(0,\mathbf{I}) \tag{9}$$

总之，VAE[56] 的总损失函数表示为

$$\mathscr{L}=\mathscr{L}_{MSE}(\mathbf{x},\hat{\mathbf{x}})+\mathscr{L}_{KLD}(q(\mathbf{z}|\mathbf{x})\|p(\mathbf{z}|\mathbf{x})) \tag{10}$$

VAE 不仅可以像自编码器那样用于降维，还可以利用其潜在状态表示[29]生成与输入不同的“一致”数据。这里，“一致”是指生成数据的统计特性与原始数据集相似。然而，VAE 也是一种无监督学习方法，无法处理特定条件的样本[57] （例如本研究中的“聚并”和“非聚并”），因为编码和解码过程均未涉及条件前提。

3.2.2 条件变分自编码器（CVAE）

为根据先验条件调节潜在分布，研究者设计了 CVAE[24]，以生成包含条件信息的数据。CVAE 将特征输入$\mathbf{x}$ 与辅助条件$\mathbf{y}$拼接，因此成为一种监督学习模型。

此外，在我们的研究中，为防止训练期间发生标签泄漏，标签 $\mathbf{y}$ 不会与$\mathbf{X}^{\text{features}}$ 一同直接输入编码器。换言之，在构建潜变量时，不会将标签 $\mathbf{y}$ 提供给编码器。如图 4c 所示，首先，仅将$\mathbf{X}^{\text{features}}$ 作为编码器的输入。然后，其对应的-

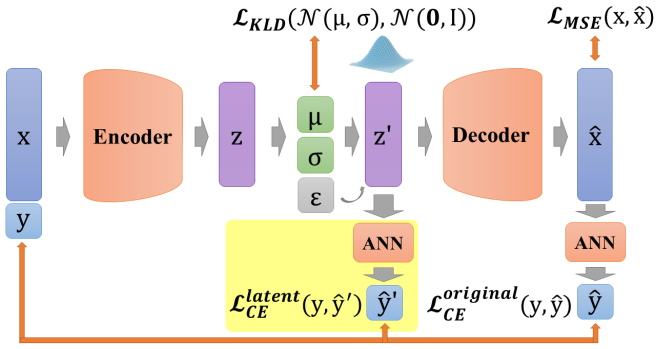


图 5 双空间条件变分自编码器（DSCVAE）

标签 $\mathbf{y}$ 用作人工神经网络分类器（ANN）输出$\hat{\mathbf{y}}$的目标，并将该 ANN 分类器与整个变分自编码器网络联合训练。根据交叉熵（CE）损失，通过预测标签$\hat{\mathbf{y}}\in(0,1)$与概率 $\mathscr{V}$之间的差异进行度量

$$\mathscr{L}_{CE}^{original}=-(\hat{\mathbf{y}}\log\mathscr{V}+(1-\hat{\mathbf{y}})\log(1-\mathscr{V})) \tag{11}$$

因此，总损失可表示为

$$\mathscr{L}=\mathscr{L}_{MSE}(\mathbf{x},\hat{\mathbf{x}})+\mathscr{L}_{KLD}(q(\mathbf{z}|\mathbf{x},\mathbf{y})\|p(\mathbf{z}|\mathbf{x},\mathbf{y}))+\mathscr{L}_{CE}^{original} \tag{12}$$

条件变分自编码器通过条件生成合成数据，并间接改变潜在空间的分布。然而，基于其条件限制，它不足以区分原始空间中非常相似的分布。

3.2.3 双空间条件变分自编码器（DSCVAE）

受 GET-3D[32]，中潜在判别器的启发，我们提出了 DSCVAE。DSCVAE 应是一种更稳健的生成模型，能够精细地区分条件分布之间的细微差异。根据[28]，仅在原始空间施加约束会促使解码器存储更多信息。相比之下，DSCVAE 同时将条件赋予潜在分布和原始空间分布，从而确保生成数据的一致性，这是判定生成模型质量的关键因素。

在 DSCVAE 中，我们首先在两个空间分别采用一个分类器，根据其共同标签对潜在表示 $\mathbf{z}'$ 和原始空间重构 $\hat{\mathbf{x}}$ 进行分类。双空间条件通过为重采样变量 $\mathbf{z}'$ 添加另一个 ANN 分类器来实现（如图 5 所示），其分类误差仍通过交叉熵损失衡量，表示为

$$\mathscr{L}_{CE}^{latent}=-(\hat{\mathbf{y}}'\log\mathscr{U}+(1-\hat{\mathbf{y}}')\log(1-\mathscr{U})) \tag{13}$$

其中，潜在空间预测标签为$\hat{\mathbf{y}}'$，概率为 $\mathscr{U}$。

这两个分类器有助于区分不同标签，而非像传统生成模型那样判断输出之间的相似性。此外，我们提出的新型潜在分类器促使 DSCVAE 学习比仅使用原始空间条件的生成模型更具信息量的潜在空间。因此，DSC-

VAE 应能更好地处理噪声较大且不同标签对应的分布相似的条件输入（此处为“聚并”或“非聚并”）。

总损失函数可写为

$$\mathscr{L} = \mathscr{L}_{MSE}(\mathbf{x},\hat{\mathbf{x}}) + \mathscr{L}_{KLD}(q(\mathbf{z}|\mathbf{x},\mathbf{y})\|p(\mathbf{z}|\mathbf{x},\mathbf{y})) + \mathscr{L}_{CE}^{original} + \mathscr{L}_{CE}^{latent} \tag{14}$$

其中， $\mathscr{L}_{KLD}$ 和 $\mathscr{L}_{CE}^{latent}$都会对潜在分布进行正则化，使潜在空间更具可解释性。

4 数值结果与分析

在本节中，我们比较并分析不同生成方法和预测方法的性能。通过消融测试展示所提出的DSCVAE模型的优势。使用SHAP值作为解释模型性能的指标。

4.1 实现细节

在生成阶段，平衡训练集（如表1所示，以下称为初始数据集）被用作生成模型的输入，同时也用于检查和验证这些模型。在训练生成式神经网络时，参数设置如下：批量大小为73，学习率为 10^{-3}，优化器为Adam，调度器为CosineAnnealingLR[58]，，训练轮次为5000。对于所有生成方法，$\mathbf{z}'$ 都会从该分布中重新采样，并与随机生成的四维高斯噪声相加，以进行正则化。将所得之和输入解码器，以获得输出$\hat{\mathbf{x}}$。不同标签的样本分别生成。总计从每个生成模型中生成6570个平衡样本。

然后，我们使用预测模型评估生成的合成数据。训练数据集列于表3，验证集和测试集也列于表1。

表3 训练数据集

	初始	合成	总计
初始数据集	438	-	438
条件变分自编码器混合数据集	438	438×15	7008
DSCVAE混合数据集	438	438×15	7008

用于评估的算法为随机森林（见第 3.1.1 节）和 XGBoost（见第 3.1.2 节）。关键超参数 $\mathbf{n}_{\text{estimators}}$ 和 $\mathbf{d}_{\max}$ 通过网格搜索选取。所有数值实验均在 Google Colab 平台上完成。深度学习生成模型在单个 Tesla K80 GPU 上运行。

4.2 生成模型的训练损失

图 6 绘出了原始空间和潜在空间中的均方误差、Kullback–Leibler散度以及分类器损失。均方误差损失和 Kullback–Leibler散度损失是构建变分自编码器的基本损失函数（图 4b）。如图 6a 所示，两种模型的均方误差损失均趋于收敛。图 6b 显示，条件变分自编码器与 DSCVAE 的 Kullback–Leibler散度损失曲线存在显著差异。分类器用于

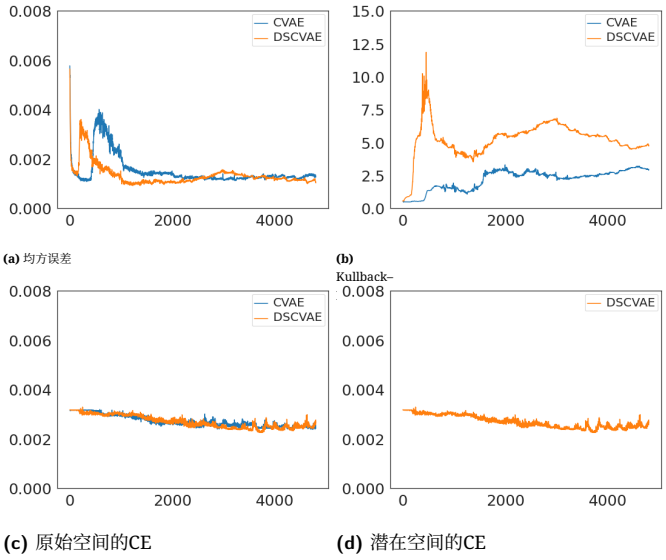


图6 生成模型的损失

区分不同条件（即标签“聚并”或“非聚并”）。图 6c 显示了条件变分自编码器和 DSCVAE 在原始空间中针对解码器输出设置的分类器，而图 6d 显示了 DSCVAE 在潜在空间中针对重采样变量设置的分类器。由于分类器在条件变分自编码器中用作正则化器，其损失均略有下降后趋于稳定。

DSCVAE中潜在空间的分类器直接影响潜在分布的构建，并间接影响原始空间的分类和重构。由于潜在空间分类器会影响潜在分布，因此如图6b所示，DSCVAE的KLD损失比CVAE的更加不稳定。此外，双空间分类器使重构训练更快收敛，如图6a所示，DSCVAE的均方误差损失比CVAE的均方误差损失下降得更快。这可能是因为潜在空间的分类器还会指导数据重构。

4.3 超参数调优

机器学习算法的参数化会影响其泛化能力和预测准确率。在本研究中，使用平衡验证数据集来寻找合适的超参数，即$\mathbf{n}_{\text{estimators}}$ 和$\mathbf{d}_{\max}$。如图7所示，深蓝色区域表示验证集中的高预测准确率。对于随机森林预测器，DSCVAE的热图（图7e）具有最大的深色区域，表明在该验证数据集中具有更高的预测准确率。对于XGBoost预测器，与仅使用原始数据集（7b）相比，CVAE（图7d）和DSCVAE（图7f）均能实现显著更准确的预测。总体而言，在验证数据集中，使用DSCVAE合成数据训练的预测模型相较于XGBoost，在随机森林中的提升更为显著。

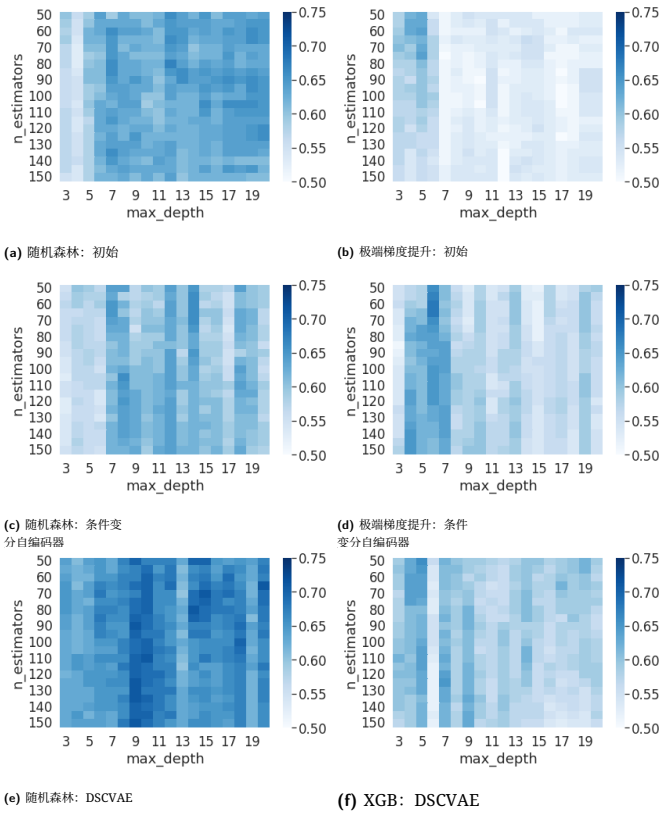
我们固定在验证数据集上表现最佳的$\mathbf{n}_{\text{estimators}}$ 和$\mathbf{d}_{\max}$ （根据图7），因为不应使用测试数据集来调整超参数。这些超参数用于预测测试数据。这些调优后的

表4 超参数调优的验证结果

评估方法	生成方法	准确率（%）	精确率（%）	召回率（%）	F1分数（%）	$n_{estimators}$	d_{max}
RF	None	67.00	67.34	67.00	66.84	80	12
	CVAE	66.00	66.10	66.00	65.95	105	8
	DSCVAE	71.00	71.01	71.00	71.00	125	9
XGB	None	64.00	64.09	64.00	63.94	60	5
	CVAE	68.00	68.12	68.00	67.95	60	6
	DSCVAE	66.00	66.03	66.00	65.99	50	5

表5 使用初始或混合数据集的测试结果

评估方法	生成方法	准确率 (%)	精确率 (%)	召回率 (%)	F1分数（%）
RF	None	58.50	58.57	58.50	58.42
	CVAE	63.00	63.13	63.00	62.91
	条件变分自编码器（L）	60.50	60.59	60.50	60.42
	DSCVAE	**66.00**	**66.42**	**66.00**	**65.78**
XGB	None	58.00	58.01	58.00	57.98
	CVAE	63.00	63.01	63.00	63.00
	条件变分自编码器（L）	61.50	61.59	61.50	61.42
	DSCVAE	**66.50**	**66.58**	**66.50**	**66.46**



(a) 随机森林：初始 (b) 极端梯度提升：初始

(c) 随机森林：条件变分自编码器 (d) 极端梯度提升：条件变分自编码器

(e) 随机森林：DSCVAE (f) XGB：DSCVAE

图7 各种生成模型超参数调优的验证集热图

超参数的确切值如表4所示。

4.4 预测结果

表5展示了基于不同合成训练数据集的随机森林和XGBoost预测结果。为了进一步考察在DSCVAE中联合训练两个分类器的必要性，我们设置了另一个CVAE(L)作为消融实验。CVAE(L)混合数据集由初始数据集和仅在潜在空间中包含一个分类器的CVAE生成的合成数据组成。它与其他三个数据集采用相同的预测模型超参数调优流程。

如表5所示，就所研究的所有评估指标而言，由DSCVAE混合数据集（合成数据集和初始数据集）训练的预测器具有最佳测试结果。事实上，所有生成的数据集都提高了预测模型的准确率。与仅使用初始数据集训练的预测模型相比，DSCVAE使随机森林和XGBoost的预测准确率分别显著提高了7.5%和8.5%。由DSCVAE混合数据集训练的预测模型在精确率、召回率和F1分数方面也表现最佳。此外，DSCVAE有助于减少预测器在验证结果和测试结果之间的过拟合。如表5所示，与初始数据集相比，DSCVAE合成数据将验证准确率与测试准确率之间的差距从8.5%缩小至5%（随机森林），并从6%缩小至 −0.5%（XGBoost）。

表6 RF 和 XGBoost 的混淆矩阵。其中，“TP”和“TN”以及“FP”和“FN”分别对应真正例和真负例预测，以及假正例和假负例预测；而“正类”和“负类”则分别对应聚并事件和非聚并事件。

评估方法	生成方法	验证集 TP	TN	FP	FN	Test TP	TN	FP	FN
RF	None	30	37	20	13	54	63	46	37
	CVAE	31	35	19	15	58	68	42	32
	条件变分自编码器（L）	30	38	20	12	56	65	44	35
	DSCVAE	35	36	15	13	**58**	**74**	42	26
XGB	None	30	34	20	16	56	60	44	40
	CVAE	32	36	18	14	62	64	38	36
	条件变分自编码器（L）	28	38	22	12	57	66	43	34
	DSCVAE	32	34	18	16	**63**	**70**	37	30

我们将“聚并”视为正类样本，将“非聚并”视为负类样本。表6显示了正确预测和错误预测的数量

。与使用初始数据集训练的预测模型相比，两类样本的准确预测数量均有所增加。真正率分别提高了 7.4% 和 12.5%，而随机森林和 XGBoost 的真负率均提高了约 17%。值得一提的是，DSCVAE 的预测准确率为 66%，在预测模型的实际应用中或许仍存在一些局限性。然而，对于二分类问题而言，准确率从 58% 提升到 66% 仍意义重大，并且清楚地展现了所提出生成模型的优势。借助该模型，我们预测每个形成中的双滴会发生“聚并”还是“非聚并”。由于聚并取决于局部工艺参数，液滴聚并概率在 0 到 1 之间变化，因此有些情况无法确定。因此，该模型的可预测性不可能很高。另一方面，可靠地生成大量合成数据，将有助于预测指定数据集的聚并概率。此外，更准确的预测模型还能使可解释性分析更有意义，详见下一节。

4.5 可解释性

此处我们使用 SHAP值来揭示训练数据集中样本的贡献。SHAP值可从全局和局部两个方面进行分析。柱状图（图 8(a-f)）显示了各特征 SHAP值的均值。基于 DSCVAE 混合数据（即合成数据和初始数据）训练的模型，对于不同输入的 SHAP值更为相似。在图 8c 和图 8f 中，与其他方法相比，全部四个特征的 SHAP值彼此相对接近，这表明这些特征对预测模型的贡献相似。例如，图 8b 和图 8e 中的两个特征“drop1”和“drop2”之间存在较大差距，这可能是由条件变分自编码器中的过拟合造成的。而如图 8c、8f 所示，“drop1”和“drop2”的 SHAP值更为相似，因此也更加符合实际。

此外，散点图（图 8（g-l））结合了特征重要性和特征效应。散点图中的每个点都表示某个特征在一个训练样本中的 SHAP值。点在 x 轴上的位置由 SHAP值决定，该值代表特征贡献。散点的颜色表示特征值（红色/蓝色分别表示对“聚并”的正向/负向贡献）。图 8i 和图 8l 中的大多数点聚集在一起，表明它们的 SHAP值较为接近。这意味着每个特征中的单个样本对模型的贡献相似，从而提高了模型的鲁棒性。此外，从图 8（i,l）可以看出，“drop1”和“drop2”的贡献方向相反。也就是说，“drop1”尺寸越大，对“聚并”概率的贡献越负；而“drop2”尺寸越大，对“聚并”概率的贡献越正。事实上，从图 2 可以看出，通常“drop1”的平均尺寸大于“drop2”。此外，当“drop1”和“drop2”的尺寸接近时，液滴发生聚并的概率更高。为进一步验证这一趋势，我们在图 9 中绘制了液滴尺寸差（即$|\mathbf{x}^{\mathrm{drop1}} - \mathbf{x}_{\mathrm{drop2}}|$）对 RF 和 XGBoost 模型预测结果的影响。

可以得出结论，液滴尺寸差越小，液滴聚并的概率就会显著提高。换言之，要在实验中提高液滴聚并概率，应尽量减小液滴尺寸差。这解释了图 8 中“drop1”尺寸和“drop2”尺寸为何具有相反的贡献，也证实了 DSCVAE 的一致性，因为此处的 RF 和 XGBoost 均使用合成数据进行训练。对于在初始数据集和 DSCVAE 数据集上训练的预测模型，可以看出$\mathbf{x}_{\mathrm{dt}}$ 对模型输出的影响有限。值得一提的是，此处我们仅考虑两个液滴形成液滴对的情况。对于较大的$\mathbf{x}_{\mathrm{dt}}$，液滴可能以单滴形式继续运动，不形成液滴对，因此不纳入考量。

总之，在分类任务中训练预测器时，DSCVAE混合数据的表现优于初始数据集和CVAE混合数据。它不仅缩小了验证准确率与测试集准确率之间的过拟合差距，还提升了整体预测性能。通过分析各预测模型的XGBoost值，我们发现DSCVAE混合数据有助于提高特征的同质性，使各特征能够得到平等对待。

5 结论

本研究提出了一种名为双空间条件变分自编码器（DSCVAE）的新型生成模型，用于生成合成表格数据。与传统生成模型相比，该模型利用潜在空间和原始空间中的两个分类器，对数据生成过程进行正则化。我们将DSCVAE模型用于生成合成数据，以训练微流控装置中液滴聚并的预测模型。数值结果表明，在多项评估指标上，预测模型——随机森林和XGBoost——相比仅使用实验数据训练的模型，能够实现显著更准确、更稳健的预测。借助DSCVAE合成数据，随机森林和XGBoost的预测准确率均达到约66%，比仅使用实验数据高约8%。值得指出的是，从预测角度来看，这是一项极具挑战性的任务，原因首先在于数据集不平衡，更重要的是，“聚并”与“非聚并”样本的输入分布相似。这些预测模型有助于优化实验设计；通过计算Shapley Addictive Explanations（SHAP）值，我们也进行了深入分析。结果表明，缩小两种液滴尺寸之间的差距会提高聚并的概率。此外，消融测试表明，与非条件VAE和标准CVAE（仅在潜在空间或原始空间中施加约束）相比，DSCVAE具有更强的性能。

本研究为解决数据量有限且不平衡的表格数据分类问题提供了一种范式，同时也凸显了数据驱动方法在预测微流控液滴聚并方面的潜力。所提出的方法可应用于其他材料和器件。未来工作可进一步提升 DSCVAE 的可解释性和鲁棒性，例如施加物理或化学约束。未来还将致力于扩展模型参数空间，将连续相和分散相的黏度、界面

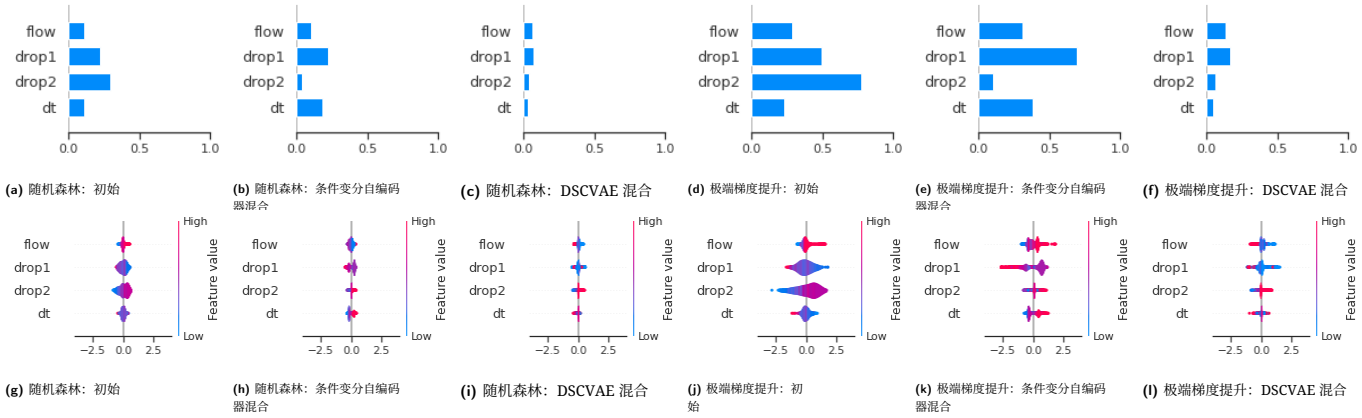


图 8 训练数据集的 SHAP值

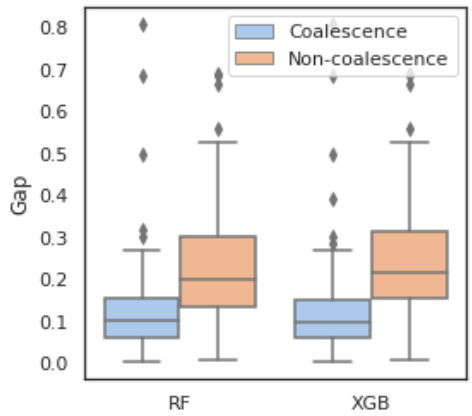


图 9 基于 DSCVAE 增强型预测器的预测标签进行的“间隙”比较

张力、动态表面活性剂效应，以及不同相中存在表面活性剂的情况纳入考量。

代码和数据可用性

本研究的计算部分使用 Python 语言完成。代码可在此处获取：https://github.com/DL-WG/ dscvae-for-drop-coalescence。

如有合理请求，可向 Nina Kovalchuk 博士索取实验数据（n.kovalchuk@bham.ac.uk）

致谢

本研究由 EP/T000414/1 PREdictive Mod- elling withQuantification of UncERtainty for MultiphasE Systems (PREMIERE) 资助。本工作还部分获得了 Leverhulme 野火、环境与社会中心通过 Lever- hulme Trust 提供的支持，资助编号为 RC-2018-023。

缩略词

AE 自编码器。

ANN 人工神经网络。

CE 交叉熵。

条件变分自编码器 条件变分自编码器。

DL 深度学习。

DSCVAE 双空间条件变分自编码器。

决策树 决策树。

假阴性 假负例。

假阳性 假正例。

生成对抗网络 生成对抗网络。

梯度提升决策树 梯度提升决策树。

不平衡比率 不平衡比率。

Kullback–Leibler散度 Kullback–Leibler散度。

机器学习 机器学习。

均方误差 均方误差。

随机森林 随机森林。

SHAP Shapley Addictive Explanations。

TN 真负例。

TP 真正例。

VAE 变分自编码器。

XGB XGBoost。

参考文献

1 T. Krebs, K. Schroen, and R. Boom, “一种用于研究破乳动力学的微流控方法,” Lab on a Chip, vol. 12, no. 6, pp. 1060–1070, 2012. 2 Z. Liu, X. Wang, R. Cao, and Y. Pang, “不同形状的微通道交叉腔室中的液滴聚并,” Soft Matter, vol. 12, no. 26, pp. 5797–5807, 2016. 3 A. Shenoy, C. V. Rao, and C. M. Schroeder, “利用流体技术实现多路粒子操控与组装的 Stokes 阱,” Proceedings of the National Academy of Sciences, vol. 113, no. 15, pp. 3976–3981, 2016. 4 X. Niu, S. Gulati, J. B. Edel, and A. J. demello, “微流控回路中由柱体诱导的液滴合并,” Lab on a Chip, vol. 8, no. 11, pp. 1837–1841, 2008. 5 J. A. Keith, V. Vassilev-Galindo, B. Cheng, S. Chmiela, M. Gastegger, K.-R. Mu"uller, and A. Tkatchenko, “结合机器学习与计算化学，对化学系统进行预测性洞察,” Chemical reviews, vol. 121, no. 16, pp. 9816–9872, 2021. 6 P. O. Dral, “机器学习时代的量子化学,” The journal of physical chemistry letters, vol. 11, no. 6, pp. 2336–2347, 2020. 7 M. Meuwly, “化学反应的机器学习,” Chemical Reviews, vol. 121, no. 16, pp. 10 218–10 239, 2021. 8 E. Wikramanayake and V. Bahadur, “用于冷凝应用的电润湿诱导液滴聚并统计建模,” Colloids and Surfaces A: Physicochemical and Engineering Aspects, vol. 599, p. 124874, 2020. 9 C. F. Rodríguez Genó and L. Alfonso, “利用基函数序列对碰撞—聚并过程进行参数化：采用机器学习方法开发 Colnetv1. 0.0 模型,” Geoscientific Model Development, vol. 15, no. 2, pp. 493–507, 2022. 10 Y. Zhuang, S. Cheng, N. Kovalchuk, M. Simmons, O. K. Matar, Y.-K. Guo, and R. Arcucci, “结合深度学习代理模型的集成潜在同化：应用于微流控装置中的液滴相互作用,” Lab on a Chip, vol. 22, no. 17, pp. 3187–3202, 2022. 11 H. He and E. A. Garcia, “从不平衡数据中学习,” IEEE Transactions on knowledge and data engineering, vol. 21, no. 9, pp. 1263–1284, 2009. 12 S. Mitra, S. Saha, and M. Hasanuzzaman, “一种用于从不平衡数据集中提取化学物质—疾病关系的多视图深度神经网络模型,” IEEE Journal of Biomedical and Health Informatics, vol. 24, no. 11, pp. 3315–3325, 2020. 13 S. Thakkar, M. Chen, H. Fang, Z. Liu, R. Roberts, and W. Tong, “肝毒性知识库（lktb）与药物诱导肝损伤（dili）分类：用于评估大类肝损伤,” Expert review of gastroenterology & hepatology, vol. 12, no. 1, pp. 31–38, 2018. 14 C. Su, S. Ju, Y. Liu, and Z. Yu, “改进随机森林和旋转森林以处理高度不平衡的数据集,” Intelligent Data Analysis, vol. 19, no. 6, pp. 1409–1432, 2015. 15 J. Błaszczy´nski and J. Stefanowski, “邻域采样用于不平衡数据的装袋法”，《Neurocomputing》，第 150 卷，页码 529–542，2015 年。16 M. Bader-El-Den、E. Teitei 和 T. Perry，“用于处理类别不平衡问题的偏置随机森林”，《IEEE transactions on neural networks and learning systems》，第 30 卷，第 7 期，页码 2163–2172，2018 年。17 L. E. B. Ferreira、H. M. Gomes、A. Bifet 和 L. S. Oliveira，“采用重采样处理不平衡数据流的自适应随机森林”，载于《2019 International Joint Conference on Neural Networks (IJCNN)》。IEEE，2019 年，页码 1–6。18 G. E. Batista、R. C. Prati 和 M. C. Monard，“若干平衡机器学习训练数据方法的行为研究”，《ACM SIGKDD explorations newsletter》，第 6 卷，第 1 期，页码 20–29，2004 年。19 J. Wei 和 K. Zou，“EDA：提升文本分类任务性能的简易数据增强技术”，arXiv 预印本 arXiv:1901.11196，页码 6383—6389，2019 年。20 I. Goodfellow、J. Pouget-Abadie、M. Mirza、B. Xu、D. Warde-Farley、S. Ozair、A. Courville 和 Y. Bengio，“生成对抗网络”，《Communications of the ACM》，第 63 卷，第 11 期，页码 139–144，2020 年。21 J. Moon、S. Jung、S. Park 和 E. Hwang，“基于条件表格生成对抗网络的短期负荷预测两阶段数据生成方案”，《IEEE Access》，第 8 卷，页码 205 327–205 339，2020 年。22 R. Burks、K. A. Islam、Y. Lu 和 J. Li，“利用生成模型进行数据增强以改进恶意软件检测：一项比较研究”，载于《2019 IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)》。IEEE，2019 年，页码 0660–0665。23 D. P. Kingma 和 M. Welling，“自编码变分贝叶斯”，《stat》，第 1050 卷，页码 1，2014 年。24 K. Sohn、H. Lee 和 X. Yan，“使用深度条件生成模型学习结构化输出表征”，载于《Advances in Neural Information Processing Systems》，C. Cortes、N. Lawrence、D. Lee、M. Sugiyama 和 R. Garnett 编，第 28 卷。Curran Associates, Inc.，2015 年。 [在线]。获取地址：https://proceedings.neurips.cc/paper/2015/file/8d55a249e6baa5c06772297520da2051-Paper.pdf 25 K. Huang 和 X. Wang，“Ada-incvae：使用变分自编码器改进不平衡分类的数据生成”，《Applied Intelligence》，第 52 卷，第 3 期，页码 2838–2853，2022 年。26 X. Chen、J. Xu、R. Zhou、W. Chen、J. Fang 和 C. Liu，“Trajvae：一种用于轨迹生成的变分自编码器模型”，《Neurocomputing》，第 428 卷，页码 332–339，2021 年。27 Y. Yang、K. Zheng、C. Wu 和 Y. Yang，“通过改进的条件变分自编码器和深度神经网络提升入侵检测的分类效果”，《Sensors》，第 19 卷，第 11 期，页码 2528，2019 年。28 K. He、X. Chen、S. Xie、Y. Li、P. Dollár 和 R. Gir-shick，“掩码自编码器是可扩展的视觉学习器”，载于《Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition》，2022 年，页码 16 000–16 009。29 L. Chagot 、 C. Quilodrán-Casas 、 M. Kalli、N. M. Kovalchuk,

M. J. Simmons、O. K. Matar、R. Arcucci 和 P. Angeli，“流动聚焦微通道中含表面活性剂液滴的尺寸预测：一种数据驱动方法”，《Lab on a Chip》，第 22 卷，第 20 期，页码 3848–3859，2022 年。30 C. Gelada、S. Kumar、J. Buckman、O. Nachum 和 M. G. Belle-mare，“Deepmdp：学习用于表征学习的连续潜在空间模型”，载于《International Conference on Machine Learning》。PMLR，2019 年，页码 2170–2179。31 P. Bojanowski、A. Joulin、D. Lopez-Pas 和 A. Szlam，“优化生成网络的潜在空间”，载于《International Conference on Machine Learning》。PMLR，2018 年，页码 600–609。32 J. Gao、T. Shen、Z. Wang、W. Chen、K. Yin、D. Li、O. Litany、Z. Gojcic 和 S. Fidler，“Get3d：一种从图像中学习高质量三维纹理形状的生成模型”，arXiv 预印本 arXiv:2209.11163，2022 年。33 P. Kim、K. W. Kwon、M. C. Park、S. H. Lee、S. M. Kim 和 K. Y. Suh，“用于微流控的软光刻：综述”，《BIOCHIP JOURNAL》，第 2 卷，第 1 期，页码 1–11，2008 年。34 N. M. Kovalchuk、M. Sagisaka、K. Steponavicius、D. Vigolo 和 M. J. Simmons，“微流控交叉结中的液滴形成：从射流到滴落再到射流的转变”，《Microfluidics and Nanofluidics》，第 23 卷，第 8 期，页码 1–14，2019 年。35 C. A. Schneider、W. S. Rasband 和 K. W. Eliceiri，“从 NIH Image 到 ImageJ：图像分析的 25 年”，《Nature Methods》，第 9 卷，第 7 期，页码 671–675，2012 年。36 S. Buzzaccaro、E. Secchi 和 R. Piazza，“幽灵粒子测速法：利用标准实验室设备实现精准的三维流动可视化”，《Physical Review Letters》，第 111 卷，第 4 期，页码 048101，2013 年。37 N. Kovalchuk、J. Chowdhury、Z. Schofield、D. Vigolo 和 M. Simmons，“利用幽灵粒子测速法研究微通道中的液滴聚并与混合”，《Chemical Engineering Research and Design》，第 132 卷，页码 881–889，2018 年。38 H. Yi、C. Zhu、T. Fu 和 Y. Ma，“对称腔室交叉聚焦微通道中微液滴的高效聚并”，《Journal of the Taiwan Institute of Chemical Engineers》，第 112 卷，页码 52–59，2020 年。39 L. Grinsztajn、E. Oyallon 和 G. Varoquaux，“为什么基于树的模型在表格数据上仍优于深度学习？”，arXiv 预印本 arXiv:2207.08815，2022 年。40 H. H. Patel 和 P. Prajapati，“基于决策树的分类算法研究与分析”，《International Journal of Computer Sciences and Engineering》，第 6 卷，第 10 期，页码 74–78，2018 年。41 H. Gong、S. Cheng、Z. Chen、Q. Li、C. Quilodrán-Casas、D. Xiao 和 R. Arcucci，“基于机器学习 SVD自编码器和广义潜在同化的高效数字孪生，用于核反应堆物理”，《Annals of Nuclear Energy》，第 179 卷，页码 109431，2022 年。42 M. Pal 和 P. M. Mather，“决策树方法在土地覆盖分类中的有效性评估”，《Remote Sensing of Environment》，第 86 卷，第 4 期，页码 554–565，2003 年。43 P. Turney，《偏差与稳定性的量化》，Machine Learning，第20卷，第1期，第23–33页，1995年。44 T. K. Ho，《随机决策森林》，载于《第3届国际文档分析与识别会议论文集》，第1卷。IEEE，1995年，第278–282页。45 L. Breiman，《装袋法预测器》，Machine learning，第24卷，第2期，第123–140页，1996年。46 S. Cheng、Y. Jin、S. P. Harrison、C. Quilodrán-Casas、I. C. Pren- tice、Y.-K. Guo 和 R. Arcucci，《利用机器学习技术进行参数灵活的野火预测：正向与逆向建模》，Remote Sensing，第14卷，第13期，第3228页，2022年。47 L. Breiman，《随机森林》，Machine learning，第45卷，第1期，第5–32页，2001年。48 J. H. Friedman，《贪婪函数逼近：梯度提升机器》，Annals of statistics，第1189–1232页，2001年。49 T. Chen、T. He、M. Benesty、V. Khotilovich、Y. Tang、H. Cho、K. Chen 等，《Xgboost：极端梯度提升》，R 包版本 0.4-2，第1卷，第4期，第1–4页，2015年。50 H. W. Kuhn 和 A. W. Tucker，《博弈论研究贡献》。普林斯顿大学出版社，1953年，第28期。51 S. M. Lundberg、G. Erion、H. Chen、A. DeGrave、J. M. Prutkin、B. Nair、R. Katz、J. Himmelfarb、N. Bansal 和 S.-I. Lee，《借助面向树模型的可解释人工智能，从局部解释走向全局理解》，Nature Machine Intelligence，第2卷，第1期，第2522–5839页，2020年。52 M. A. Kramer，《使用自联想神经网络进行非线性主成分分析》，AIChE journal，第37卷，第2期，第233–243页，1991年。53 S. Cheng、J. Chen、C. Anastasiou、P. Angeli、O. K. Matar、Y.-K. Guo、C. C. Pain 和 R. Arcucci，《利用机器学习代理模型在异质降维空间中进行广义潜变量同化》，Journal of Scientific Computing，第94卷，第1期，第1–37页，2023年。54 S. Cheng、I. C. Prentice、Y. Huang、Y. Jin、Y.-K. Guo 和 R. Arcucci，《基于数据驱动的潜变量数据同化代理模型：在野火预测中的应用》，Journal of Computational Physics，第111302页，2022年。55 M. Yousefi-Azar、V. Varadharajan、L. Hamey 和 U. Tupakula，《基于自编码器的特征学习在网络安全应用中的应用》，载于《2017年国际联合神经网络会议（IJCNN）》。IEEE，2017年，第3854–3861页。56 B. Esmaeili、H. Wu、S. Jain、A. Bozkurt、N. Siddharth、B. Paige、D. H. Brooks、J. Dy 和 J.-W. Meent，《结构化解缠表示》，载于《第22届国际人工智能与统计会议》。PMLR，2019年，第2525–2534页。57 T. Zhao、R. Zhao 和 M. Eskenazi，《使用条件变分自编码器为神经对话模型学习话语层面的多样性》，载于《第55届计算语言学协会年会论文集（第1卷：长篇论文）》，2017年，第654–664页。58 I. Loshchilov 和 F. Hutter，《SGDR：带热重启的随机梯度下降，” arXiv电子印本，第 arXiv–1608 页，2016年。