作品说明书

多源异质大数据联合分析

项目类别｜自然科学类论文　　指导老师｜张逸群、姬玉柱、骆文君
项目成员｜张云帆、陈欣禧、谭泽熙、甘国基、龙志全、高逸菲、蔡升宏、陈俊仰

研究背景　充分开发数据资源，推进智能分析决策，成为重要命题

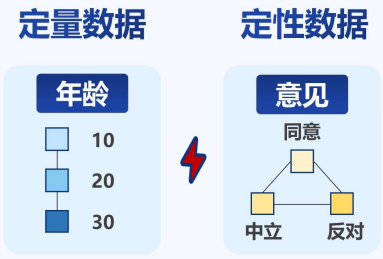
大数据 已成为推动社会进步和经济发展的核心资源

挑战性分析

针对多源数据的异质和标签匮乏难题，创新联邦聚类系统性解决方案

信息融合损耗大

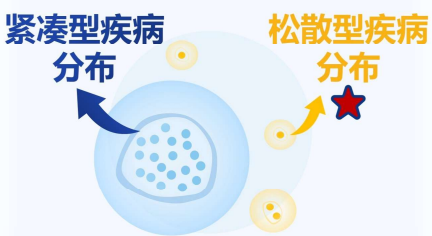
数据异质
对齐程度弱



异质特征数据距离
新度量

联邦聚类精度低

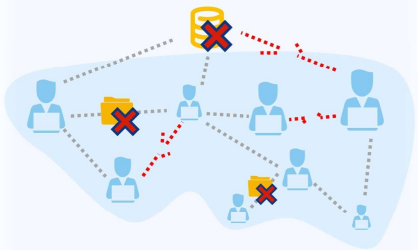
信息非完备
聚类依据匮乏



多源数据联邦聚类
新策略

知识更新时效差

数据动态
更新链路复杂

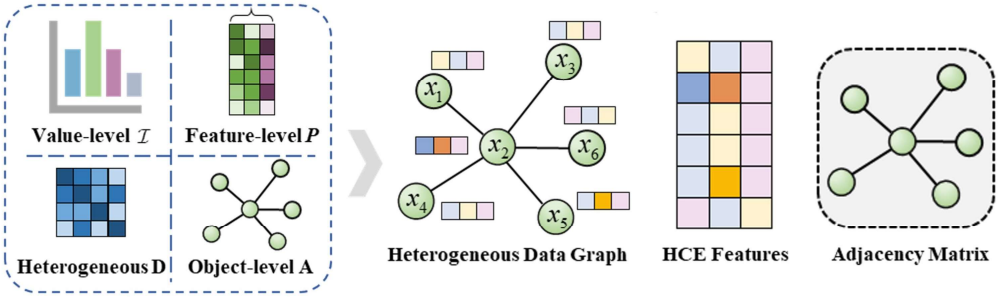


动态环境持续学习
新模型

创新点一　异质特征数据距离新度量　自研统一度量+多层次耦合编码

异质信息提取更充分，分析决策精度高，发表CCF-A类成果

层次耦合编码策略



信息融合损失

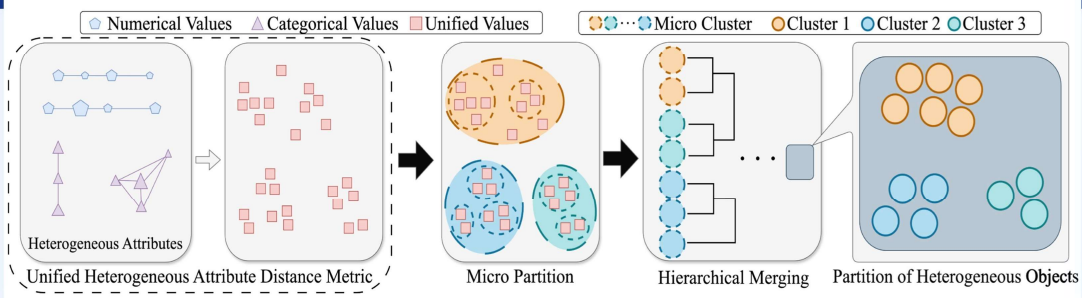
降低高达

21.02%

创新点二　多源数据联邦聚类新策略　多粒度层次脱敏+簇分布自组织

分布检测更细致，知识粒度自适应，发表SCI一区成果

微簇划分+分层合并策略



知识嗅探精度

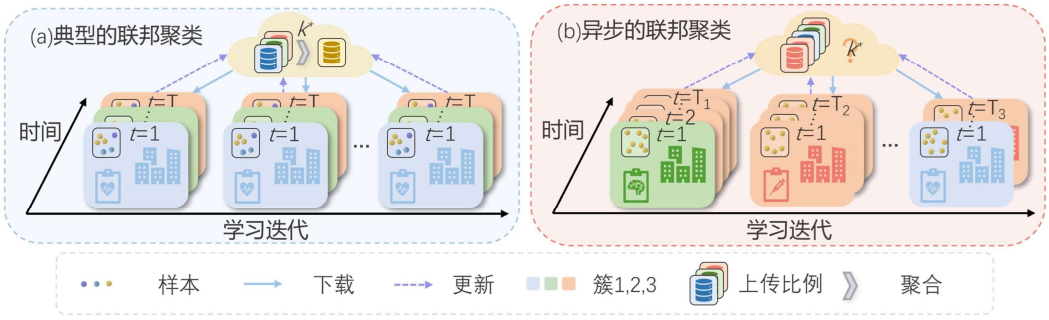
高达

90.05%

创新点三　动态环境持续学习新模型　异步更新平衡+漂移感知网络

分析鲁棒性强，决策精度再提升，发表CCF-A类成果

传统联邦vs异步联邦



知识更新时效

提升高达

74.28%

团队成员

张云帆（负责人）

- 录用SCI一区期刊一篇（一作）
- 录用CCF-A会议一篇（一作）
- 录用CCF-C会议两篇（一作）
- 录用IEEE会议一篇（最佳论文奖）
- 在投SCI一区期刊一篇（二作）
- 获得本科生国家奖学金
- 已被香港高校直博全奖录取

陈俊仰

- 录用SCI一区期刊一篇（一作）
- 录用CCF-A会议一篇（一作）
- 在投SCI一区期刊一篇（一作）
- 保研至清华大学

蔡升宏

- 录用CCF-B会议一篇（一作）
- 在投CCF-A会议一篇（一作）
- 国家级大创项目第一负责人
- 多所港澳高校直博全奖条件录取

陈欣禧

- 录用SCI二区期刊一篇（二作）
- 录用CCF-C会议一篇（二作）
- 在投SCI一区期刊一篇（二作，指导老师一作）
- 国家级大创项目第一负责人

谭泽熙

- 录用SCI一区Top期刊一篇（第一本科生作者）
- 录用SCI二区期刊一篇（三作）
- 录用EI会议论文一篇（一作）
- 国家级大创项目第一负责人
- 广东省级科技创新项目第一负责人

甘国基

- 在投SCI一区期刊一篇
- 在投CCF-B会议一篇
- 实审两项国家发明专利

龙志全

- 在投SCI一区期刊一篇
- 在投CCF-B会议一篇
- 实审两项国家发明专利

专家推荐

沈卫明

加拿大工程院院士
华中科技大学教授
福耀科技大学讲席教授

推荐理由摘选

"对无监督联邦学习和异质数据分析的研究具有前沿性和前瞻性，项目所提出的方案为解决该领域的关键基础性技术难题提供了创新思路。"

蔡宏民

国家杰出青年科学基金获得者
华南理工大学教授

推荐理由摘选

"成果构成了联邦异质数据分析方法体系，理论性和科学性较强，研究拓展了数据科学和机器学习研究领域。有望提升行业应用中的数据分析效能。"

所获成果

团队已取得论文成果12篇，其中8篇为本科一作
另有在投高水平论文6篇

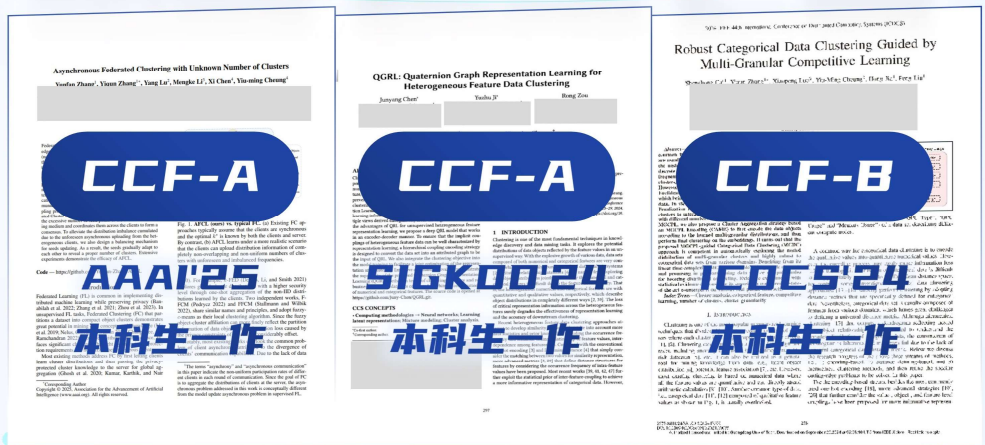
SCI一区
2 篇

CCF-A类
2 篇

CCF-B类
1 篇

CCF-C类
3 篇

IEEE最佳论文奖
1 篇



所获论文成果列表

No.	名称	
1	Yiqun Zhang,..., Zexi Tan, Xiaopeng Luo, Yuzhu Ji*, Rong Zou Learning SelfGrowth Maps for Fast and Accurate Imbalanced Streaming Data Clustering, *TNNLS*, 2025. 探索动态数据分布自适应的高精度高效率非平衡数据聚类算法	SCI一区 IF: 8.9
2	Junyang Chen, Guoheng Huang,..., and Zhixin Huang. Quaternion Cross-Modality Spatial Learning for Multi-Modal Medical Image Segmentation, *JBHI*, 2024. 面向医学影像分割应用的距离度量空间学习深度模型	SCI一区 IF: 6.8
3	Yunfan Zhang, Rong Zou, Yiqun Zhang*,..., Kangshun Li Adaptive Micro Partition and Hierarchical Merging for Accurate Mixed Data Clustering, *CAIS*, 2024. 提出数据微簇划分机制，引导高精度异质数据聚类	SIC二区 IF: 4.6
4	Chuyao Zhang, Xinxi Chen, Zexi Tan,..., Yuzhu Ji, Yiqun Zhang* ANDI: ANy-type Attributed Data Imputation via Cluster-Guided Missing Value Inference, *CIS*'24. 聚类引导的异质特征数据缺失值推断与补全	SIC二区 IF: 2.3
5	Yunfan Zhang, Yiqun Zhang* , Yang Lu, Mengke Li, Yiu-ming Cheung Asynchronous Federated Clustering with Unknown Number of Clusters, *AAAI'25*. 本地节点异步传输且合理簇数未知情形下的联邦聚类范式	CCF-A
6	Junyang Chen, Yuzhu Ji , Rong Zou, Yiqun Zhang, Yiu-ming Cheung QGRL: Quaternion Graph Representation Learning for Heterogeneous Feature Data Clustering, *SIGKDD*'24. 引入超复空间深度表征学习机制突破异质特征数据聚类精度瓶颈	CCF-A
7	Shenghong Cai, Yiqun Zhang*, ..., Hong Jia, Peng Liu Robust Categorical Data Clustering Guided by Multi-Granular Competitive Learning, *ICDCS*'24. 拓展多粒度竞争学习理论提升聚类分析鲁棒性	CCF-B
8	Yunfan Zhang, ..., Yiqun Zhang*, Yiu-ming Cheung Towards Unbiased Minimal Cluster Analysis of Categorical-and-Numerical Attribute Data, *ICPR*'24. 提出异质特征数据最小簇理论实现低归纳偏执聚类分析	CCF-C
9	Pengkai Wang†, Yunfan Zhang†, Yiqun Zhang*, ..., Yiu-ming Cheung Clustering by Learning the Ordinal Relationships of Qualitative Attribute Values, *IJCNN*'24. 进行异质特征数据取值顺序关系学习以增强聚类精度	CCF-C
10	Haoyi Xiao, Xinxi Chen, Xiaopeng Luo* , ..., Wei Ai MACL: Metric and Attribute Space Co-Learning for Qualitative Data Clustering , *ICIC*'25. 提出度量空间和属性子空间协同学习机制增强聚类精度	CCF-C
11	Rong Zou, Yunfan Zhang, Yiqun Zhang,..., Yiu-ming Cheung* Federated Clustering with Unknown Number of Clusters, IEEE *DOCS*'24. 针对普遍的真实簇数未知情形提出簇分布自探索联邦聚类算法	EI IEEE 最佳论文奖
12	Chuyao Zhang, Xinxi Chen, Zexi Tan,..., Yuzhu Ji, Yiqun Zhang* ANDI: ANy-type Attributed Data Imputation via Cluster-Guided Missing Value Inference, *CIS*'24. 聚类引导的异质特征数据缺失值推断与补全	EI

注： 团队本科生成员 指导老师 通讯作者*

查新结果

项目名称

多源异质大数据联合分析

查新机构

中国科学院上海科技查新咨询中心
(国家一级)

查新结论

国内外文献检索中
未见相同文献报道

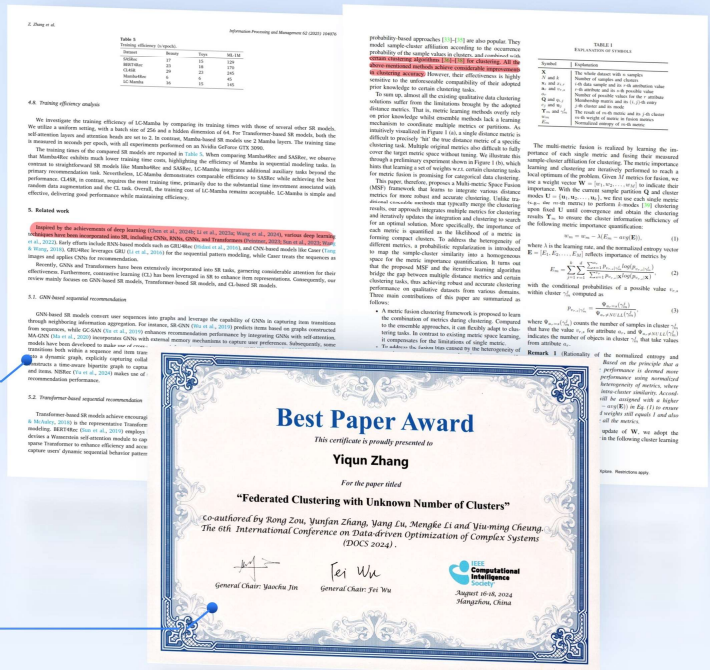
本查新项目具有新颖性

学术认可

论文成果获知名学者

- 香港浸会大学讲席教授张晓明
- 电子科技大学杨波教授

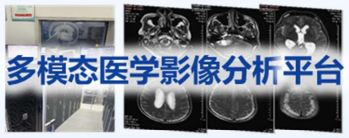
多次正面引用与评价



论文成果获
IEEE最佳论文奖
1项

现实应用价值

- ✓ 与各三甲医院展开广泛合作与应用部署
  （如广州医科大学附属第三医院、深圳市妇幼保健院）
- ✓ 成果价值广受合作单位认可



预处理 + 预分析
医院本地数据

提取 + 融合
复杂临床信息

联合分析
甲方医院-乙方企业

广泛应用拓展

为下游任务提供数据分析与理解方法

数据约简 | 簇数自动判定

将不同模态的医学数据进行聚类

用户画像 | 用户行为表征

增强数据共享，促进国防决策智能化

网络攻击检测|情报分析与整合

结合不同来源的信息，找出异常行为

信用风险评估 | 实时欺诈检测

多源异质大数据联合分析

——系列论文中文汇总

作品类别:	自然科学类学术论文
指导老师:	张逸群、姬玉柱、骆文君
团队成员:	张云帆、陈欣禧、谭泽熙、甘国基
	龙志全、高逸菲、蔡升宏、陈俊仰

前言

本项目总计发表 12 篇相关论文，其中 8 篇为本科生一作，含 SCI 一区 Top 期刊论文 2 篇，CCF-A 类会议论文 2 篇，CCF-B 类会议论文 1 篇，CCF-C 类会议论文 3 篇，EI 会议论文 2 篇，SCI 二区论文 2 篇。为了节省评委的时间，我们将系列论文由英文原文翻译为中文，制作了中文汇总。系列论文以及查新报告等将以附录形式置于论文集后，方便评委查阅。

摘要

聚类是数据挖掘和人工智能领域最重要的数据分析与知识获取工具之一，其旨在探索各种类型的复杂数据分布模式，从而为疾控预警、异常邮件检测、潜在客户群体推荐等诸多下游任务提供核心的数据分析与理解方法。随着全球人口剧增、环境恶化、以及全球交通的日益频密，传染性疾病、各类慢性病、自然灾害等威胁人类健康的问题逐渐凸显，如若防范不当，极易造成巨大的生命健康与财产损失。因此，以疾控和公共卫生安全为例，预警中心的实时监控与快速响应在公共卫生安全中扮演关键角色。现有的疾病与健康监控系统多依赖于传统的单中心集中式数据处理和分析方式。现实场景的疾控大数据往往分布于各个疾控检测节点如各级医院、社康、诊所、地方疾控中心、区域级健康数据库等。而单节点的本地数据也存在指标类型各异但又普遍关联，且不同场景的关注群体分布极不均衡等问题。可见，真实多源大数据分析中普遍存在数据特征异质、分布信息非完备、多源数据质量不齐这一系列相互关联但又相互制约的痛点问题，大大影响了分析决策效果，同时也是本地数据资源的巨大浪费。具体来说，数据中心对本地数据的整合分析能力不理想将影响最终预警和检测的时效性与精确性；而本地节点的指标融合分析效果不佳将加剧非平衡簇检测的效果，从而会向数据中心提供不准确的分析信息，影响数据中心对各个节点数据融合的效果，最终大大影响了聚类分析和数据表征的有效性和准确性。本文围绕复杂场景下的联邦大数据聚类任务开展研究，针对数据特征异质、分布信息非完备、多源数据质量不齐三大关键问题，在现有联邦大数据聚类方法基础上，提出多种能够提升联邦多源大数据聚类性能的新方法。本文主要工作和创新点如下:

- 针对数据多层次的异质问题，包括特征层面、簇分布形态层面以及簇尺度层面的异质，本文提出了**异质特征数据距离度量新方法**，利用概率和熵统一度量，最小化异质特征指标之间存在的信息差异，降低异质的融合信息损失，在此基础上，从“值-特征-异质-样本”四个层次进行编码，深入挖掘隐式信息，实现数据表征增强。通过端到端的机器学习方法，实现下游任务的自适应，确保模型能够灵活应对不同场景需求，实现**高精度的异质特征隐式信息提取**。

- 针对分布信息非完备的问题，提出了**多源数据联邦聚类新策略**，对各节点检测数

据进行知识概括，提取关键信息，随后实施脱敏处理以保护隐私，并通过跨节点知识互补与簇分布推理实现整合分析。同时，系统动态接收并更新各节点数据，确保分析的准确性，最终实现**动态接收并整合分析**。

- 针对动态环境下的联邦学习，知识更新时效性差的问题，提出了**动态环境持续学习新模型**；构建分布漂移敏锐检测网络，通过自增长映射和层次融合策略，实时评估多源数据质量并纠偏，实现**稳健高效的聚类结果增量更新**。

关键词：无监督学习；聚类分析；联邦学习；异质数据；非平衡数据分布；增量学习；持续学习

ABSTRACT

Clustering is one of the most important data analysis and knowledge acquisition tools in the fields of data mining and artificial intelligence. It aims to explore complex data distribution patterns of various types, thereby providing core data analysis and understanding methods for downstream tasks such as disease control early warning, abnormal email detection, and potential customer group recommendation. With the rapid global population growth, environmental degradation, and increasingly frequent global transportation, threats to human health such as infectious diseases, various chronic diseases, and natural disasters have gradually become prominent. If not properly prevented, these threats can easily cause enormous losses in life, health, and property. Therefore, real-time monitoring and rapid response from early warning centers play a critical role in public health security. Existing disease and health monitoring systems mostly rely on traditional single-center centralized data processing and analysis methods. In real-world scenarios, big data for disease control is often distributed across various disease control detection nodes, such as hospitals at all levels, community health centers, clinics, local disease control centers, and regional health databases. Moreover, local data at a single node also exhibits issues such as diverse but generally correlated indicator types, and highly unbalanced distribution of target groups across different scenarios. It is evident that in real multi-source big data analysis, there are a series of interrelated yet mutually restrictive pain points: heterogeneous data features, incomplete distribution information, and uneven quality of multi-source data. These issues significantly affect the effectiveness of analytical decision-making and also result in a huge waste of local data resources. Specifically, the suboptimal integration and analysis capabilities of data centers for local data will impact the timeliness and accuracy of final early warnings and detection; poor indicator fusion analysis at local nodes will exacerbate the effectiveness of unbalanced cluster detection, thereby providing inaccurate analytical information to data centers, which in turn affects the data fusion effect of data centers across nodes, ultimately greatly reducing the effectiveness and accuracy of clustering analysis and data representation. This paper conducts research on federated big data clustering tasks under complex scenarios. Aiming at the three key issues of heterogeneous data features, incomplete

distribution information, and uneven quality of multi-source data, this paper proposes multiple new methods to improve the performance of federated multi-source big data clustering based on existing federated big data clustering methods. The main work and innovations of this paper are as follows:

- To address the problem of heterogeneous data features, a new distance measurement method for heterogeneous feature data is proposed. This method uses unified measurement with probability and entropy to minimize the information differences between heterogeneous feature indicators. On this basis, encoding is performed at four levels: ”value-feature-heterogeneity-sample” to deeply mine implicit information and enhance data representation capabilities. Through end-to-end machine learning methods, adaptive downstream tasks are achieved to ensure that the model can flexibly respond to different scenario requirements and realize high-precision extraction of implicit information from heterogeneous features.

- To address the problem of incomplete distribution information, a new intelligent federated collaborative analysis architecture is proposed. This architecture performs knowledge summarization on the detection data from each node, extracts key information, then implements desensitization processing to protect privacy, and achieves integrated analysis through cross-node knowledge complementarity and cluster distribution reasoning. Meanwhile, the system dynamically receives and updates data from each node to ensure the real-time nature and accuracy of analysis, ultimately realizing dynamic reception and integrated analysis.

- To address the problem of uneven quality of multi-source data, an original continuous federated learning model is proposed. This model constructs a distribution drift-sensitive detection network, uses self-growing mapping and hierarchical fusion strategies to real-time evaluate and correct the quality of multi-source data, and achieves robust and efficient incremental updates of clustering results.

Key words: Unsupervised Learning; Clustering Analysis; Federated Learning; Heterogeneous Attribute Data; Unbalanced Data Distribution; Incremental Learning; Continual Learning

目录

第一章　绪论

本章节将从本工作的背景及研究意义出发，引入现实复杂场景下多源大数据分析所面临的问题。然后，将通过本文的研究内容“面向多源异质数据的联邦聚类分析方法”来解决上述现实挑战。本章节将在最后一部分给出全文的组织架构。

1.1 本课题研究背景及研究意义

1.1.1 研究背景

着数据挖掘与人工智能技术的飞速发展，聚类分析作为发现数据内在结构、洞察潜在规律的核心技术，已广泛应用于医疗诊断、金融风控、精准营销、公共安全等众多领域。传统聚类方法通常依赖于将所有数据集中到一处进行处理，这种中心化的模式在过去发挥了重要作用。然而，进入大数据时代，随着数据规模的爆炸式增长和数据隐私保护意识的日益增强，特别是《通用数据保护条例》（GDPR）等法规的实施，数据集中存储和处理所带来的隐私泄露风险已成为一个严峻的挑战，严重限制了数据价值的进一步挖掘的可能。

为了突破这一瓶颈，联邦学习应运而生，为在保障数据隐私的前提下进行分布式数据分析提供了全新的范式。联邦学习的核心思想是：数据保有方在本地训练模型，仅将模型参数或梯度等中间脱敏信息上传至中心服务器进行聚合，从而避免了原始数据的传输与泄露。将这一范式应用于聚类分析，就称为联邦大数据聚类。它旨在充分利用分散在各机构、设备上的海量数据，充分提取同时严格遵循数据隐私与安全规范。

然而，联邦大数据聚类在走向实际应用的路上并非一帆风顺，其面临的挑战是多维度、深层次的，主要体现在以下几个方面，构成了本课题研究的严峻背景：

（1） 数据固有的异构性与非独立同分布（Non-IID）特性：现实世界中，数据往往分散在不同的机构，其数据分布、特征空间、数据格式乃至语义都可能存在显著差异。例如，不同医院的病历数据、不同银行的交易数据、不同区域的交通数据，都可能呈现出独特的局部模式。这种普遍存在的异构性和 Non-IID 问题，使得传统的、假设数据同分布的聚类算法在联邦场景下难以发挥作用，导致聚类精度和鲁棒性大打折扣。如何有效处理这种**复杂的跨域数据差异**，是联邦聚类面临的首要技术挑战。

(2) 效率与资源约束：尽管联邦学习避免了原始数据的集中传输，但模型参数的频繁交互仍可能带来巨大的通信开销，尤其是在大规模联邦网络或边缘设备受限的场景下。此外，复杂的聚类算法在本地训练时可能需要消耗大量的计算资源。如何在保证聚类性能的同时，**有效优化通信效率、降低计算负担，使其更具可扩展性和实用性**，是联邦大数据聚类亟待解决的工程性难题。

(3) 模型性能与可解释性：在高度分散和异构的数据环境下，联邦聚类模型能否持续输出高精度、高鲁棒性的聚类结果，并有效应对噪声、异常值等干扰，是衡量其有效性的关键。更进一步，对于许多实际应用场景，如金融反欺诈、疾病预测等，用户不仅需要知道“是什么”，更需要了解“为什么”，因此，**提升联邦聚类结果的可解释性**，帮助用户理解聚类决策，辅助业务专家进行判断，同样具有重要意义。

这些挑战相互交织，使得联邦大数据聚类成为当前数据挖掘和人工智能领域的研究热点和难点。

1.1.2 研究意义

课题正是基于上述严峻挑战，以解决多源异质数据的联邦聚类问题为核心，旨在探索新颖的算法和框架，其研究意义体现在以下几个方面：

(1) **推动隐私保护下的大数据智能应用**：本研究将构建一套在数据隐私得到严格保障的前提下，对分布式、异构大数据进行有效聚类分析的理论与方法体系。这将为医疗健康数据共享、金融联合风控、智慧城市管理、跨机构合作等领域提供坚实的技术支撑，打破数据孤岛，加速数据价值的深度挖掘，从而实现社会经济效益的显著提升。

(2) **提升联邦学习与聚类分析的理论与技术深度**：本研究将深入探讨联邦环境下数据异构性、Non-IID 分布等复杂特性对聚类性能的影响机制，并针对性地提出创新性的算法优化策略，如自适应聚合机制、异构特征对齐方法、鲁棒性增强技术等。这将不仅提升联邦聚类算法的精度与鲁棒性，更将丰富联邦学习与聚类分析领域的理论基础，推动交叉学科的融合发展。

(3) **优化系统效能与可扩展性**：针对联邦场景下普遍存在的通信效率和计算资源限制，本研究将探索轻量化模型、分布式优化策略以及高效通信协议，旨在显著降低联邦聚类过程中的资源消耗，提升算法的可扩展性和实时性，使其能更好地适应大规模、动态变化的实际部署环境。

(4) **增强聚类结果的可解释性与实用价值**：本课题还将关注联邦聚类结果的可解释性问题，通过引入适当的机制，使聚类过程和结果更加透明，便于领域专家理解和验证。这将极大地提高联邦聚类模型的实用价值，促进其在关键决策系统中的可靠应用，从而真正将技术优势转化为实际业务成果。

综上所述，本课题的研究工作具有重要的理论价值和深远的实践意义。它不仅致力于解决联邦大数据聚类所面临的核心挑战，为隐私计算背景下的数据协作分析提供新的解决方案，更将为推动人工智能技术在各行业的深度融合与创新发展贡献力量。

1.2 挑战性问题

联邦多源大数据聚类分析是聚类分析领域的一个重要研究方向，但在复杂场景下面临着很多挑战。

(1) 异质数据难以度量：大数据常常存在数据特征异质或者簇分布大小非平衡的问题，数值型数据与类别型数据难以统一度量，大簇和小簇难以得到精确的划分与识别，而现有算法只能针对有限且粗糙的中间数据分布信息进行分析，缺少对数据在特征层面和样本层面进行细致的划分，这会**导致算法易受到误判或漏检**。

(2) 分布信息非完备：现实场景中，本地数据大都是以非独立同分布的形式存在的，但是由于隐私问题难以进行统一的联合分析，因此需要用到联邦数据分析，但是现有的联邦分析技术难以对数据提取关键信息实现对本地节点的概况脱敏，**增加了算法对数据联合分析的难度**。

(3) 动态环境实时性差：在实时环境中，存在着数据漂移以及各联邦节点传输效率不一的情况，这往往会导致整体模型迭代更新的效率下降，更新方向出现偏差的问题，最终**容易出现模型性能下降问题**。

为了解决这些问题，然而当前研究者们采用一些方法在理想的场景下解决多源异质大数据分析的挑战性难题，但在现实场景的数据集下却表现欠佳。包括基于深度学习的

算法、数据增强技术、多尺度特征融合、目标上下文信息利用等（更具体的讨论请见第 二章研究现状综述）。这些策略的综合应用，可以提高多源异质大数据融合分析聚类算法的精度和鲁棒性，为实际应用提供更好的支持。但是，现有工作他们所假设的数据分析场景仍然是简单且局限的。这种简单的应用场景假设阻碍了联邦多源大数据聚类分析的落地应用。为了弥合现有联邦学习与显示应用的差异，本文开展了复杂大数据联邦聚类的相关研究

1.3 研究内容

本文的研究内容可分为三大板块，分别解决上述三大挑战性难题。

（1） **针对异质数据难以度量的问题**，提出了异质特征度量新方法，解决类别型数据以及数值型数据统一度量的难题，将它们进行编码映射，在子空间中度量类别型数据以及数值型数据的距离；同时引入了微簇划分的机制，对簇进行细粒度划分，对微簇进行最小细粒度度量，从而解决了异质数据的统一度量问题。

（2） **针对分布信息非完备的问题**，本文在本地节点提取关键知识信息，再对这些关键信息进行联邦分析，实现本地节点的概况脱敏，既满足了本地节点的隐私性要求，也实现了对本地节点的联合分析，解决了现实场景中分布信息非完备的问题。

（3） **针对动态环境实时性差的问题**，本文通过异步信息传输的方式以及自增长层次合并技术对动态变化的数据进行分析处理，确保各本地节点传输能够以异步的方式进行从而避免等待延迟，并且自增长技术可以使得动态的流式数据能够刻画出簇的形状，再进行层次融合完成聚类分析，解决了动态环境实时性差的问题。

1.4 组织架构

本文共由七章组成。第一章为引言部分，主要介绍了分布式和大数据分析领域中交互类题的研究背景，并分别从联邦分析架构、异质数据度量、非平衡分布簇检测三个经典任务展开讨论; 第二章为研究现状综述，从上面三个任务出发，讨论与多源异质数据的联邦聚类分析任务相关的研究进展，主要包括了传统与先进方法，以及我们的方法相比于这些方法的先进之处; 第三章介绍了上面的第一个任务，异步联邦聚类学习任务。在全局上提出了一个智能联邦协同分析新架构，通过客户端中自适应地学习最优的聚类数。它为客户端统一生成种子点，并通过种子与客户端内对象之间的差异来

积累周围对象的分布信息，从而捕捉客户端自身的分布; 第四章介绍了异质数据度量任务。在第三章的框架下，针对本地数据提出了一个新的多元异质数据综合测度方法，该方法能够有效地融合多模态异质数据，通过图的结构来表示异质数据之间的关系，并提供统一的表征; 第五章介绍了非平衡簇检测任务，在第四章度量的基础上，提出了基于多粒度竞争学习的分类数据聚类方法，专注于解决分类数据聚类难题。方法通过设计多粒度竞争惩罚学习算法，自动学习不同粒度的对象划分，有效揭示了分类数据中复杂的嵌套多粒度簇效应; 第六章对本文进行了总结，并对未来研究方向进行了展望。第七章总结了本文所包含的成果列表。附录 A 展示了第三方论文评价，包括了科技查新包裹，专家推荐，名学者正面引用评价。附录 B 展示了本文论文成果，所获论文奖项和国家发明专利（实审）内容。

第二章　研究现状综述

由于本研究主要解决的是一个系统性的问题，主要涵盖联邦聚类，在未知簇数的情况下聚类，异质特征数据聚类，用于聚类的深度图表示学习，四元数表示学习，分类数据聚类方法和混合数据聚类方法七大类聚类与表征方法，同时我们也将说明所有技术的意义与不同技术之间的联系。

2.1 联邦聚类

最近，一种名为 k-FED[1] 的单次通信联邦聚类（FC）方法被提出，其旨在缓解通信过程中信息泄漏的问题；而 FedKKM 则提出了一种新型的 Lanczos 算法，通过分布式矩阵提升了通信效率。与此同时，F-FCM[2] 和 FFCM[3] 利用模糊聚类技术，通过仅传输对象-聚类隶属关系来增强隐私保护。此外，最近有一种多视角 FC 方法[4] 被提出，旨在通过设计共识原型学习策略，将多视角聚类扩展到联邦学习场景。然而，这些方法都假设客户端和服务器事先知道真实的聚类数量。最近，一种名为 VKMC 的 FC 框架被提出，用于改进基于核心集的垂直联邦学习（FL）；而 HFDPC[5] 则提出了一种基于密度的 FC 方法，通过引入类似密度链的机制，提升数据划分的效果。最新的联邦子空间聚类（Fed-SC)[6] 和联邦谱聚类（FedSC)[7] 方法，分别解决了高维和噪声数据的 FC 问题。尽管上述一些 FC 方法考虑了客户端的非独立同分布（non-IID）或异步性问题，但它们的大多数解决方案仍然严重依赖于‘真实’聚类数量 $k*$ 的可用性，这在实际复杂场景中限制了它们的应用。

2.2 未知簇数情形下的聚类

近年来，更现实的无监督或弱监督学习引起了广泛关注，尤其在一些重要的应用领域[8–11]。聚类是一个关键的无监督学习技术，其中传统的聚类方法需要手动确定最佳的聚类 $k*$[12, 13]。为了实现自动选择 $k*$，基于密度的聚类[14–16] 已被提出，用于在聚类探索过程中通过“膝点”自动确定 $k*$。最近，更先进的基于学习的方法[17, 18] 引入了合作或竞争机制，以避免过多的聚类中心。它们同时确保对象分布的全面表示，并使冗余聚类中心可以被学习，从而实现令人满意的聚类性能。最近，基于显著性的方法[4, 19] 被提出，用来严格判断当前 k 下聚类分布的重要性。然而，所有上述解决方案需要详细的全数据集统计信息，这限制了它们在联邦聚类（FC）中的应用。

最近的异质特征数据聚类方法试图开发更多考虑数据统计和先验知识的相似性度量，包括特征值的出现频率和语义序关系、特征之间的相互依赖性等。得益于深度图卷积网络在揭示图节点关系中的强大能力，基于深度图表示学习的聚类方法引起了广泛关注，并取得了具有竞争力的聚类性能。在图表示学习领域，主流的图卷积网络 (GCN) 及其变体同时表征图结构和特征，以获得更全面的数据表示。随后，图自编码器 (GAE) 及其变体也专门用于无监督图数据的表示学习。

2.3 异质数据聚类

异质数据包括数据特征层面的类别异质，也包括簇大小或分布层面的异质，而现有的异质特征数据聚类方法主要分为两类：1）基于距离度量的聚类，2）基于数值编码的聚类，3）基于竞争学习的聚类。第一类方法利用统计信息优化距离计算，例如通过概率或信息熵衡量相似性，或利用条件概率分布（CPD）反映特征间的依赖关系，并结合语义顺序定义统一距离度量。第二类方法将类别特征转换为数值数据，传统独热编码因忽略耦合关系而被改进，例如利用邻接矩阵编码或结合聚类目标的可学习编码策略。第三类方法将非平衡的簇划分为细粒度足够小的微簇，再对这些微簇进行竞争学习实现层次合并，最终实现对非平衡数据的聚类。然而，这些方法常依赖领域知识，限制了其普适性。编码后的数据可通过传统聚类算法（如 K 均值或谱聚类）进行处理。三类方法各有优势，但均需进一步解决依赖性和普适性问题。

2.4 深度图聚类

受卷积神经网络（CNN）特征提取能力的启发，图卷积网络（GCN）被提出以将卷积操作扩展到图数据，整合图结构与特征信息进行表示学习。Kipf 等人借鉴自编码器（AE）和变分自编码器（VAE）的编码器-解码器框架，提出了 GAE 和 VGAE，通过可学习的方式将输入投影到低维空间并重构图结构，以捕获关键特征。GAE 的变体进一步引入了多种编码增强机制，例如 DAEGC 通过注意力机制整合属性信息与图结构，实现更全面的表示学习；其他工作则将聚类目标融入 GAE 训练过程，或通过 R-GAE 从数学角度缓解噪声特征和特征漂移的影响。尽管这些基于 GCN 的方法在聚类上取得了显著进展，但仍面临图卷积的过度平滑效应和特征异质性问题的挑战。

2.5 四元数表示学习

四元数是一种由四部分组成的超复数，其哈密顿积可视为在正交虚轴空间中的高效旋转。为利用四元数乘积的优越性，研究者将特征编码从实数域扩展至四元数域，以提升特征耦合学习能力。四元数神经网络（QNNs）在少样本分割、图像分类和语音识别等任务中展现出强大的特征提取能力。QCLNet 通过四元数表示学习减轻高维张量的计算负担，并探索查询与支持图像间的交互。其方法将 RGB 图像视为四元数，并利用哈密顿积与可学习权重嵌入，以增强表示学习。凭借正交虚轴和旋转特性，四元数在复杂特征表示学习方面潜力巨大。

第三章　异质特征数据距离新度量

为应对第二章所提到的数据异质特征度量以及非平衡簇分析的新挑战，本章节提出一种新颖的四元数表征学习框架（QRL）。章节依次通过问题引入与本文贡献（3.1）、相关工作综述（3.2）、提出的 GCGQ 方法（3.3）、实验设置与分析（3.4）、本章总结（3.5）深入阐述四元数表征学习框架（QRL）的优势。该框架融合数据的统计先验知识，并创新性地将四元数引入无监督学习范式，实现了对异质特征的高效解耦，从而达成精确的全局距离度量，弥合信息差的效果。

3.1 引言

多元异质数据的综合测度问题一直以来是数据科学和人工智能领域的一个重要研究课题。随着数据类型和数据来源的日益多样化，如何有效地融合来自不同模态（如文本、图像、时间序列、传感器数据等）的异质数据，成为了提升模型性能、优化决策过程的关键问题。传统的数据分析方法大多针对单一数据类型或同质数据进行设计，但在面对复杂的、来自多源的数据时，这些方法常常难以应对。例如，单一模态的数据分析方法在处理文本数据时可能表现良好，但在处理图像或时间序列数据时却效果不佳。此外，异质数据之间的关联性和互补性往往被忽视，导致信息利用不充分，进而影响模型的整体性能。

在本研究中，我们提出了一种新的多元异质数据综合测度方法，该方法能够有效地融合多模态异质数据，并提供统一的表征。首先，在 IJCAI（International Joint Conference on Artificial Intelligence）的研究框架下，我们设计了一种新的图表征方法，通过图的结构来表示异质数据之间的关系。具体而言，我们将每种数据类型视为图中的一个节点，并通过边来表示不同数据类型之间的关联。这种方法不仅能够捕捉到数据内部的复杂关系，还能够有效地处理数据之间的异质性。我们发现，尽管 IJCAI 方法能够提供有效的聚类性能，但其得到的图表征存在一定的局限性，特别是在跨任务的泛化能力方面。例如，在某些情况下，IJCAI 方法在处理特定类型的任务时表现优异，但在面对其他类型的任务时却表现不佳。

基于这一观察，在 KDD（Knowledge Discovery and Data Mining）扩展中，我们提出了一种改进的图表征方法，使其不仅适用于聚类任务，还能够广泛应用于分类、回归等多种数据分析任务。具体来说，我们引入了一种新的图神经网络（Graph Neural Network,

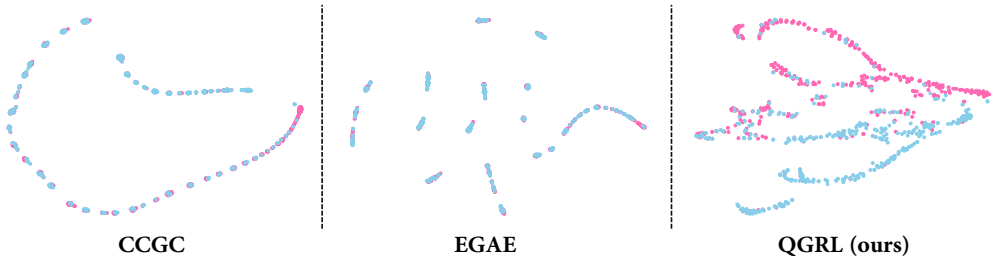


图 3-1 可视化展示了 CCGC、EGAE 和本文提出的 QGRL 在 MM 数据集上的比较

GNN）架构，该架构能够自适应地学习不同数据类型之间的关系，并根据任务需求动态调整图表征。此外，我们还引入了一种多任务学习机制，使得模型能够在多个任务之间共享知识，从而提高模型的泛化能力。实验结果如图3-1表明，我们的方法在多个公开数据集上均取得了显著的效果提升，特别是在处理复杂的多模态数据时，表现尤为突出。我们的贡献概括如下：

- 提出了一个新颖的 QRL 框架，用于准确和鲁棒的异质特征数据聚类。它通过构建图连接了原始异质数据与表示学习之间的信息通道，并通过联合学习方法将表示学习和聚类任务连接起来。
- 为表示学习提供高信息保真度的基础，本文精心设计了一种编码策略，结合了数据的统计先验，包括特征内概率、特征间依赖性以及通过统一在异质特征上的度量计算出的对象间距离。
- 这是首次将四元数引入无监督表示学习。通过我们的模型设计，形成了一种对异质特征数据表示的高效解耦，这对于将四元数应用于其他无监督学习任务也具有重要的参考价值。

3.2 相关工作

最近的异质特征数据聚类方法试图开发更多考虑数据统计和先验知识的相似性度量，包括特征值的出现频率和语义序关系、特征之间的相互依赖性等。得益于深度图卷积网络在揭示图节点关系中的强大能力，基于深度图表示学习的聚类方法引起了广泛关注，并取得了具有竞争力的聚类性能。在图表示学习领域，主流的图卷积网络 (GCN)[20] 及其变体同时表征图结构和特征，以获得更全面的数据表示。随后，图自编码器 (GAE) 及其变体[21] 也专门用于无监督图数据的表示学习。

3.2.1 异质特征数据聚类

现有的异质特征数据聚类方法主要分为两类：1）基于距离度量的聚类，2）基于数值编码的聚类。第一类方法[22],[23],[24] 利用统计信息优化距离计算，例如通过概率或信息熵衡量相似性，或利用条件概率分布（CPD）常被现有方法[25],[26],[27] 用于反映特征间的依赖关系，并结合语义顺序[28] 和方法[29] 定义统一距离度量。第二类方法将类别特征转换为数值数据，传统独热编码因忽略耦合关系而被改进，例如利用邻接矩阵作为编码[30] 或结合聚类目标的可学习编码策略[10]。然而，这些方法常依赖领域知识，限制了其普适性。编码后的数据可通过传统聚类算法（如 K 均值或谱聚类）[31] 进行处理。两类方法各有优势，但均需进一步解决依赖性和普适性问题。

3.2.2 用于聚类的深度图表示学习

受卷积神经网络（CNN）特征提取能力的启发[32]，图卷积网络（GCN）[20] 被提出以将卷积操作扩展到图数据，整合图结构与特征信息进行表示学习。Kipf 等人[33] 借鉴自编码器（AE）[34] 和变分自编码器（VAE）[35] 的编码器-解码器框架，提出了 GAE 和 VGAE，通过可学习的方式将输入投影到低维空间并重构图结构，以捕获关键特征。GAE 的变体[36],[21] 进一步引入了多种编码增强机制，例如 DAEGC[21] 通过注意力机制整合属性信息与图结构，实现更全面的表示学习；其他工作则将聚类目标融入 GAE 训练过程，或通过 R-GAE 从数学角度缓解噪声特征和特征漂移的影响。尽管这些基于 GCN 的方法在聚类上取得了显著进展，但仍面临图卷积的过度平滑效应和特征异质性问题的挑战。

3.2.3 四元数表示学习

四元数是一种由四部分组成的超复数，其哈密顿积可视为在正交虚轴空间中的高效旋转。为利用四元数乘积的优越性，研究者将特征编码从实数域扩展至四元数域，以提升特征耦合学习能力。四元数神经网络（QNNs）[37] 在少样本分割、图像分类和语音识别[38] 等任务中展现出强大的特征提取能力。QCLNet[39] 通过四元数表示学习减轻高维张量的计算负担，并探索查询与支持图像间的交互。其方法[40] 将 RGB 图像视为四元数，并利用哈密顿积与可学习权重嵌入，以增强表示学习。凭借正交虚轴和旋转特性，四元数在复杂特征表示学习方面潜力巨大。

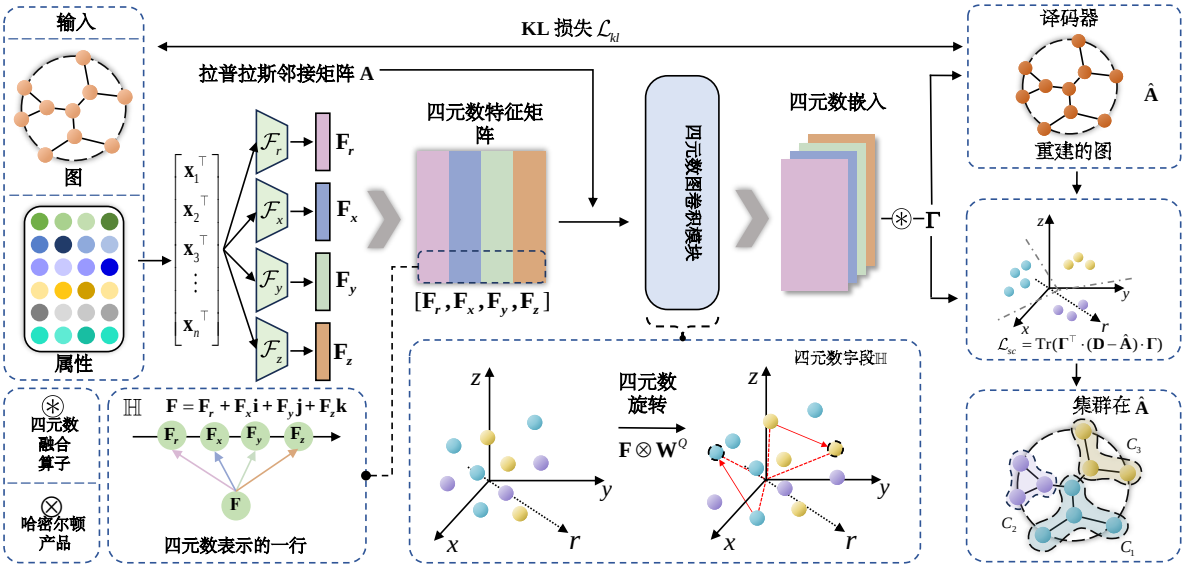


图 3-2 GCGQ 方法运行流程示意图

3.2.4 深度属性图聚类

深度属性图聚类将用属性值描述的连通节点划分为紧凑的聚类，近年来引起了广泛关注。受益于自动编码器 (AE)[34] 和变分自动编码器 (VAE)[35] 强大的表示重构能力，GAE 和 VGAE[33] 提出了使用图卷积算子进行图重构。受到 GAE 成功的启发，最近的研究通过引入注意力机制[21] 和对抗学习机制[36] 进一步改进了它。为了执行更准确的图聚类，最近的一些研究如 EGAE 和[41];Mrabah 提出通过在模型训练期间优化重构和聚类目标来定制表示以使其适合聚类。最近，对比学习[42] 作为一种强大的学习能力增强范式也被引入到图聚类中。它采用聚类作为数据增强的代理任务，并生成更具判别性的节点嵌入。后来，对比图聚类[43] 进一步考虑了一种可学习的可逆扰动恢复代理任务。它在增强中更可靠地保留了语义信息，从而实现了更令人满意的聚类性能。

3.3 提出的 GCGQ 方法

在这一部分中，将详细介绍 GCGQ 的网络结构以及拓展讲一下 QGRL 的结构。首先介绍了系统的总体结构，然后介绍所提出的图聚类方法 GCGQ，该方法的概览如图3-2所示。

3.3.1 GCGQ 的总体架构

如图3-2所示，首先输入系统接受图数据（如邻接矩阵 A）和属性数据（如节点特征矩阵），其中每个节点都有与之相关的属性信息；然后进行四元数特征矩阵构建，将

输入的节点特征矩阵经过处理，得到四元数特征矩阵，其中 F 表示特征矩阵，其中四元数融合操作是通过四元数融合操作对节点的特征进行合并和扩展，将每个节点的多维属性表示成四元数形式。这能够有效地结合不同的特征信息，提升特征耦合的能力；在四元数图卷积模块中，基于四元数的表示，我们利用图卷积神经网络（GCN）对四元数特征矩阵进行卷积处理，从而学习到图的表示。在这一模块中，四元数的每个部分会分别通过卷积层进行处理，并结合图结构的特性进行信息传递。通过这种方式，每个节点的嵌入能够同时捕捉节点特征和图结构之间的关系。然后在学习到的四元数图嵌入后，进行四元数旋转操作，目的是将图的表示从一个空间变换到另一个空间，从而增强图结构的表达能力。在图嵌入和旋转后，接下来通过解码器部分使用学习到的四元数表示来重构图的邻接矩阵；一旦图的嵌入得到优化，就可以进行图聚类任务。图的聚类过程通过对节点的嵌入进行聚类（如使用 k-means 算法等），将节点分配到不同的簇中。其中我们在训练过程中，模型通过最小化 KL 散度损失函数来优化四元数嵌入，这样可以提升图表示的准确性。

3.3.2 广义的四元数表示学习

四视图投影与大多数现有的 QRL 场景不同，数据集具有元组特征组件（例如 RGB 图像），属性图数据具有不同数量的 4Preprint 属性和各种图结构。为了在属性图表示学习中利用 QRL，我们设计了一种可学习的投影机制，将属性 X 投影到四个视图中。这种机制可以解除 QRL 中输入特征的元组限制，还可以利用汉密尔顿积实现属性的有效耦合学习。具体来说，利用四个独立的初始 MLP 将属性

$$\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n]^\top$$

表示的节点投影到四个视图 F_r、F_x、F_y 和 F_z 中，写成如下形式：

$$F_\Delta = \mathcal{F}_\Delta(X) = W_\Delta^L X + B_\Delta^L, \quad \Delta \in \{r, x, y, z\} \tag{3.1}$$

生成的四个视图形成特征四元数的公式如下：

$$F = F_r + F_x i + F_y j + F_z k \tag{3.2}$$

通过引入一个可学习的权重四元数 W^Q，其大小与 F 相同，F 可以基于 W^Q 进行投影，即：

$$F \otimes W^Q = \begin{bmatrix} F_r \\ F_x \\ F_y \\ F_z \end{bmatrix}^T \begin{bmatrix} W_r^Q & W_x^Q & W_y^Q & W_z^Q \\ -W_x^Q & -W_r^Q & W_z^Q & W_y^Q \\ -W_y^Q & W_r^Q & -W_x^Q & W_z^Q \\ -W_z^Q & W_x^Q & W_y^Q & -W_r^Q \end{bmatrix} \tag{3.3}$$

通过调整 W^Q 中的权重，可以高效地转换 F 中的特征并捕捉它们的耦合。

四元数图编码器：为了进一步将特征四元数 F 与图拓扑 A 融合，F 被前馈到由堆叠编码器组成的四元数图卷积模块，其中编码器的操作可以写成：

$$H_l = \varphi_l(\tilde{\mathbf{A}} \cdot H_{l-1} \otimes \mathbf{W}_l^Q) \tag{3.4}$$

每个编码器根据图拓扑汇总节点的 $L-HOP$ 四元素表示，以产生更抽象的级表示 HL。与 M 编码器的四元组图卷积模块的输出嵌入被整合到单个矩阵中：

$$\Gamma = \mathrm{Re}(H_m) \otimes \mathrm{Im}(H_m) \tag{3.5}$$

最后，我们根据嵌入重构：

$$\hat{\Lambda} = \Gamma \cdot \Gamma^{\top} \tag{3.6}$$

每个编码器根据图拓扑 A 聚合节点的四元数表示，以产生更抽象级别的表示 H。这里图重构充当解码器，确保图拓扑的保存。

3.3.3 面向聚类的损失和优化

从模型的宏观角度来看，FVP 和 QGE 模块协同强调 Γ 中的属性信息，而图重构的作用是使 Γ 适应图结构，以寻求平衡的属性和图共识。为了使重构图稀疏以利于图聚类，联合损失函数被设计为图重构项、谱聚类项和正则化项的组合，可以写成:

$$\mathcal{L} = \mathcal{L}_{kl} + \alpha \mathcal{L}_{reg} + \beta \mathcal{L}_{sc} \tag{3.7}$$

重建损失：

$$L_{kl} = \frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} \widetilde{\mathbf{A}}_{ij} \log \frac{1}{\widetilde{\mathbf{A}}_{ij}} \tag{3.8}$$

谱聚类损失项定义为：

$$\mathcal{L}_{sc} = \mathrm{Tr}(\Gamma^{\top} \mathbf{L} \Gamma) \tag{3.9}$$

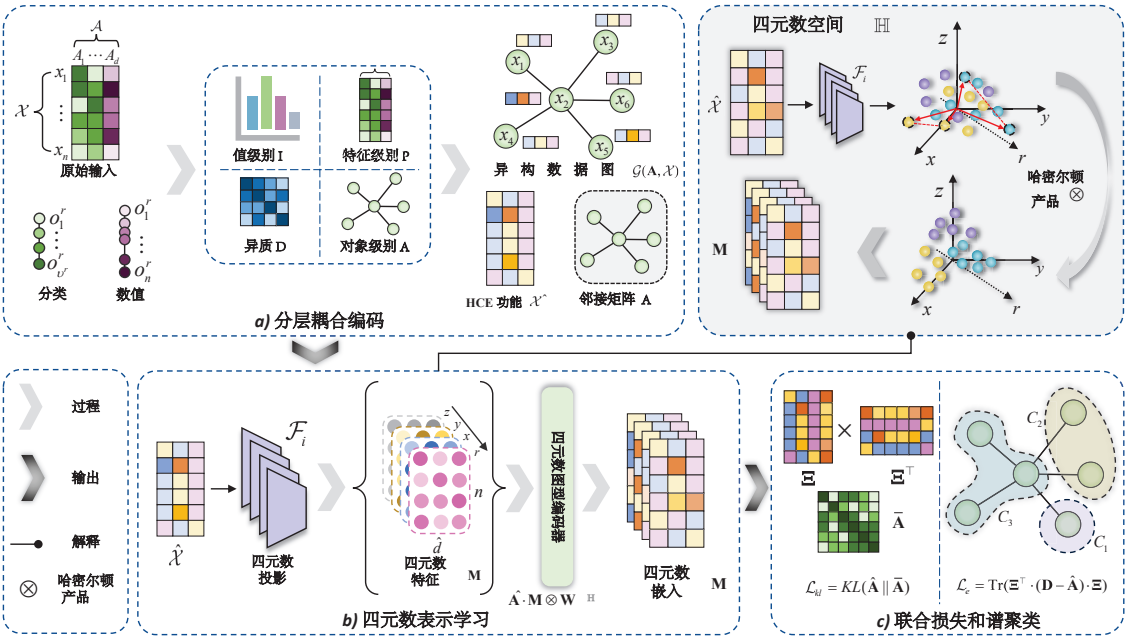


图 3-3 QGRL 方法运行流程示意图

谱聚类目标的优化：

$$\arg\min_{\mathbf{H}} \mathrm{Tr}(\mathbf{H}^{\top}\mathbf{L}\mathbf{H}) \quad \text{s.t.} \quad \mathbf{H}^{\top}\mathbf{H} = \mathbf{I}. \tag{3.10}$$

3.3.4 GCGQ 的拓展-QGRL

四元数图表示学习 (QGRL)：首先在异质特征数据上构建一个图，以捕捉隐含的值级、特征级和对象级耦合，然后引入一个强大的四元数表示学习机制来缓解图表示学习的过度平滑效应。更具体地说，从数据中推导出一个邻接矩阵以形成图结构，称为异质数据图 (HDG)。为了确保图构建的有效性，通过设计的分层耦合编码 (HCE) 策略对数据的多样化统计信息进行编码，以计算邻接矩阵。HDG 充当异质特征数据与后续表示学习之间的信息通道。通过从构建的图中生成四视图编码，四元数表示学习 (QRL) 的哈密顿积可以使得全局特征的高效旋转，为表示学习带来更高的自由度。这弥补了浅层图卷积网络结构的不足，从而减轻了学习到的节点嵌入的过度平滑问题。通过整合图重构和谱聚类损失，模型被促使在生成的四元数潜在空间中学习有利于聚类的表征，方法 QGRL 的架构如图3-3所示。

3.4 实验设置与分析

本节首先概述实验设置，然后展示实验结果并进行讨论。

表 3-1 统计数据

编号	数据集	n	d	k
1	ACM	3025	1870	3
2	WIKI	2405	4973	17
3	CITESEER	3327	3703	6
4	DBLP	4057	334	4
5	FILM	7600	932	5
6	CORNELL	183	1703	5
7	CORA	2708	1433	7
8	WISC	251	1703	5
9	UAT	1190	239	4
10	AMAP	7650	745	8

3.4.1 实验设置

为了更好的展现 GCGQ 的优势，数据集实验在 10 个真实的基准属性图数据集上进行，包括 CORA[44]、CITESEER[44]、DBLP[45]、ACM[45]、WIKI[46]、FILM[47]，以及[47] 的 CORNELL、WISC、UAT、AMAP 四个数据集。CORA 和 DBLP 数据集为引文网络。ACM 和 DBLP 数据集为论文引文关系。WIKI 和 FILM 数据集分别是维基百科链接和电影的关系。CORNELL、WISC、UAT、AMAP 数据集是美国大学网站链接。如下表3-1十个图数据集的统计数据。n 为节点数，d 为属性维数，k 为数据集标签提供的聚类数。

所有实验均在 NVIDIA A5000 GPU、64GB RAM 上的 PyTorch 1.8.0 中实现。我们首先通过 10 轮训练来热身模型，仅使用 KL 损失 Lkl 和正则化损失 $Lreg$。我们遵循最新的图聚类研究[42][41][21][48][42] 来获得聚类性能：每个结果都是在比较方法的十种实现上具有标准差的平均性能。对于每种实现，模型都经过 50 轮训练。在每个 $epoch$ 中，我们对模型进行四次迭代训练，然后进行聚类。选择 50 个 $epoch$ 中最好的聚类性能作为当前实现的性能。

本实验对应设置比较了 11 种聚类方法，包括两种传统方法，即 K-Means[31] 和谱聚类（Spectral-C，以区别于内部评估指标 SC）[49]，两种传统的基于表示学习的聚类方法，即 GAE[33] 和 VGAE[33]，七种最先进的深度聚类方法包括 ARGAE 和 ARVGAE[33]、CONVERT[36]、CCGC[42]、DFCN[43]、DAEGC[21] 和 EGAE[50]。我们让 K-Means 直接对

Algorithm 1 GCGQ: 广义 QRL 的图聚类算法的实现细节

Require: 属性图 $G = \{A, X\}$；聚类数量 k；损失权重 α 和 β。

Ensure: k 个非重叠子图 $\{G_1, G_2, ..., G_k\}$。

1: 将邻接矩阵 A 转换为对称归一化拉普拉斯矩阵 $\hat{A}$；

2: **repeat**

3: 将 X 投影为四个视图 F_b，如公式((3.1)) 所示，并形成特征四元数 F，如公式((3.2)) 所示；

4: 使用公式((3.3)) 和 ((3.4)) 定义的四元数图编码器对 F 进行编码；

5: 通过公式 ((3.5)) 定义的四元数融合操作符获得输出嵌入 Γ；

6: 根据公式 ((3.6)) 从 Γ 重构邻接矩阵 $\hat{A}$；

7: 根据公式 ((3.7)) , ((3.8))和((3.9)) 计算目标函数 $\mathcal{L}$ 的值；

8: 更新可学习参数 W_b^L, B_b^L 和 W_l^Q。

9: **until** 达到最大迭代次数

10: 基于最终 Γ 重构的 $\hat{A}$，执行谱聚类以求解 ((3.10))。

数据属性进行聚类。所有其他方法都首先获取节点表示，然后在表示上实施 K-means。实验中使用了六个评估指标。三个外部指标：聚类准确度 (ACC)、归一化互信息 (NMI) 和平均兰德指数 (ARI)，分别在区间 $[0, 1]$、$[0, 1]$ 和 $[-1, 1]$ 内。三个内部指标：轮廓系数 (SC) 、戴维斯-博尔丁指数 (DBI) 和 Calinski-Harabasz 指数 (CHI) ，不依赖于标签，在区间 $[-1, 1]$、$[0, +\infty)$ 和 $[0, +\infty)$ 内。所有这些指标都被大多数比较的最先进的方法所普遍使用，并且除了 DBI 之外，值越高表示聚类性能越好。如下10为广义 QRL 的图聚类算法的实现细节。

我们进行了四组定量实验：1）比较使用外部指标的聚类性能，以说明 GCGQ 的聚类精度优越性；2）比较不同 ks 下使用内部指标的聚类性能，以验证 GCGQ 学习到的嵌入的可分离性和普适性；3）比较执行时间以验证 GCGQ 的效率；4）比较 GCGQ 的不同简化版本，以证明其核心模块的有效性。

通过外部指标评估的聚类性能表 4.4.2 报告了所有比较方法使用数据标签提供的 k 的聚类性能。从表 4.4.2 可以看出，所提出的 GCGQ 在大多数情况下优于比较方法。在 294 次比较中，GCGQ 获胜了 290 次，这总体上证明了其优越性。请注意，在 AMAP

数据集上实施 ARGAE 和 ARVGAE 时发生了六次“N/A”情况，因为它们遭受了梯度爆炸。下面提供了另外三个关键观察结果：

1）比较方法的性能分为四组：根据“AR”行中的平均排名，有四组方法具有显著的 AR 差距。AR 在 9 左右的 K-Means 和谱聚类（Spectral-C）属于第一组，因为它们是没有表示学习的传统方法。第二组是 GAE、VGAE、ARGAE、ARVGAE、DFCN 和 DAEGC，AR 在 [5.7, 7.3] 之间。它们都基于 GAE，后两者（即 DFCN 和 DAEGC）进一步结合了聚类目标进行训练。第三组是由 CONVERT、CCGC 和 EGAE 组成，它们的 AR 都在 4 左右。所提出的 GCGQ 无疑属于第四组，其 AR 接近于 1。

2）GCGQ 与 CCGC/CONVERT：与表现最佳的同类方法（通常是 CCGC 和 CONVERT）相比，所提出的 GCGQ 实现了巨大的性能改进。具体而言，在 WIKI、DBLP、CORA 和 WISC 数据集上，在 ARI 方面，GCGQ 分别比表现最佳的同类方法高出 16.5%、121.2%、13.2%、29.2%。在大多数其他数据集上，我们的 GCGQ 也比竞争对手实现了约 5% 的显著改进。与我们的 GCGQ 相比，CCGC 和 CONVERT 方法采用了对比学习范式，将 K-Means 视为代理任务。它们的数据增强效果依赖于选择合适的聚类数 k，这是一项不简单的任务，因为数据集标签提供的原始 k 不一定是“真实”k。因此，CCGC 和 CONVERT 在不同数据集上的性能相对不稳定。

3）GCGQ 与 EGAE：EGAE 采用宽松的 K-Means 来优化表示，这也需要适当的聚类数 k，从而实现令人满意的聚类性能。尽管我们的 GCGQ 的训练过程不受 k 的指导，但它在大多数情况下仍然稳定地表现最佳。原因可能是即使原始数据集提供的“真实”k 仍可能不适合融合不一致的属性和图拓扑。给定 k 的使用可以看作是引入了一个强假设，可能会隐式地限制表示学习的拟合能力。相比之下，GCGQ 采用宽松的聚类目标，而不限制节点表示集中在 k 个潜在聚类上。因此，GCGQ 促进了高自由度学习，从而可以获得通用的聚类友好表示。

对于聚类，通常通过内部指标来评估学习到的表示的可分离性。为了验证 GCGQ 广义损失的有效性，我们将 GCGQ 在不同 ks 下的聚类性能与 EGAE，CCGC 和 CONVERT 进行了比较，图3-2中它们是表现更好的最先进的对应方法。比较结果如图3-4所示，可以看出，所提出的 GCGQ 在所有指标下在不同 ks 下始终优于最先进的对应方法。这样的结果同时证明了 GCGQ 学习到的嵌入具有出色的可分离性和通用性。更具体的观

表 3-2 GCGQ 与现有方法相比的聚类性能

Dataset	Metric	K-Means	Spectral-C	GAE	VGAE	ARGAE	ARVGAE	CONVERT	CCGC	DFCN	DAEGC	EGAE	GCGQ (Ours)
ACM	ACC	36.78±0.01	74.21±0.00	44.22±4.11	59.88±1.57	78.56±5.10	86.94±1.37	80.53±2.91	89.26±0.60	86.04±2.18	74.61±10.00	85.54±3.62	**90.37±0.41**
	NMI	00.82±0.01	52.45±0.01	14.67±4.53	18.78±1.13	44.88±7.13	58.20±3.29	47.45±4.35	65.36±1.21	59.66±4.51	47.92±10.35	56.09±8.26	**67.50±1.17**
	ARI	00.24±0.01	47.65±0.00	03.66±2.39	15.59±1.86	46.36±11.01	64.94±3.29	51.30±6.03	71.06±1.37	63.94±4.79	48.70±12.59	62.10±8.21	**73.57±1.06**
WIKI	ACC	25.81±0.89	17.46±0.35	33.11±2.08	31.73±0.75	28.11±1.47	44.47±3.66	51.41±1.15	51.29±0.84	43.10±3.67	25.38±3.35	47.49±1.13	**52.95±0.88**
	NMI	22.69±1.21	08.84±0.16	31.62±1.51	27.25±0.38	23.15±1.94	44.13±2.65	48.46±0.62	46.19±1.01	38.33±2.91	15.15±2.63	43.33±1.99	**49.26±1.32**
	ARI	02.54±0.32	-00.30±0.09	05.61±0.89	15.63±0.79	06.23±1.13	24.44±3.24	28.39±1.33	25.50±2.72	17.17±3.75	07.68±2.25	28.99±1.58	**33.77±0.87**
CITESEER	ACC	26.10±1.33	19.56±0.01	32.93±3.01	55.10±2.19	44.64±7.66	54.37±2.96	62.14±1.53	66.31±2.27	42.37±2.05	42.66±4.74	58.71±3.68	**66.57±1.14**
	NMI	06.92±1.36	00.31±0.00	20.11±2.63	27.92±0.86	19.07±6.89	27.54±2.85	34.68±1.78	**40.45±2.68**	23.90±1.83	18.79±3.56	33.15±2.99	40.36±1.21
	ARI	00.31±1.93	00.08±0.00	04.64±2.01	26.78±1.68	16.07±7.15	25.11±3.66	34.69±1.88	39.12±3.36	19.19±2.43	16.81±4.38	31.46±4.61	**41.43±1.94**
DBLP	ACC	32.74±0.06	29.92±1.01	46.10±1.43	47.07±2.43	55.31±4.93	54.97±6.88	54.52±2.37	54.78±1.97	38.91±0.04	43.36±4.72	53.64±1.46	**72.46±2.24**
	NMI	02.98±0.01	00.28±0.22	19.71±1.83	17.72±2.11	20.63±3.63	22.61±5.44	22.33±1.93	23.81±2.53	08.11±0.04	11.41±3.55	18.19±1.07	**39.12±2.27**
	ARI	15.31±1.87	00.20±2.80	05.78±0.87	14.39±1.95	18.14±4.36	17.70±5.12	17.81±1.17	18.64±1.28	06.63±0.02	10.40±3.70	15.07±2.02	**41.24±2.55**
FILM	ACC	24.21±0.01	24.05±0.04	25.64±0.02	21.40±0.79	23.84±0.47	24.31±1.32	**27.43±0.23**	26.36±0.11	25.91±1.64	24.61±0.33	22.79±0.25	26.81±0.65
	NMI	00.01±0.00	00.11±0.01	00.09±0.01	00.07±0.01	00.16±0.05	00.22±0.39	00.79±0.07	00.15±0.01	00.28±0.03	00.09±0.03	00.21±0.08	**01.47±0.22**
	ARI	00.00±0.01	-00.14±0.02	00.13±0.01	00.01±0.02	00.11±0.03	00.31±0.51	01.34±0.17	00.24±0.05	00.27±0.04	00.15±0.10	00.17±0.08	**01.78±0.26**
CORNELL	ACC	42.40±0.65	37.81±2.34	38.03±1.09	26.66±1.16	36.99±2.54	36.55±2.65	41.86±2.98	39.61±2.09	39.72±1.90	36.28±2.11	39.23±0.53	**38.25±1.84**
	NMI	02.71±0.17	03.69±0.62	05.35±0.36	03.25±0.97	06.01±1.22	03.19±0.56	**09.80±2.68**	04.89±1.04	03.25±0.37	06.83±1.36	06.49±0.73	08.86±2.51
	ARI	-02.14±0.10	-00.15±0.55	02.11±0.48	-00.06±0.54	02.43±1.96	00.85±1.18	**06.19±3.25**	02.07±1.06	-01.10±1.23	02.15±2.08	03.17±0.91	05.48±1.32
CORA	ACC	31.14±3.76	24.47±0.03	49.47±5.76	63.47±0.69	65.96±4.12	66.72±3.04	66.34±1.80	72.00±1.77	45.94±5.80	45.30±5.92	72.11±1.35	**75.82±1.51**
	NMI	06.67±5.28	01.48±0.01	40.86±4.81	45.45±0.59	44.75±3.69	48.96±2.62	46.84±1.68	55.02±1.91	36.46±3.44	25.88±4.35	52.89±1.17	**59.02±1.20**
	ARI	07.83±1.69	-00.08±0.01	22.49±7.27	39.01±0.85	39.52±4.55	42.80±2.69	40.13±1.67	49.17±2.40	23.95±5.52	20.09±5.99	48.49±2.16	**55.64±3.06**
WISC	ACC	42.03±2.04	30.31±0.11	42.11±1.73	25.77±1.34	36.01±2.18	37.17±2.58	**47.61±1.91**	44.14±0.86	40.95±5.44	25.38±3.35	37.01±2.01	44.46±1.58
	NMI	06.25±1.13	03.73±0.01	08.09±0.63	02.60±1.40	11.02±2.61	05.35±3.26	09.70±3.35	08.39±0.45	07.01±0.58	15.15±2.63	11.02±1.16	**16.31±1.66**
	ARI	-03.02±1.68	00.02±0.01	02.85±0.54	00.03±0.48	05.56±1.86	02.02±1.63	04.76±2.69	03.60±0.90	04.45±0.99	07.68±2.25	06.00±1.12	**09.92±1.38**
UAT	ACC	32.69±0.12	32.52±0.01	44.55±0.07	37.45±3.46	49.36±1.30	41.85±1.63	**55.18±1.34**	47.88±2.69	39.33±4.72	52.49±1.25	53.10±0.79	53.84±0.27
	NMI	20.63±0.63	03.43±0.00	18.61±9.44	17.68±0.95	23.33±1.71	15.86±2.38	**27.31±1.18**	20.63±2.64	13.95±1.67	21.42±1.29	21.80±0.85	24.15±1.16
	ARI	06.42±0.44	01.57±0.00	11.61±5.90	14.35±0.84	16.76±0.68	10.33±2.82	19.46±1.90	12.95±1.80	07.28±3.16	21.07±1.11	20.77±0.64	**22.54±0.91**
AMAP	ACC	22.66±0.31	17.24±0.01	60.47±0.87	68.61±0.51	N/A	N/A	66.28±1.86	**77.07±0.38**	58.51±3.96	47.45±3.36	76.37±1.32	76.02±0.94
	NMI	02.37±0.09	00.53±0.00	58.01±0.56	55.04±0.46	N/A	N/A	52.57±0.79	**67.06±0.72**	55.95±1.13	38.83±4.24	65.44±1.61	66.47±1.50
	ARI	00.37±0.03	00.00±0.01	33.31±1.02	46.55±0.67	N/A	N/A	42.89±1.49	57.55±0.44	41.76±1.58	25.03±4.68	57.51±1.74	**57.73±1.50**
-	AR	9.9	10.7	7.8	8.2	7.0	6.4	3.5	3.6	7.0	7.6	4.4	1.5

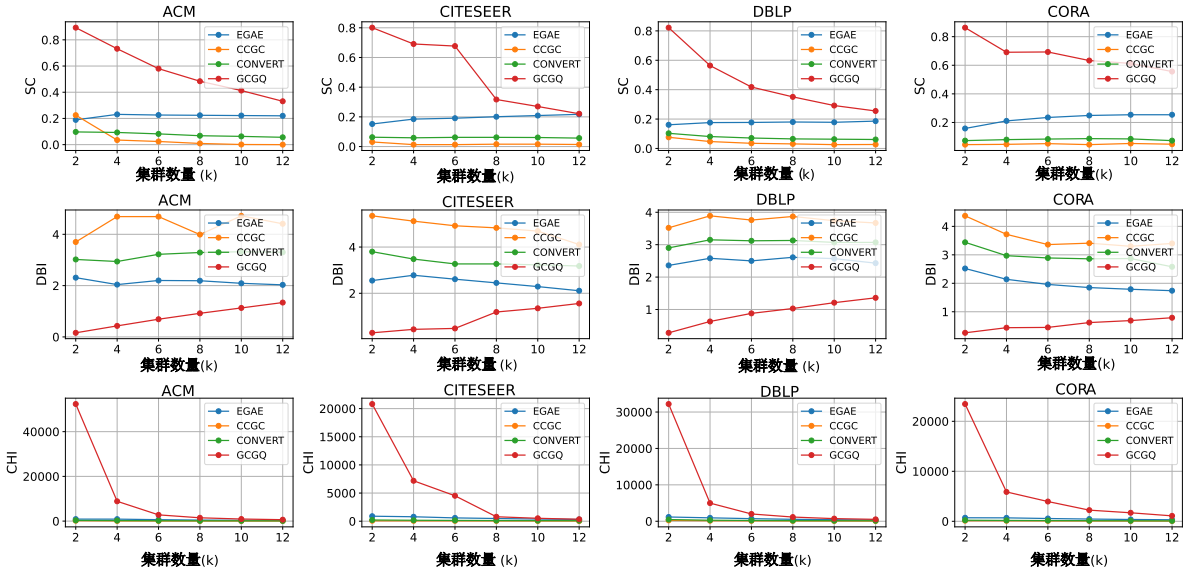


图 3-4 本文方法在不同 ks 下使用内部指标的聚类性能比较

察如下：1）随着 k 值的增加，GCGQ 的性能逐渐下降。这是合理的，因为内部指标主要衡量簇之间的分离度和簇内的紧凑度。当存在许多簇（k 很大）时，指标区分不同表示能力的能力自然会减弱。更极端的情况是，当 $k = n$ 时，所有比较的方法的表现会比较相似；2）图3-4覆盖了数据集标签提供的 ks。在标签提供的 ks 上，GCGQ 的表现优于其他方法，证明了它的表示具有更好的可分离性，从而可以实现更准确的聚类。

对应图3-4所示的前文实验中的四个数据集和三种先进方法，我们还将它们的执行时间与本文提出的 GCGQ 在六个不同 k 值上的平均执行时间进行了比较。执行时间比较结果如图3-5所示。可以看出，比较方法的执行时间高于 GCGQ。这是因为 GCGQ 可以学习通用表示来支持不同 ks 下的聚类而无需重新训练模型，因此其在六次聚类运行中平均的模型训练时间相对较低。相比之下，其他三种方法需要根据指定的 k 对模型进行训练，这会造成更多的训练开销。

3.4.2 消融研究

表格3-3将 GCGQ 的聚类性能与以下模型进行了比较：1）Baseline：由一个 MLP 和两个堆叠 GCN 编码器组成的模型，2）GCGQ w/o FVP：带有冻结 FVP 模块的 GCGQ，3）GCGQ w/o QGE：用传统 GCN 编码器替换其 QGE 模块的 GCGQ。通过将 GCGQ 与 GCGQ w/o FVP、GCGQ w/o QGE 和 Baseline 进行比较，可以分别验证设计的 FVP 的有效性、引入 QGE 的必要性以及 FVP 和 QGE 的适应性。以下提供了三个观察结果：

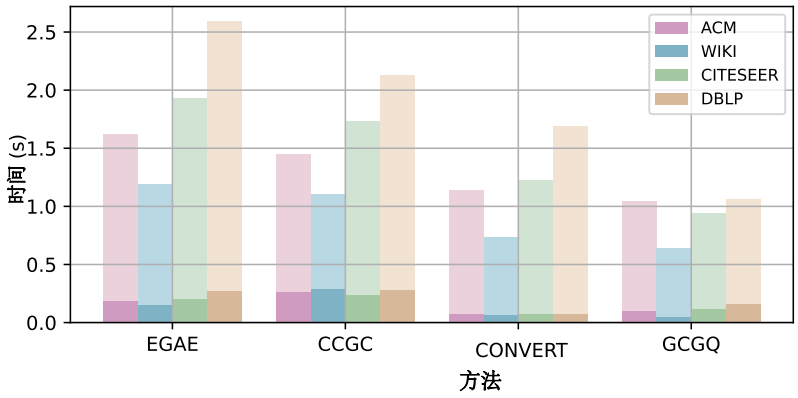


图 3-5 本文方法 GCGQ 与其他方法在不同 k 值上的平均执行时间

表 3-3 GCGQ 关键模块的消融研究

数据集	基线			GCGQ w/o FVP			GCGQ w/o QGE			GCGQ		
	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI
ACM	89.34	64.54	70.92	89.90	65.80	72.27	84.53	56.38	60.85	**90.37**	**67.50**	**73.57**
WIKI	51.60	49.15	32.43	51.57	47.91	31.99	51.64	48.66	32.45	**52.95**	**49.26**	**33.77**
CITESEER	65.73	40.24	40.72	66.21	**40.44**	41.28	66.57	40.38	41.35	**66.57**	40.36	**41.43**
DBLP	67.89	35.20	35.48	71.41	38.15	39.73	67.26	36.59	35.71	**72.46**	**39.12**	**41.24**
FILM	26.95	1.12	1.76	27.41	1.28	1.97	**27.70**	**1.51**	**2.01**	26.81	1.47	1.78
CORNELL	35.85	6.87	3.69	36.99	6.52	4.51	36.61	6.52	3.93	**38.25**	**8.86**	**5.48**
CORA	72.73	55.84	51.44	73.12	55.13	50.61	72.11	55.35	49.68	**75.82**	**59.02**	**55.64**
WISC	40.92	13.54	8.03	43.03	16.09	9.37	**44.74**	**17.21**	**10.45**	44.46	16.31	9.92
UAT	53.68	23.69	21.87	53.71	23.77	22.36	**54.37**	23.26	22.19	53.84	**24.15**	**22.54**
AMAP	73.23	61.13	53.81	74.69	63.65	55.53	74.98	64.34	56.37	**76.02**	**66.47**	**57.73**

1）GCGQ 在 30 次比较中的 29 次中表现优于 Baseline，清楚地说明了 FVP 和 QGE 的适应性。

2）GCGQ 在 30 次比较中的 27 次中表现优于 GCGQ w/o FVP。这显然表明 FVP 是 QGE 的必要前期阶段。GCGQ w/o FVP 使得四个 MLP 无法学习，因此 FVP 退化为输入属性的随机投影，这无疑失去了为 QGE 提供更合适的特征四元数的能力。

3）GCGQ 在 30 次比较中的 22 次中优于 GCGQ w/o QGE。这通常表明 QGE 在聚合节点信息和防止过度支配效应方面是有效的。如果没有 QGE 中的汉密尔顿积，GCGQ w/o QGE 无法促进属性信息的高自由度学习，因此图拓扑可能主导节点信息聚合。也就是说，两个非常不同的连接节点的嵌入在最终的嵌入中可能是同质的，这可能会严重妨碍聚类准确性。

图 3-6 不同方法在 ACM 数据集上的应用效果图

为了直观地展示 GCGQ 的表示效果，我们在图3-6中可视化了最先进的 EGAE、CCGC、CONVERT 和我们的 GCGQ 在 ACM 数据集上生成的嵌入分布。二维图是使用 $t-SNE$ Van der Maaten & Hinton 生成的，我们使用不同的颜色标记标签提供的簇。直观地看，CCGC、CONVERT 和我们的 GCGQ 的表现优于 EGAE，具有更多可分离的簇。原因在于 CCGC 和 CONVERT 采用了对比增强，GCGQ 采用了四元数旋转，有效提升了表示模型的学习能力。由于 GCGQ 对属性的四个视图进行了结构旋转，节点的全局分布得到了更好的保留，因此 GCGQ 的嵌入簇比 CCGC 和 CONVERT 的嵌入簇更易分离。基于 GAE 的 EGAE 方法可能过于受图拓扑支配，因为它没有特别强调属性信息的保留。

3.5 本章总结

本节中提出了一种新的属性图聚类方法 GCGQ。它利用四元数高效汉密尔顿积的优势，同时解决了限制聚类性能的过度平滑和过度支配问题。通过广义设计，形成了一个由可学习的 FVP 和 QGE 组成的表示学习模型，用于聚类友好的表示学习。FVP 模块弥合了任意维度属性与 QGE 的四部分四元数运算之间的差距，这两个模块协同增强了：1）模型的学习能力，2）属性信息的保存。广义聚类目标损失引导模型学习具有高自由度的通用表示，而不限制嵌入集中在预先指定的聚类数量上。因此，GCGQ 可以获得更具判别性和聚类友好的节点表示，这些表示对于不同的 ks 都是一致的。这被认为是实际应用和数据理解的重要优势。大量实验证明了 GCGQ 的优越性。虽然 GCGQ 被证明是有效的，但它并非没有局限性。也就是说，GCGQ 的通用性和效率针对的是静态数据上不同的所需聚类数 k。而我们的方法提到的 KDD 则是 IJCAI 的拓展，提出了一种名为 QGRL 的新型四元数图表示学习方法，用于异构特征数据聚类。为了更全面地学习异构数值和分类特征的表示，我们精心编码了复杂的层次耦合到属性图的形式中，以揭示数据中的值级、特征级、异构和对象级关系。为了获得更简洁的数据表

示形式以进行聚类，我们将强大的四元数表示学习和谱聚类目标结合起来，执行无监督表示学习。结果表明，在信息丰富的异构特征编码的基础上，可以充分学习到聚类友好的表示。由于 QGRL 旨在表示异构特征，它肯定适用于任何特征类型的数据，包括数值数据、分类数据和异构数据。广泛的比较和消融研究表明 QGRL 在聚类方面的优越性。

本章节所提出的 QGRL 方法虽然在处理异质特征聚类问题上展现了显著优势，但也存在一些值得关注的局限性：其一，该方法依赖于预设的聚类数目（k），无法自动确定最优簇数，这限制了其在真实场景中聚类结构未知时的应用灵活性；其二，其构建层次耦合编码（HCE）和进行四元数图表示学习（QGRL）的过程涉及复杂的统计量计算（如多层级条件概率、基于图的统一不相似度计算、多视图投影）以及四元数哈密顿积等运算，导致模型训练和推理的计算开销相对较高，导致其应用在实时疾控预测等需要快速响应的场景会有所限制；其三，当前模型主要针对静态数据集设计，缺乏对动态数据流或增量数据的有效处理机制，难以适应数据持续演化的场景。下一章节本文将非平衡异常分布簇检测新策略展开深入探讨，通过 MGCPL 算法从细粒度的聚类出发，利用竞争惩罚机制和特征权重计算挖掘分类数据的多粒度簇结构，结合所提出的 CAME 策略以新编码策略和迭代求解目标函数实现准确聚类，同时对混合数据采用基于邻域粗糙集的处理策略，通过异构属性距离度量、基于密度和密度间隙的微划分及分层合并机制实现合理聚类。

第四章　多源数据联邦聚类新策略

鉴于前文对 QGRL 方法在异质特征度量中局限性的分析，本章节聚焦多源数据联邦聚类新策略展开系统研究。章节依次通过问题引入与本文贡献（4.1）、相关工作综述（4.2）、方法原理剖析（4.3）、实验设计与验证分析等模块（4.4），本章总结（4.5）深入阐释基于多粒度竞争学习的分类数据聚类方法（MCDC）。通过多组对比实验验证了新策略在簇结构识别、异常鲁棒性及混合数据适应性上的优势，同时客观分析了当前研究的局限，并对动态数据流处理、跨模态聚类扩展等未来方向进行展望。

4.1 引言

在数据挖掘和机器学习领域，聚类分析作为无监督学习的核心技术之一，致力于发现数据中潜在的结构与模式，将数据对象划分成具有相似特征的簇。在众多实际应用场景中，如医疗诊断、金融风险评估、市场细分等，精准的聚类分析能够为决策提供有力支持，帮助人们从海量数据中挖掘出有价值的信息。然而，现实世界中的数据往往呈现出复杂的分布特征，非平衡异常分布簇的存在给聚类分析带来了巨大挑战。

非平衡数据集中，不同簇的样本数量差异悬殊，这使得传统聚类算法容易偏向大样本簇，导致小样本簇的特征被忽视，难以准确识别。异常分布簇中的数据对象具有独特的分布模式，与正常簇差异显著，传统方法在区分异常簇和噪声数据时常常遭遇困境，严重影响聚类结果的准确性和可靠性。

在处理非平衡数据聚类方面，数据采样技术通过调整样本比例来改善数据平衡性，但可能会丢失关键信息。在异常分布簇检测方面，基于密度的聚类算法试图通过分析数据点的密度分布来识别异常簇，但在复杂数据分布下，易受噪声和数据稀疏性的干扰，检测效果不尽人意。

我们提出的基于多粒度竞争学习的分类数据聚类方法（MCDC），专注于解决分类数据聚类难题。分类数据在实际中广泛存在，如医疗数据中的症状分类、市场调研中的消费者属性分类等。MCDC 方法通过设计多粒度竞争惩罚学习（MGCPL）算法，自动学习不同粒度的对象划分，有效揭示了分类数据中复杂的嵌套多粒度簇效应。同时，基于 MGCPL 编码的聚类聚合（CAME）策略融合多粒度结果，实现了更精准的聚类。实验结果表明，该方法在多个真实分类数据集上表现卓越，具有较高的聚类准确性和鲁棒性。

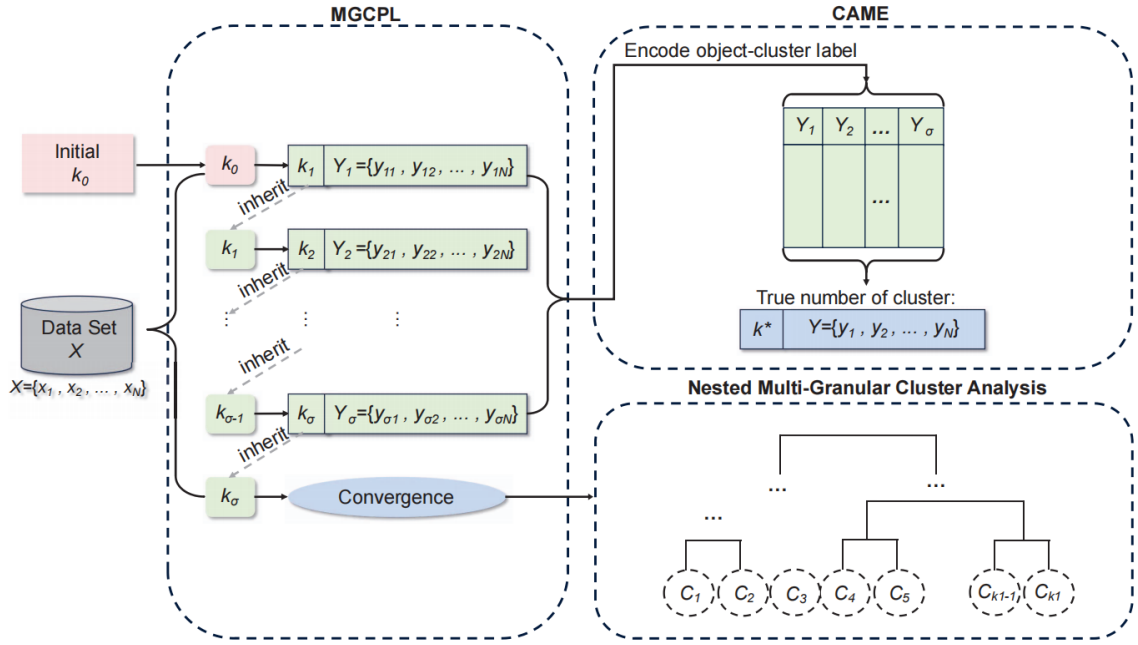


图 4-1 MCDC 运作流程图

同时我们提出的基于邻域粗糙集理论的自适应微划分和分层融合的混合数据精准聚类方法（AMPHM），则针对混合数据（包含数值和分类属性）的聚类问题。混合数据集在现实中也极为常见，如医学数据集中既有数值属性（如年龄、血压），又有分类属性（如症状、疾病类型）。传统聚类方法在处理混合数据时，难以准确度量数据对象间的相似性，导致聚类效果不佳。AMPHM 方法通过提出统一的距离度量，将样本划分为紧凑细粒度的簇，再经分层合并机制形成合理的簇，有效解决了混合数据聚类对先验知识和假设的依赖问题，在多个真实混合数据集上验证了其优越性。

本章贡献概括如下：

- 据我们所知，这是首次尝试揭示类别数据中复杂但普遍存在的嵌套多粒度簇效应，这有望启发后续的类别数据分析工作。
- 提出了一种新的聚类分析机制 MGCPL，用于探索类别数据的嵌套多粒度簇。MGCPL 效率高，可以为下游任务提供丰富的数据表示信息。
- 设计了一种称为 CAME 的聚合策略来融合 MGCPL 的多粒度结果。CAME 实现了更准确的聚类，其表示形式还可以增强现有的类别数据聚类方法。

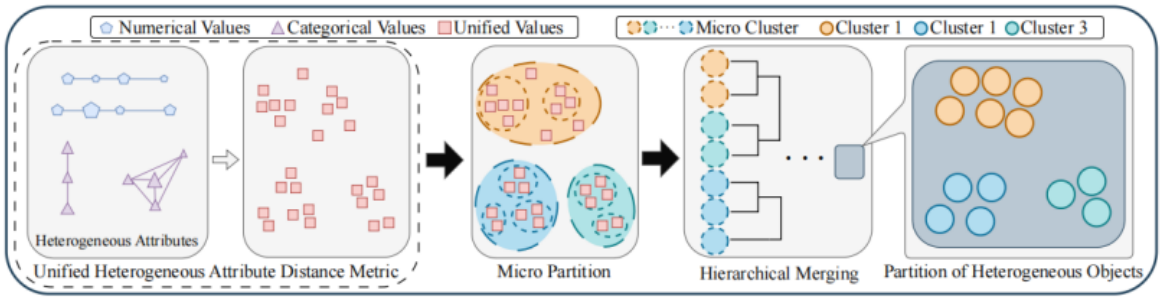


图 4-2 AMPHM 运作流程图

4.2 相关工作

4.2.1 分类数据聚类方法

在分类数据聚类领域，研究人员提出了多种方法来解决聚类过程中遇到的问题。早期，k-prototypes 方法及其变体[51, 52] 通过 one-hot 编码[53] 将分类属性值表示为二进制向量，将其视为数值数据进行处理，数值属性使用欧几里得距离测量，分类属性使用汉明距离。然而，这种方法忽略了分类属性不包含可良好定义空间的事实，在区分不同值对之间的差异方面存在局限性。

为了改进分类数据聚类效果，许多基于相似度的方法应运而生。这些方法利用相关属性的统计信息来反映目标分类属性的相似度，如基于关联的度量[54]、Ahmad 的度量[55] 以及基于上下文的度量[25, 56] 等。它们通过计算由相关属性派生的两个条件概率分布之间的距离来表明不相似性，但这些方法没有解决各种分类属性的异质性问题。

Lin 的距离度量[22] 通过计算属性值的熵值来表达属性值的相似性，基于熵的距离度量[10, 24], 则统一了分类属性的相似性并提供了属性加权方案。此外，基于远程学习的方法[28] 创新性地学习和调整了序数属性的有序结构并进行聚类，统一的距离度量也被引入来统一标称属性和序数属性之间的距离度量。然而，这些方法在处理复杂的分类数据时，仍面临着一些挑战，如对不太相似的样本分入同一组或划分相似的数据对象等问题。

近年来，一些基于深度学习模型的聚类方法，如 Mix2Vec[50]，将异构属性值转换为矢量，同时保留其结构分布信息。但它仍然区别对待数值属性和分类属性，没有充分考虑到分类属性的异质性。我们提出的 MCDC 方法有效解决了分类数据聚类中的诸多问题，通过 MGCPL 算法自动学习不同粒度的对象划分，以及 CAME 策略融合多粒度结果，在多个真实分类数据集上取得了良好的聚类效果。

4.2.2 混合数据聚类方法

混合数据聚类由于同时包含数值属性和分类属性，其聚类难度更大。传统的聚类方法在处理混合数据时，通常将数值属性和分类属性分别进行处理，然后再进行融合。例如，欧几里得距离和汉明距离常被分别用于数值属性和分类属性的度量，但这种简单的组合方式无法充分考虑属性之间的内在联系，导致聚类性能不理想。

为了解决混合数据聚类问题，研究人员提出了许多改进方法。一些方法尝试设计统一的距离度量来处理异构属性，如相似性度量在统一的概率框架内量化数据对象的相似性，但忽略了不同可能属性值之间存在的分歧概念。Lin 的距离度量和基于熵的相似性度量考虑了分类属性中值顺序，从信息论的角度衡量潜在值之间的不相似性，但在聚类时仍可能对样本划分不准确。

还有一些方法通过开发统一的距离度量来探索分类属性之间的相互依赖关系，如基于上下文的距离度量量化了分类属性之间的相互依赖关系，选择相关属性，并使用条件概率分布（CPDs）来指示潜在属性值之间的相似性。然而，这些方法都依赖于先验知识或假设，在面对复杂的真实数据集时，应用受到限制。

AMPHM 方法，针对混合数据聚类问题，利用邻域粗糙集理论将样本划分为紧凑细粒度的簇，再通过分层合并机制形成合理的簇。该方法突破了预先设定的簇数 k 和簇分布偏差带来的聚类性能瓶颈，能够对包含各种数值属性和分类属性组合的数据集进行聚类，在多个真实混合数据集上验证了其优越性。

4.2.3 联邦聚类方法

联邦聚类作为一种新兴的聚类技术，在分布式数据环境下具有重要的应用价值。它旨在通过在多个客户端之间协同进行聚类分析，同时保护数据隐私。早期的联邦聚类方法，如 k-FED，通过对客户端学习的非独立同分布（non-IID）进行一次性聚合，探索具有更高安全性级别的全局聚类分布。F-FCM 和 FFCM 则采用 fuzzy-c-means 作为本地聚类算法，通过仅传输对象 - 聚类隶属关系来增强隐私保护。

然而，这些方法大多假设客户端和服务器事先知道真实的聚类数量，这在实际应用中往往难以满足。为了解决这一问题，一些方法尝试在联邦聚类中自动确定最佳聚类数 k。例如，基于密度的聚类方法通过考虑对象的分布连接来进行聚类探索，并可以根据聚类划分的质量选择 k 。但这些方法的学习过程依赖于详细且充足的数据统计，

表 4-1 符号总结

符号	解释
X	数据集
F	特征
C	簇
n	数据对象的数量
d	特征的数量
Q	数据对象的划分矩阵
C_v	获胜簇
C_h	竞争的最近获胜者
η	学习率
u	竞争学习期间的簇权重
σ	MGCPL 学习到的粒度级别数量
κ	MGCPL 学习到的一系列 k 值
Γ	由 MGCPL 引导的数据表示
Θ	表示 Γ 的特征权重
k^*	簇的真实数量
Z	簇的模式

在联邦聚类中应用时受到限制。

最近，一些更先进的联邦聚类方法被提出，如联邦谱聚类方法通过聚合中间变量来构建全局相似度矩阵，应对异步场景，对通信频率偏差具有较强的鲁棒性。但它仍然依赖于真实聚类数 k 已知的假设，限制了其应用范围。在处理非平衡异常分布簇检测问题时，联邦聚类方法面临着更大的挑战，需要进一步探索有效的解决方案。

4.3 非平衡分布簇检测新技术

在解决非平衡异常分布簇检测问题时，我们运用 MCDC 方法和 AMPHM 方法的思路，针对不同类型的数据特点设计相应的策略。

4.3.1 基于多粒度竞争学习的分类数据处理策略

对于分类数据，我们参考 MCDC 方法，采用基于多粒度竞争学习的策略。该策略核心在于多粒度竞争惩罚学习（MGCPL）算法和基于 MGCPL 编码的聚类聚合（CAME）策略。常用符号总结如表4-1所示。

MGCPL 算法旨在自动学习不同粒度的对象划分，以适应分类数据中复杂的簇结

构。在实际数据聚类分析中，分类数据的簇往往存在多粒度效应，即数据对象在不同层次上形成有意义的簇。MGCPL 算法从一个相对较大的初始聚类数 k_0 开始竞争学习，通过竞争机制淘汰不重要的簇，逐步得到更合理的聚类数 k_1 、k_2 、$\cdots$ 、k_σ 。在这个过程中，引入竞争惩罚机制，对于每个输入数据对象 x_i ，确定获胜簇 C_v 和竞争簇 C_h ，获胜簇向 x_i 更新，竞争簇则受到惩罚远离获胜簇，这样能让竞争簇有更多机会探索距离空间中的簇分布，避免获胜簇过度吸收周围种子点，从而有效挖掘多粒度簇结构。

同时，考虑到不同分类特征对聚类的贡献不同，MGCPL 算法通过计算特征的簇间差异 α_{rl} 和簇内相似度 β_{rl} 来衡量特征 F_r 对簇 C_l 的贡献 H_{rl}，进而得到特征权重 ω_{rl} 。这一特征权重机制在计算对象与簇的相似度时，能够更合理地考虑不同特征的重要性，提高聚类的准确性。当竞争惩罚学习在某一聚类数下收敛后，学习机制继承当前结果并重新启动学习，探索更粗粒度的簇，通过不断递归，最终得到一系列聚类数和相应的聚类结果。

基于 MGCPL 算法得到的多粒度簇分布信息，CAME 策略通过新的编码策略将对象 - 簇隶属关系作为数据表示。由于不同粒度的特征对最终聚类的贡献因所求聚类数 k 而异，CAME 策略将聚类聚合表示为特征重要性学习的形式，通过最小化目标函数 $P(Q,\Theta)$ 来实现更准确的聚类。在目标函数中，Q 是对象划分矩阵，Θ 是一组特征权重，Z_{lr} 表示第 l 个簇在表示 Γ 中的模式，$d(x_{ir},Z_{lr})$ 是对象 x_{ir} 与簇 Z_{lr} 的特征值之间的汉明距离。通过迭代求解固定对象划分 Q 更新特征权重 Θ 和固定特征权重 Θ 计算对象划分 Q 这两个最小化问题，使得学习过程收敛到一个最小解，从而得到最终的聚类结果 Q 。

4.3.2 基于邻域粗糙集的混合数据处理策略

对于混合数据，我们参考 AMPHM 方法，采用基于邻域粗糙集的处理策略。该策略主要由异构属性距离度量、基于邻域粗糙集的微划分和分层合并机制组成。

首先，异构属性距离度量采用转换成本作为度量方式，统一关于不同异构属性 A_r 的距离 $D_r(x_{ir},x_{jr})$ 。通过将一种条件概率分布（CPD）转换为另一种所需的工作量来量化距离，具体使用移土距离（EMD）来测量两个分类取值的 CPD 之间的差异。同时，考虑属性间的依赖关系计算权重，使得距离度量能更好地反映数据对象之间的真实差异。在处理数值属性时，当假设每个数值属性的距离空间是独立的一维连续欧氏距离

空间且其域为 [0, 1] 时，该距离度量能统一处理分类属性和数值属性，证明了其适用性和有效性。

基于邻域粗糙集的微划分方法旨在创建不重叠的邻域集，更合理地划分对象。现有的传统邻域粗糙集创建的邻域集可能存在重叠，导致计算成本高且划分不准确。该微划分方法通过计算样本的密度 ρ_i 和密度间隙 ξ_i 来选择代表样本。密度的计算可以直观地选择邻域集，确保超过重要类边界的对象不被纳入 x_i 邻域集；而密度间隙则用于衡量样本周围的密度情况，具有更高密度间隙的对象更适合作为代表样本。根据密度差距按降序对样本进行排序，然后创建微划分，直到所有样本被邻域集合并。

分层合并机制则是在微划分的基础上，对邻域集进行合并。在每一层迭代更新邻域集和数据集，分为两个步骤：保持数据集不变，计算邻域集和代表样本集；保持代表样本集不变，通过代表样本集更新数据集。重复这两个步骤，直到邻域代表样本的数量等于簇的数量。通过这种分层合并机制，将微簇逐渐合并为更大的簇，实现混合数据的合理聚类。

4.4 实验设置与分析

为了全面评估所提出的非平衡异常分布簇检测新策略的性能，我们进行了一系列实验，实验内容包括实验设置、聚类性能评估、消融实验等，旨在验证新策略的有效性和优越性。

4.4.1 实验设置

我们选取了多个具有代表性的数据集，涵盖分类数据和混合数据。分类数据集包括来自 UCI 机器学习库的 Car Evaluation、Congressional 等，这些数据集在分类数据聚类研究中被广泛使用，具有不同的特征数量和类别分布。混合数据集则选取了 Dermatology、Autism - Adolescent 等，这些数据集包含数值和分类属性，能有效测试方法在处理混合数据时的性能。同时，我们还生成了一些合成数据集，以模拟特定的数据分布情况，进一步评估方法在不同场景下的表现。

在对比方法的选择上，对于分类数据聚类，我们选取了传统的 k-modes[57]、ROCK[58]，以及近期提出的先进方法 WOCIL[59]、GUDMM[60]、FKMAWCW[61]、ADC[62] 等。这些方法在分类数据聚类领域具有一定的代表性，能够为评估新策略提供全面的参考。对于混合数据聚类，我们选择了 Jia 提出的距离度量（JDM）[63]、耦合度

量相似度（CMS）[64]、基于熵的距离度量（EDM）[10]、统一距离度量（UDM）[65] 结合 *k*-modes 和 *k*-prototype 等方法。这些对比方法涵盖了处理混合数据聚类的不同思路和技术。

为了准确评估聚类性能，我们采用了多个有效性指标，包括聚类准确率（ACC）、调整兰德指数（ARI）、调整互信息（AMI）、轮廓系数指数（SC）、Calinski - Harabasz 指数（CH）等。ACC 反映了聚类结果中正确分类的样本比例，直观地展示了聚类的准确性；ARI 和 AMI 通过比较聚类结果与真实标签的一致性，从不同角度量化聚类质量；SC 综合考虑了聚类的内部紧密度和聚类间的分散度，能够更全面地评估聚类效果；CH 计算聚类之间的距离与聚类内部距离的比率，值越大表示聚类性能越好。

实验在统一的硬件和软件环境下进行，以确保实验结果的准确性和可重复性。对于每个数据集，我们对所有参与比较的方法进行多次实验，记录每次实验的性能数据，并计算平均性能和标准偏差，以提高实验结果的可靠性。

4.4.2 聚类性能评估

在分类数据聚类实验中，按照设定的实验方案，将基于多粒度竞争学习的分类数据处理策略与对比方法在多个分类数据集上进行聚类性能对比。结果如表4-2所示，MCDC + G. 和 MCDC + F. 分别是 MCDC 的变体，它们采用了 GUDMM 和 FKMAWCW，每个数据集上的最优结果和次优结果分别用 **粗体**和 下划线 突出显示在 Car Evaluation 数据集上，新策略的 ACC 达到 0.373，在所有对比方法中表现突出，表明其能有效识别数据对象的真实类别，聚类准确性高。在 ARI 指标上，新策略在 Congressional 数据集上取得了 0.557 的高分，远超部分对比方法，体现了其聚类结果与真实标签的高度一致性。在 AMI 指标方面，新策略在多个数据集上也展现出良好的性能，如在 Vote 数据集上，AMI 值达到 0.566，说明该方法能从信息论角度有效度量聚类结果与真实标签的匹配程度。

在混合数据聚类实验中，将基于邻域粗糙集的混合数据处理策略与对比方法在多个混合数据集上进行实验，实验结果记录于表4-3和表4-4。以 Dermatology 数据集为例，新策略的 CA 值高达 0.7677，ARI 值为 0.6776，显著优于其他对比方法，表明其在处理混合数据时，能精准划分数据对象，有效挖掘数据中的潜在结构。在 Breast Cancer 数据集上，新策略同样表现出色，CA 值达到 0.7657，ARI 值为 0.1092，进一步验证了其

表 4-2 类别数据集上各指标聚类性能

指标	数据	K-MODES	ROCK	WOCIL	FKMAWCW	GUDMM	ADC	MCDC	MCDC+G.	MCDC+F.
ACC	Car.	0.372±0.00	0.326±0.00	0.270±0.00	0.371±0.00	0.372±0.00	0.361±0.00	0.373±0.00	0.270±0.00	**0.414±0.00**
	Con.	0.866±0.00	0.506±0.00	**0.874±0.00**	0.796±0.01	0.818±0.00	**0.874±0.00**	**0.874±0.00**	**0.874±0.00**	**0.874±0.00**
	Che.	0.551±0.00	0.505±0.00	0.531±0.00	0.561±0.00	0.554±0.00	0.548±0.00	0.578±0.00	0.547±0.00	**0.585±0.00**
	Mus.	0.740±0.02	0.509±0.00	0.678±0.00	0.000±0.00	0.501±0.00	0.752±0.02	0.710±0.00	0.613±0.00	**0.784±0.00**
	Tic.	0.557±0.00	**0.674±0.00**	0.526±0.00	0.538±0.00	0.507±0.00	0.535±0.00	0.602±0.00	0.642±0.00	0.646±0.00
	Vot.	0.869±0.00	0.500±0.00	0.888±0.00	0.778±0.01	0.828±0.00	0.888±0.00	**0.905±0.00**	**0.905±0.00**	**0.905±0.00**
	Bal.	0.448±0.00	0.496±0.00	0.419±0.00	0.463±0.00	0.000±0.00	0.442±0.00	0.464±0.00	0.453±0.00	**0.506±0.00**
	Nur.	0.332±0.00	0.000±0.00	0.239±0.00	0.315±0.00	0.000±0.00	0.337±0.00	0.340±0.00	0.305±0.00	**0.432±0.00**
ARI	Car.	0.027±0.00	0.023±0.00	0.001±0.00	-0.002±0.00	**0.054±0.00**	0.017±0.00	0.051±0.00	0.001±0.00	0.027±0.00
	Con.	0.536±0.00	-0.004±0.00	**0.557±0.00**	0.385±0.05	0.394±0.00	**0.557±0.00**	**0.557±0.00**	**0.557±0.00**	**0.557±0.00**
	Che.	0.014±0.00	-0.001±0.00	0.003±0.00	0.020±0.00	0.012±0.00	0.015±0.00	0.024±0.00	0.008±0.00	**0.028±0.00**
	Mus.	0.303±0.07	-0.001±0.00	0.125±0.00	0.000±0.00	-0.003±0.00	0.321±0.06	0.186±0.01	0.051±0.00	**0.323±0.00**
	Tic.	0.017±0.00	**0.120±0.00**	0.000±0.00	-0.002±0.00	-0.001±0.00	0.007±0.00	0.038±0.00	0.079±0.00	0.062±0.00
	Vot.	0.543±0.00	-0.004±0.00	0.600±0.00	0.349±0.05	0.427±0.00	0.600±0.00	**0.655±0.00**	**0.655±0.00**	**0.655±0.00**
	Bal.	0.027±0.00	**0.080±0.00**	0.005±0.00	0.055±0.00	0.000±0.00	0.025±0.00	0.052±0.00	0.016±0.00	0.079±0.00
	Nur.	0.049±0.00	0.000±0.00	0.002±0.00	0.028±0.00	0.000±0.00	0.052±0.00	0.051±0.00	0.004±0.00	**0.166±0.00**
AMI	Car.	0.049±0.00	0.050±0.00	0.003±0.00	0.082±0.00	0.117±0.00	0.047±0.00	**0.123±0.00**	0.003±0.00	0.015±0.00
	Con.	0.473±0.00	0.001±0.00	**0.484±0.00**	0.337±0.03	0.380±0.00	**0.484±0.00**	**0.484±0.00**	**0.484±0.00**	**0.484±0.00**
	Che.	0.012±0.00	0.000±0.00	0.003±0.00	**0.021±0.00**	0.011±0.00	0.015±0.00	0.020±0.00	0.005±0.00	0.020±0.00
	Mus.	0.280±0.05	0.000±0.00	0.235±0.00	0.000±0.00	0.044±0.00	**0.347±0.04**	0.209±0.01	0.036±0.00	0.248±0.00
	Tic.	0.012±0.00	**0.120±0.00**	0.007±0.00	0.005±0.00	0.000±0.00	0.006±0.00	0.020±0.00	0.058±0.00	0.023±0.00
	Vot.	0.457±0.00	0.000±0.00	0.522±0.00	0.301±0.03	0.417±0.00	0.522±0.00	**0.566±0.00**	**0.566±0.00**	**0.566±0.00**
	Bal.	0.026±0.00	0.071±0.00	0.008±0.00	0.048±0.00	0.000±0.00	0.026±0.00	0.083±0.00	0.017±0.00	**0.089±0.00**
	Nur.	0.060±0.00	0.000±0.00	0.004±0.00	0.043±0.00	0.000±0.00	0.061±0.00	0.077±0.00	0.022±0.00	**0.208±0.00**
FM	Car.	0.409±0.00	0.394±0.00	0.369±0.00	0.406±0.00	0.413±0.00	0.401±0.00	0.407±0.00	0.369±0.00	**0.434±0.00**
	Con.	0.774±0.00	0.518±0.00	**0.784±0.00**	0.711±0.01	0.754±0.00	**0.784±0.00**	**0.784±0.00**	**0.784±0.00**	**0.784±0.00**
	Che.	0.544±0.00	0.525±0.00	0.507±0.00	**0.578±0.00**	0.554±0.00	0.555±0.00	0.573±0.00	0.519±0.00	0.532±0.00
	Mus.	0.667±0.02	0.525±0.00	0.657±0.00	0.000±0.00	0.687±0.00	**0.721±0.01**	0.640±0.00	0.544±0.00	0.662±0.00
	Tic.	0.538±0.00	0.581±0.00	0.527±0.00	0.547±0.00	0.524±0.00	0.526±0.00	0.548±0.00	0.562±0.00	**0.612±0.00**
	Vot.	0.772±0.00	0.500±0.00	0.800±0.00	0.696±0.01	0.734±0.00	0.800±0.00	**0.827±0.00**	**0.827±0.00**	**0.827±0.00**
	Bal.	0.426±0.00	0.441±0.00	0.437±0.00	0.424±0.00	0.000±0.00	0.425±0.00	**0.464±0.00**	0.460±0.00	0.452±0.00
	Nur.	0.303±0.00	0.000±0.00	0.260±0.00	0.306±0.00	0.000±0.00	0.305±0.00	0.309±0.00	0.321±0.00	**0.396±0.00**

表 4-3 所提出算法与其他方法基于 CA 指标的聚类性能比较结果

数据集	KMD/KPT	WKM	OCIL	JDM	CMS	EDM	UDM	AMPHM
DT	0.5538±0.10	0.6234±0.09	0.6750±0.10	0.6654±0.10	0.6017±0.14	0.5872±0.10	0.6845±0.11	**0.7677±0.00**
AA	0.5301±0.03	0.5254±0.02	0.5186±0.03	0.5788±0.05	0.5244±0.03	0.5577±0.03	0.5579±0.02	**0.5962±0.00**
CT	0.5298±0.02	0.5234±0.03	0.5056±0.00	0.5224±0.02	0.5259±0.02	0.5365±0.03	**0.5781±0.02**	0.5450±0.00
HR	0.3638±0.01	0.4078±0.05	0.3761±0.04	0.3753±0.02	0.4048±0.04	0.4070±0.03	0.4040±0.03	**0.4167±0.00**
BC	0.5187±0.02	0.5835±0.09	0.5414±0.06	0.5823±0.10	0.5281±0.04	0.5304±0.02	0.5685±0.19	**0.7657±0.00**
LG	0.4531±0.04	0.4385±0.05	0.5003±0.04	0.4730±0.04	0.4189±0.05	0.4523±0.04	0.4704±0.04	**0.5608±0.00**
VT	0.8643±0.00	0.8570±0.07	**0.8810±0.00**	0.8681±0.00	0.8649±0.01	0.8317±0.10	0.8721±0.00	0.8736±0.00
ES	0.3674±0.03	0.3750±0.04	0.3839±0.04	0.3505±0.03	**0.4016±0.04**	0.3664±0.02	0.3681±0.03	0.3934±0.00
SW	0.3924±0.03	0.3750±0.03	0.3732±0.03	0.3337±0.03	0.3310±0.03	0.3321±0.01	0.3744±0.03	**0.4350±0.00**
Ave. Rank	5.7778	4.6667	4.5556	4.8889	5.6667	5.6667	3.4444	1.3333

在混合数据聚类任务中的优越性。

表 4-4 各类对比方法与所提出算法在 ARI 指标上的聚类性能比较

数据集	KMD/KPT	WKM	OCIL	JDM	CMS	EDM	UDM	AMPHM
DT	0.4222±0.12	0.5091±0.09	0.6063±0.10	0.6140±0.13	0.5176±0.17	0.4393±0.12	0.6266±0.15	**0.6776±0.00**
AA	-0.0031±0.01	-0.0055±0.01	-0.0073±0.01	0.0182±0.03	-0.0086±0.01	0.0063±0.01	-0.0153±0.01	**0.0185±0.00**
CT	-0.0081±0.01	-0.0079±0.01	-0.0208±0.00	-0.0142±0.01	-0.0075±0.01	0.0017±0.01	**0.0133±0.02**	-0.0010±0.00
HR	-0.0121±0.00	0.0069±0.02	-0.0035±0.02	-0.0063±0.01	0.0075±0.02	0.0080±0.02	0.0070±0.02	**0.0094±0.00**
BC	-0.0041±0.00	0.0396±0.07	0.0112±0.04	0.0405±0.07	0.0029±0.02	0.0066±0.01	0.0620±0.02	**0.1092±0.00**
LG	0.1128±0.04	0.0845±0.04	**0.1817±0.05**	0.1231±0.04	0.0884±0.04	0.0893±0.03	0.1317±0.05	0.1293±0.00
VT	0.5297±0.00	0.5273±0.11	**0.5795±0.00**	0.5411±0.01	0.5319±0.03	0.4776±0.17	0.5527±0.01	0.5568±0.00
ES	0.1621±0.02	0.1717±0.03	0.1929±0.02	0.1666±0.02	**0.2115±0.03**	0.1627±0.04	0.2106±0.02	0.1725±0.00
SW	0.0571±0.02	0.0456±0.02	0.0516±0.02	0.0519±0.01	0.0505±0.02	0.0591±0.01	0.0760±0.01	**0.0847±0.00**
Ave. Rank	6.3333	5.8889	4.4444	4.5556	5.1111	4.8889	2.8889	1.8889

4.4.3 消融实验

为了深入探究新策略中各组件的有效性，我们分别对基于多粒度竞争学习的分类数据处理策略和基于邻域粗糙集的混合数据处理策略进行消融实验。

对于基于多粒度竞争学习的分类数据处理策略，去除其主要技术组件构建不同版本。去除 CAME 策略中的特征加权机制后得到的版本，在多个数据集上的 ARI 性能下降明显。如在 Mushroom 数据集上，原策略的 ARI 值为 0.323，而去除特征加权机制后的版本 ARI 值仅为 0.051，这充分表明 CAME 策略中的特征加权机制在学习嵌入特征重要性以及融合 MGCPL 提供的多粒度信息方面发挥着关键作用。

去除整个 CAME 模块，使用 MGCPL 学习到的 kσ 进行聚类得到的版本，在几乎所有数据集上的性能优于采用传统竞争学习初始化的版本。这表明 MGCPL 从不同 k 值学习簇分布的机制是有效的，其前期学习的结果能为最后一轮学习提供更合理的初始化，进而提升聚类性能。

为了进一步验证所提出的 MCDC 方法各个主要组成部分的有效性，我们将其简化为以下四个版本:1)MCDC4 是用固定的相同权重替换 CAME 的特征加权机制；2)MCDC3 是通过从 MCDC 中删除整个 CAME 模块并使用 MGCPL 学习到的 kσ 进行聚类而获得的；3)MCDC2 是用常规竞争学习以 k*+2 作为初始化替换 MCDC3 的 MGCPL 而获得的版本；4)MCDC1 是通过从 MCDC2 中进一步删除竞争学习机制而形成的版本。实验结果如图4-3所示。

4.4.4 显著性检验

为了从统计层面验证新策略的优越性，我们进行显著性检验。在分类数据聚类实验中，采用 Wilcoxon signed - rank test 对基于多粒度竞争学习的分类数据处理策略与其

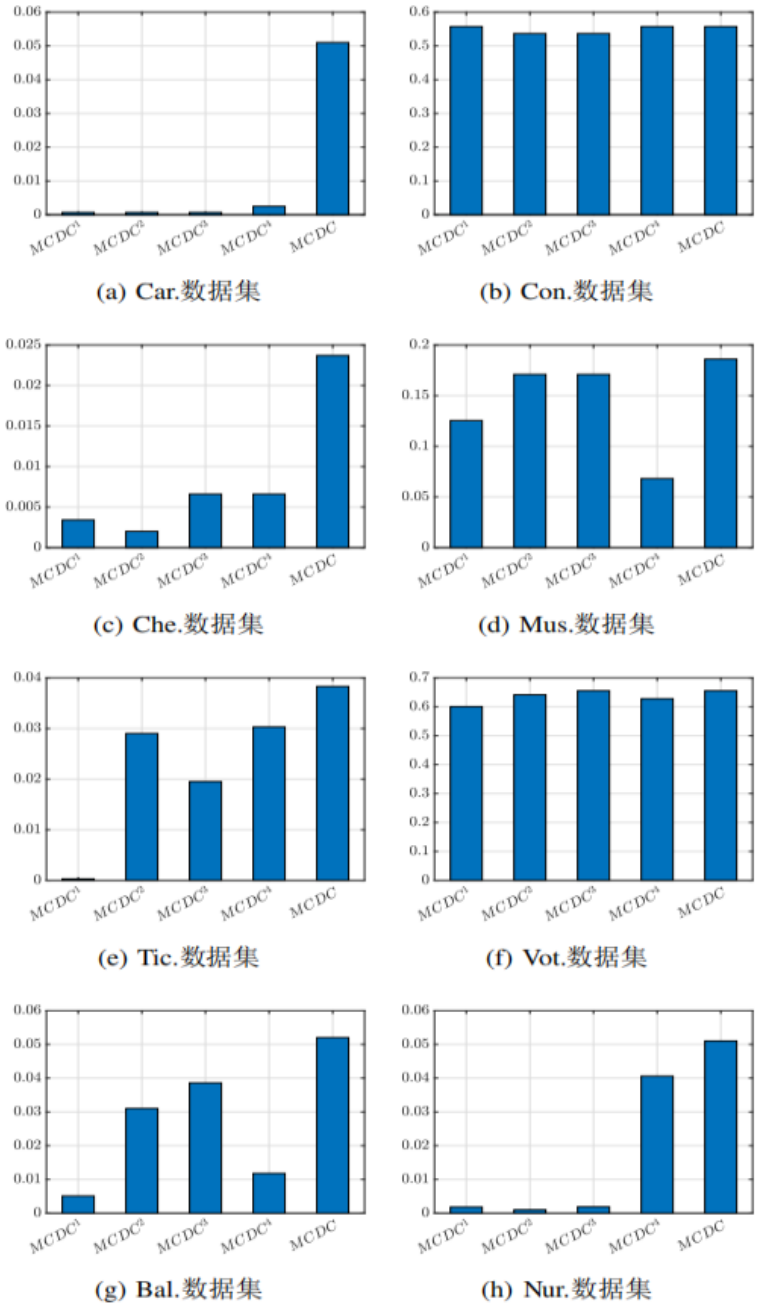


图 4-3 MCDC 与其四个简化版本比较结果

表 4-5 显著性检验结果

算法	ACC	ARI	AMI	FM
K-MODES	+	+	-	+
ROCK	+	+	-	+
WOCIL	+	+	+	+
FKMAWCW	+	+	+	+
GUDMM	+	+	+	+
ADC	+	+	-	-

他对比方法的聚类性能进行检验。以该策略表现最佳的变体与其他对比方法在 90% 置信区间下进行比较，使用“+”表示该变体显著优于对应对比方法。从表4-5可以看出，在几乎所有指标下，该变体都显著优于大部分对比方法，有力证明了该策略的优越性。

在混合数据聚类实验中，采用类似的显著性检验方法，对基于邻域粗糙集的混合数据处理策略与其他对比方法的聚类性能进行统计分析，进一步验证该策略在处理混合数据聚类问题上的优势。

4.5 本章小结

本章围绕多源数据联邦聚类新策略展开深入研究，结合 MCDC 方法和 AMPHM 方法，针对分类数据和混合数据分别提出了有效的处理策略。

通过 MGCPL 算法自动学习不同粒度的对象划分，有效揭示了分类数据中复杂的嵌套多粒度簇效应；CAME 策略融合多粒度结果，显著提升了聚类准确性。实验结果表明，该策略在多个真实分类数据集上性能卓越，在聚类准确率、调整兰德指数等指标上优于多种对比方法，且具有良好的可扩展性和稳定性。

基于邻域粗糙集的混合数据处理策略，利用邻域粗糙集理论将样本划分为紧凑细粒度的簇，再通过分层合并机制形成合理的簇。该策略突破了传统聚类方法的限制，在多个包含数值和分类属性的数据集上表现出色，有效提高了混合数据聚类的精度和可靠性。

通过对这两种策略的研究和实验验证，我们为非平衡异常分布簇检测提供了更有效的解决方案。然而，目前的研究仍存在一定局限性。基于多粒度竞争学习的分类数据处理策略尚未拓展到更复杂的异构特征数据和多模态数据，且未考虑动态数据聚类；基于邻域粗糙集的混合数据处理策略在处理高维数据和大规模数据时可能面临挑战。

在下一章，本文将针对现实中知识更新时效差的问题，提出动态环境持续学习新模型。

第五章　动态环境持续学习新模型

在多源异质大数据联合分析中，由于面向的本地节点大多为隐私敏感的重要数据，如医院中病人的个人信息，战场中战略资源的位置信息等数据。这种全局数据分布的碎片化与隐私保护需求的相互约束构成了本项目大数据联合分析的其中一个核心矛盾。针对此问题，一个有效的数据分析框架为联邦学习框架，但现有联邦聚类方法都存在着假设所有本地节点同时通信且依赖预设簇数进行数据分析的理想场景，但真实场景中客户端通信异步性、数据非独立同分布（non-IID）以及簇数未知性，使得传统方案难以有效整合分散数据。本章节提出了一种新的联邦聚类方法，在无需依赖“真实”聚类数量的前提下，可以从异步通信的客户端中学习分布共识，本文的新型平衡机制也可以自适应地平衡不同客户端对服务器的贡献，所提出的方法（AFCL）能够在非独立同分布这种接近现实的复杂情况下表现良好。

5.1 引言

联邦学习（Federated Learning, FL）在实现分布式机器学习并保护隐私方面广泛应用[66–68]，在无监督的联邦学习任务中，联邦聚类（Federated Clustering, FC）通过将数据集划分为紧凑的对象聚类，在挖掘数据概念和知识方面展现了巨大的潜力[69–71]。然而，在没有标签指导的情况下，联邦聚类面临着由于隐私保护要求和异质性客户端的非独立同分布（non-IID）带来的重大挑战。大多数现有方法通过首先让客户端学习聚类分布，然后将保护隐私的聚类知识传递给服务器进行全局聚合来解决联邦聚类（FC）问题[72, 73]。例如，k-FED[1] 通过对客户端学习的非独立同分布（non-IID）进行一次性聚合，探索具有更高安全性级别的全局聚类分布。两个独立的工作，F-FCM[2] 和 FFCM[3]，名称和原理相似，采用 fuzzy-c-means 作为本地聚类算法。由于模糊对象，聚类隶属度能更精细地反映数据对象的划分信息，因此 FL 的隐私约束所导致的信息损失可以得到显著弥补。

值得注意的是，大多数现有的研究忽视了由于客户端通信能力差异导致的客户端异步问题。由于缺乏数据标签，这种问题可能会严重偏向那些更频繁上传其分布的客户端。为此，最近提出的联邦谱聚类方法[7] 通过聚合中间变量来构建全局相似度矩阵，从而应对异步场景，这种方法对通信频率偏差具有较强的鲁棒性。然而，它仍然依赖于一个简单的假设，即真实的聚类数 k 已知，这限制了联邦聚类在许多实际应用中面

对未知 k 时的应用范围。

自动确定最佳聚类数 k 一直是近年来聚类研究的一个热门话题。轮廓系数（Silhouette Coefficient）[12] 通过考虑聚类的内部紧密度和聚类间的分散度来确定 k。基于密度的聚类方法[74–76] 通过考虑对象的分布连接来进行聚类探索，并可以根据聚类划分的质量选择 k。更先进的基于学习的方法[18, 75–78] 提出了通过让种子点竞争或合作，消除冗余的低质量聚类来学习 k。最近，基于显著性的聚类方法[19, 79] 提出了使用间隙统计量（gap statistics）来估计 k。然而，它们的学习过程依赖于详细且充足的数据统计，这使得它们无法在联邦聚类中应用。因此，k 的缺失给联邦聚类带来了聚类指导的缺乏，同时也增加了额外的 k 学习目标，共同使得异步联邦聚类成为一个具有挑战性的任务。异步联邦聚类（FC）与典型联邦聚类的详细比较如图5-1所示。与典型联邦聚类中客户端同步通信并且已知聚类数 k 不同，我们关注的异步联邦聚类问题则是从由于客户端通信不均匀所导致的复杂上传分布中学习聚类及聚类数。更具体地说，异步联邦聚类的难点在于聚类数 k 学习与客户端异步上传的交叉耦合问题。即，不同客户端学习到的分布并不与统一的 k 相关联，而服务器在学习 k 时，由于异步上传的不可靠本地分布，面临着较大的挑战。因此，我们提出了异步联邦聚类学习（AFCL）方法，该方法能够在异步通信的客户端中自适应地学习最优的聚类数。它为客户端统一生成种子点，并通过种子与客户端内对象之间的差异来积累周围对象的分布信息，从而捕捉客户端自身的分布。然后，将这些累积的信息传递给服务器，以更新种子并进行客户端与种子分布信息的融合。为了在客户端之间达成共识，我们让邻近的种子共享它们的更新强度，从而实现种子与种子之间的信息补全，因为非独立同分布（non-IID）的客户端可能只提供部分全局分布。同时，我们还设计了一个平衡机制，用来评估和调整来自不同客户端的更新强度，以缓解它们异步参与所带来的潜在偏差。因此，AFCL 能够在具有挑战性的异步联邦场景下，自动收敛冗余的邻近种子，学习出适当数量的聚类。大量实验验证了 AFCL 的有效性。本工作的主要贡献总结为以下三点：

- 本文提出了一种新的联邦聚类（FC）方法，能够从异步通信的客户端中学习分布共识，而无需依赖“真实”聚类数。这有助于增强联邦学习（FL）的鲁棒性和普适性。

- 本文首次考虑并尝试解决联邦聚类中一个更加现实但具有挑战性的非独立同分

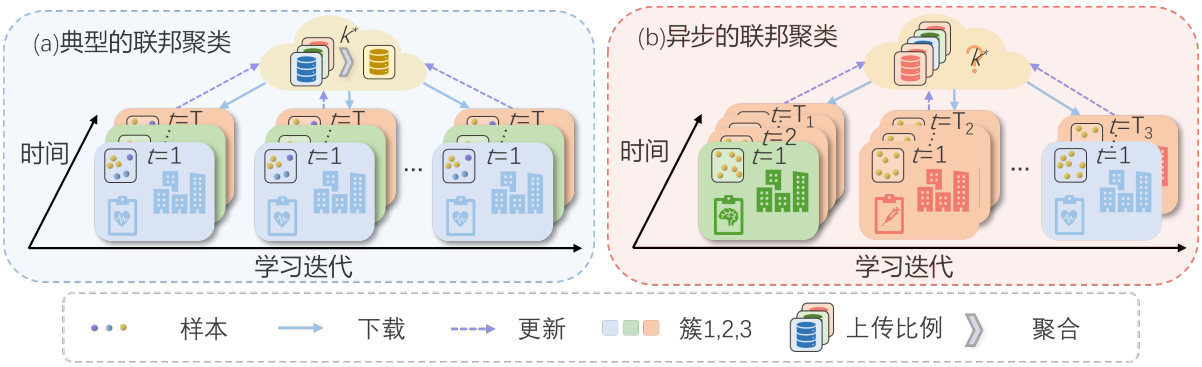


图 5-1 典型联邦聚类（FC）方法与本文提出的异步联邦簇学习（AFCL）方法对比

布（non-IID）情况，即一个全局聚类可能由属于不同客户端的完全不重叠的子聚类组成。

- 本文引入了一种平衡机制，用于分配具有异质通信方式的客户端在服务器上的交互学习中的贡献，即使它们仅通过一次参与学习。

5.2 相关工作

5.2.1 联邦聚类

一种名为 k-FED[1] 的单次通信联邦聚类（FC）方法已被提出，旨在缓解通信过程中信息泄漏的问题；而 FedKKM[80] 则提出了一种新型的 Lanczos 算法，通过分布式矩阵提升了通信效率。与此同时，F-FCM[2] 和 FFCM[3] 利用模糊聚类技术，通过仅传输对象-聚类隶属关系来增强隐私保护。此外，最近有一种多视角 FC 方法[4] 被提出，旨在通过设计共识原型学习策略，将多视角聚类扩展到联邦学习场景。然而，这些方法都假设客户端和服务器事先知道真实的聚类数量。最近，一种名为 VKMC[81] 的 FC 框架被提出，用于改进基于核心集的垂直联邦学习（FL）；而 HFDPC[5] 则提出了一种基于密度的 FC 方法，通过引入类似密度链的机制，提升数据划分的效果。最新的联邦子空间聚类（Fed-SC)[6] 和联邦谱聚类（FedSC)[7] 方法，分别解决了高维和噪声数据的 FC 问题。尽管上述一些 FC 方法考虑了客户端的非独立同分布（non-IID）或异步性问题，但它们的大多数解决方案仍然严重依赖于‘真实’聚类数量 k 的可用性，这在实际复杂场景中限制了它们的应用。

表 5-1 符号总结

符号	说明
$\mathbf{X}$	全局数据集
$\mathbf{X}^{\{g\}}$	第 g 个客户端的数据集
$\mathbf{M}^{\{g\}}$	第 g 个客户端的种子点
$\mathbf{m}_l$	第 l 个全局种子点
$\mathbf{Q}^{\{g\}}$	对应第 g 个客户端的对象-簇隶属矩阵
$\mathbf{B}^{\{g\}}$	第 g 个客户端的簇中心集合
$\mathbf{R}_l^{\{g\}}$	第 g 个客户端上第 l 个种子的更新强度
k^*	数据集 $\mathbf{X}$ 的“真实”全局簇数量

5.2.2 在未知簇数的情况下聚类

近年来，更现实的无监督或弱监督学习引起了广泛关注，尤其在一些重要的应用领域[8–11]。聚类是一个关键的无监督学习技术，其中传统的聚类方法需要手动确定最佳的聚类数 k[12, 13]。为了实现自动选择 k，基于密度的聚类[14–16] 已被提出，用于在聚类探索过程中通过“膝点”自动确定 k。最近，更先进的基于学习的方法[17, 18, 82] 引入了合作或竞争机制，以避免过多的聚类中心。它们同时确保对象分布的全面表示，并使冗余聚类中心可以被学习，从而实现令人满意的聚类性能。最近，基于显著性的方法[19, 79] 被提出，用来严格判断当前 k 下聚类分布的重要性。然而，所有上述解决方案需要详细的全数据集统计信息，这限制了它们在联邦聚类（FC）中的应用。

5.3 AFCL：异步联邦聚类的学习方法

在本节中，我们首先定义了异步 FC 问题，然后提出了由两个关键技术组件组成的 AFCL 算法：1）CSUA：客户端侧更新累积，以及 2）SSSI：服务器侧种子交互。常用符号总结在表5-1中，AFCL 的概述如图5-2所示。

5.3.1 问题定义

假设有一个由 p 个客户端组成的联邦网络，将整个数据集 $\mathbf{X}$ 分割为相应的子集 $\{\mathbf{X}^{\{1\}}, \mathbf{X}^{\{2\}}, ..., \mathbf{X}^{\{g\}}, ..., \mathbf{X}^{\{p\}}\}$，其中第 g 个客户端的子集 $\mathbf{X}^{\{g\}}$ 包含 $n^{\{g\}}$ 个对象 $\{\mathbf{x}_1^{\{g\}}, \mathbf{x}_2^{\{g\}}, ..., \mathbf{x}_{n^{\{g\}}}^{\{g\}}\}$，$\sum_{g=1}^{p} n^{\{g\}} = n$，且每个对象 $\mathbf{x}_i^{\{g\}} = [x_{i,1}^{\{g\}}, x_{i,2}^{\{g\}}, ..., x_{i,d}^{\{g\}}]^\top$ 是一个 d 维向量，我们使用 $\mathbf{Q} \in \mathbb{R}^{n \times k}$ 来表示对象与聚类的隶属关系，传统的聚类目标是最小化对象与聚类中心之间的总体聚类内不相似度（也称为种子点或种子交换问题），该目标可

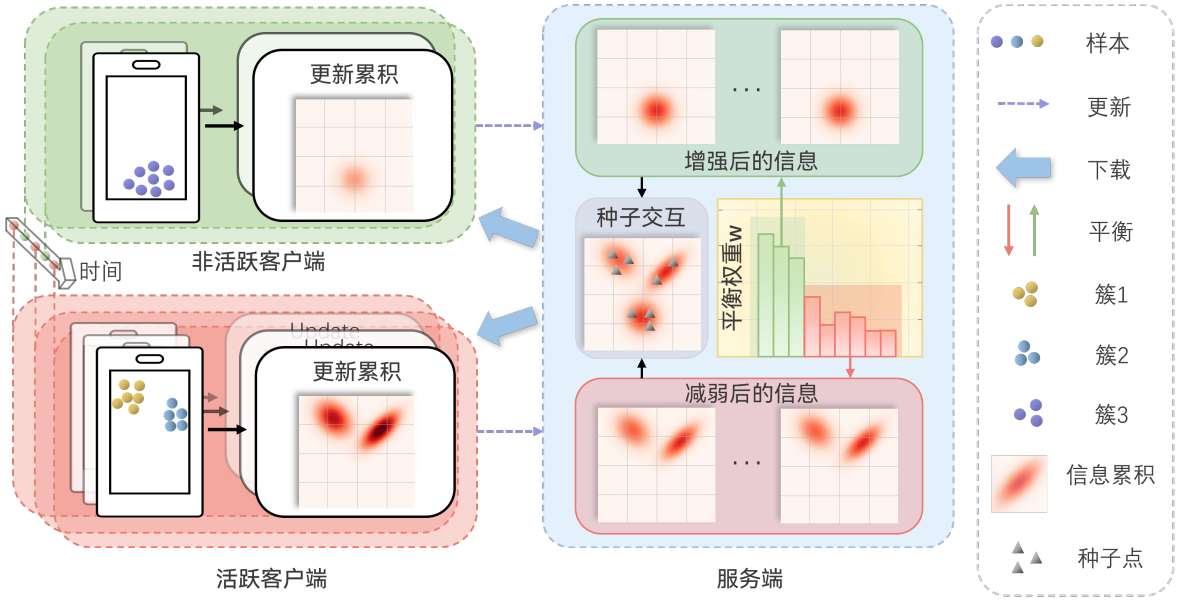


图 5-2 本文所提出的 AFCL 框架示意图

以表示为：

$$Z(\mathbf{Q},\mathbf{M})=\sum_{i=1}^{n}\sum_{l=1}^{k}q_{i,l}\boldsymbol{\Phi}(\mathbf{x}_i,\mathbf{m}_l), \tag{5.1}$$

其中，$\boldsymbol{\Phi}(\mathbf{x}_i,\mathbf{m}_l)$ 表示第 i 个对象 $\mathbf{x}_i$ 和第 l 个种子点 $\mathbf{m}_l$ 之间的欧几里得距离。所有的 k 个种子可以组织成一个矩阵 $\mathbf{M}\in\mathbb{R}^{d\times k}$，其中 $q_{i,l}$ 是矩阵 Q 中第 i 行、第 l 列的元素，满足 $\sum_{l=1}^{k}q_{i,l}=1$ 且 $q_{i,l}\in\{0,1\}$。对于联邦聚类，每个第 g 个客户端可以在其本地数据集 $\mathbf{X}^{\{g\}}$ 上执行上述聚类操作，最终目标是在在服务器端最小化全局数据集 $\mathbf{X}$ 上的目标函数，并且通过在所有客户端之间学习得到的共识种子点 $\mathbf{M}$ 来优化结果。

5.3.2 CSUA：客户端更新累积

为了保护客户端在向服务器传输全局聚类的分布信息时的隐私，一些中间量（例如种子的更新强度、聚类中心和聚类内不相似度）将在每个客户端上执行本地聚类以提取，然后上传到服务器进行种子交互。对于第 g 个客户端中的每个对象 $\mathbf{x}_i^{\{g\}}$，其所属的聚类由以下公式确定：

$$q_{i,l}=\begin{cases}1, & \text{if } l=\arg\min\limits_{r}\gamma_r\parallel\mathbf{x}_i^{\{g\}}-\mathbf{m}_r^{\{g\}}\parallel^2\\ 0, & \text{otherwise,}\end{cases} \tag{5.2}$$

其中，$\mathbf{m}_r^{\{g\}}$ 是 $\mathbf{M}^{\{g\}}$ 中的第 r 个种子，$\gamma_r^{\{g\}}$ 是通过过以下公式计算的 $\mathbf{m}_r^{\{g\}}$ 的权重：

$$\gamma_r^{\{g\}}=\frac{s_r^{\{g\}}}{\sum_{l=1}^{k}s_l^{\{g\}}}. \tag{5.3}$$

其中，$s_r^{\{g\}}$ 代表第 g 个种子 $\mathbf{m}_r^{\{g\}}$ 在一次迭代中的获胜次数，更新方式为：

$$s_c^{\{g\}} = s_c^{\{g\}} + 1, \tag{5.4}$$

对于每一个 $q_{i,c} = 1, i \in \{1, 2, ..., n^{\{g\}}\}$。

为了促进跨客户端种子的交互，我们还在本地计算种子的更新强度，但直到它们上传到服务器后才进行更新。对于获胜的种子 $\mathbf{m}_c^{\{g\}}$，其更新强度 $\mathbf{r}_{c,i}^{\{g\}}$ 由对象 $\mathbf{x}_i^{\{g\}}$ 计算：

$$\mathbf{r}_{c,i}^{\{g\}} = \eta(\mathbf{x}_i^{\{g\}} - \mathbf{m}_c^{\{g\}}), \tag{5.5}$$

其中，$\mathbf{r}_{c,i}^{\{g\}} \in \mathbf{R}_c^{\{g\}}$ 和 η 是学习率。通过计算所有 $n^{\{g\}}$ 个对象提供的 k 个种子的更新强度，我们得到 $R^{\{g\}} = \{\mathbf{R}_1^{\{g\}}, \mathbf{R}_2^{\{g\}}, ..., \mathbf{R}_k^{\{g\}}\}$ 上传到服务器。

- **讨论 1：关于隐私保护（相对于 $R^{\{g\}}$）**：AFCL 上传 $R^{\{g\}}$ 至服务器，以促进种子之间的信息交互，从而融合客户端的信息。由于每个对象提供的第 r 个种子的更新强度（将其视为赢家，即 $R^{\{g\}}$））将被上传，因此对象恢复和隐私泄漏的风险将增加。因此，现有的隐私保护技术，如同态加密[83] 和差分隐私[84] 可以在一些隐私保护要求较高的场景中结合使用。具体来说，这些技术可以用于扰动 $R^{\{g\}}$ 中每个对象提供的更新强度，同时确保合作种子选择的半径和种子的整体更新保持不变。请注意，本文更关注更强大的联邦聚类（FC），而非提高隐私保护水平。

  为了判断服务器上的收敛性，我们还计算了由种子划分的 k 个聚类的聚类中心 $\mathbf{B}^{\{g\}} = \{\mathbf{b}_1^{\{g\}}, \mathbf{b}_2^{\{g\}}, ..., \mathbf{b}_k^{\{g\}}\}$，通过以下公式进行计算：

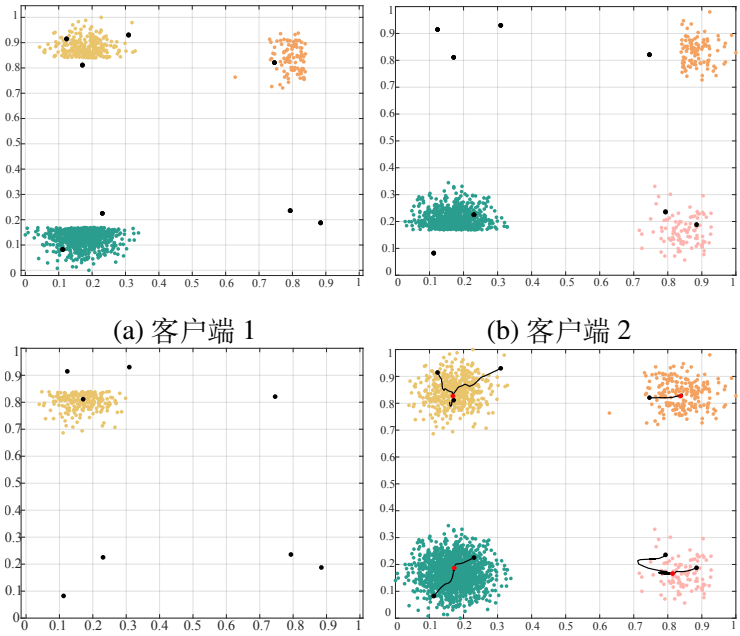
$$\mathbf{b}_r^{\{g\}} = \frac{1}{o_r^{\{g\}}} \sum_{i=1}^{n^{\{g\}}} q_{i,r} \mathbf{x}_i^{\{g\}}, \tag{5.6}$$

  其中，$o_r^{\{g\}}$ 是与 $\mathbf{m}_r^{\{g\}}$ 对应的第 r 个聚类中的对象数。基于 $\mathbf{B}^{\{g\}}$，第 r 个聚类对全局目标 Z 的贡献可以通过以下公式计算：

$$z_r^{\{g\}} = \sum_{i=1}^{n^{\{g\}}} q_{i,r} \parallel \mathbf{b}_r^{\{g\}} - \mathbf{x}_i^{\{g\}} \parallel^2, \tag{5.7}$$

  来自第 g 个客户端所有种子的贡献可以统一表示为 $\mathbf{z}^{\{g\}} = [z_1^{\{g\}}, z_2^{\{g\}}, ..., z_k^{\{g\}}]^\top$。

- **讨论 2：中间值 $\mathbf{B}^{\{g\}}$ 和 $\mathbf{z}^{\{g\}}$ 的必要性：**由于种子 M 的由于种子轨迹通常是复杂的，基于种子直接计算目标函数可能会导致目标函数值的波动。因此，在每个客户端上局部计算的中间值 $\mathbf{B}^{\{g\}}$ 和 $\mathbf{z}^{\{g\}}$ 相对稳定，可以更准确地反映当前种子的



(a) 客户端 1　(b) 客户端 2

(c) 客户端 3　(d) 聚类过程

图 5-3 FCCL 学习过程中的种子点轨迹

好坏，帮助获得平滑且更接近的全局目标函数值 Z。

5.3.3 SSSI：服务器端种子交互

在学习过程中，每个客户端的通信频率应记录 $\Theta = [\theta^{\{1\}}, \theta^{\{2\}}, .., \theta^{\{p\}}]^\top$。因此，为了平衡因异步通信引起的种子更新偏差，通信频率的权重 $\mathbf{w} = [w^{\{1\}}, w^{\{2\}}, ..., w^{\{p\}}]^\top$ 应保持。在第 g 个参与客户端中，其平衡权重通过以下公式计算：

$$w^{\{g\}} = \frac{\xi}{\xi + \frac{\theta^{\{g\}}}{\sum_{j=1}^{p} \theta^{\{j\}}}}, \tag{5.8}$$

其中，ξ 是一个超参数，用于控制平衡权重相对于通信频率 Θ 的敏感度。直观上，较大的 ξ 使平衡权重对频率的敏感度降低，频率较高的客户端将具有较低的权重，以削弱其贡献。

在接收到来自客户端的上传后，服务器初始化聚类中心及其目标贡献，带有平衡权重，计算公式为：

$$\mathbf{b}_r = \sum_{j=1}^{\bar{p}} \frac{w^{\{j\}} o_r^{\{j\}} \mathbf{b}_r^{\{j\}}}{\sum_{j=1}^{\bar{p}} o_r^{\{j\}}}, \quad z_r = \sum_{j=1}^{\bar{p}} \frac{w^{\{j\}} o_r^{\{j\}} z_r^{\{j\}}}{\sum_{j=1}^{\bar{p}} o_r^{\{j\}}}, \tag{5.9}$$

其中，$\bar{p}$ 表示参与的客户端数，$\mathbf{B} = \{\mathbf{b}_1, \mathbf{b}_2, ..., \mathbf{b}_k\}$ 和 $\mathbf{z} = [z_1, z_2, ..., z_k]^\top$ 分别表示从服务器视角汇总的聚类中心和目标贡献。相应地，服务器上的目标的近似可以推导出来，

如下所示：

$$Z(\mathbf{B},\mathbf{z})=\frac{1}{k}\sum_{l=1}^{k}\max_{r\neq l}\left(\frac{z_l+z_r}{\parallel \mathbf{b}_l-\mathbf{b}_r \parallel^2}\right), \tag{5.10}$$

这是 DBI 指数的形式[85] 它同时反映了聚类的紧凑度（分子）和聚类的分散度（分母）。

为了确保从数据中学到的种子能够共识性地最小化 Z，我们为每个种子 $\mathbf{m}_r$ 确定一个合作集：

$$C_r=\{\mathbf{m}_l \mid\mid\mid \mathbf{m}_r-\mathbf{m}_l \parallel^2 \leq \parallel \mathbf{m}_r-\mathbf{x}_i^{\{g\}} \parallel^2\} \tag{5.11}$$

并让所有 C_r 中的种子共同接收来自样本的更新，更新公式如下：

$$\mathbf{m}_u=\mathbf{m}_u+\eta(\mathbf{x}_i^{\{g\}}-\mathbf{m}_u), \tag{5.12}$$

其中 $\mathbf{m}_u \in C_r$。然而，由于隐私限制，原始样本在服务器上不可用，因此我们根据已上传到服务器的更新强度 $\mathbf{r}_{r,i}^{\{g\}}$ 交替计算前两个公式，如下所示：

$$C_r=\{\mathbf{m}_l \mid\mid\mid \mathbf{m}_r-\mathbf{m}_l \parallel^2 \leq \parallel \frac{w^{\{g\}}\mathbf{r}_{r,i}^{\{g\}}}{\eta} \parallel^2\} \tag{5.13}$$

以及：

$$\mathbf{m}_u=\mathbf{m}_u+w^{\{g\}}\mathbf{r}_{r,i}^{\{g\}}+w^{\{g\}}\eta(\mathbf{m}_r-\mathbf{m}_u) \tag{5.14}$$

其中，$w^{\{g\}}$ 用于减轻异步上传造成的种子更新偏差。直观上，较小的 $w^{(g)}$ 对应于上传频繁的客户端。因此，邻居识别半径 $\parallel w^{\{g\}}\mathbf{r}_{r,i}^{\{g\}}/\eta \parallel^2$ 会较小，以避免过度更新 C_r 中的种子。在下面公式中，$\mathbf{r}_{r,i}^{\{g\}}$ 已经是通过学习率 η 计算出来的，因此后两项与相同系数 $w^{\{g\}}\eta$ 相关，相当于通过小步更新 $\mathbf{m}_u$，使其向数据对象靠近，该数据对象提供了更新强度 $\mathbf{r}_{r,i}^{\{g\}}$。

- **讨论 3：种子间的交互：**根据前两个公式，所有 C_r 中的种子将会更接近由 m_r 表示的聚类分布，依据来自不同客户端的分布信息，这有助于促进不同客户端之间的分布完成。

5.3.4 整体 AFCL 算法

在 AFCL 算法中，在所有客户端上进行种子点的全局初始化和本地更新累积后，通过服务器在客户端之间进行充分的交互，直到 Z 收敛为止。在这一过程中（总结为算法2），客户端主要实现聚类分布学习，而服务器端负责隐私保护的分布信息融合。对于 AFCL 算法的每次学习迭代，时间复杂度为 $\mathcal{O}(kn^{\{g\}}dp+n^{\{g\}}k^2d)$，与最先进的联邦

Algorithm 2 AFCL: 异步联邦聚类算法

输入: 数据集 $\mathbf{X}^{\{1\}}, \mathbf{X}^{\{2\}}, \ldots, \mathbf{X}^{\{p\}}$，聚类数 k，参数 ξ 和 η。

for 所有客户端并行 **do**
    使用 k-means++ 初始化 k 个种子点；
    将初始化的种子点发送到服务器；
end for
根据所有客户端发送的种子点，使用 k-means++ 初始化全局种子点 $\mathbf{M}$；
repeat
    将全局种子点 $\mathbf{M}$ 发送给参与的客户端；
    for $g = 1$ **到** p **do**
        更新 $\theta^{\{g\}} = \theta^{\{g\}} + 1$；
        计算 $\mathbf{Q}^{\{g\}}, R^{\{g\}}, \mathbf{B}^{\{g\}}, \mathbf{z}^{\{g\}}$，分别根据公式 (5.2)，(5.5)，(5.6) 和 (5.7)；
        将 $\mathbf{Q}^{\{g\}}, R^{\{g\}}, \mathbf{B}^{\{g\}}, \mathbf{z}^{\{g\}}$ 上传到服务器；
    end for
    使用公式 (5.14) 和 (5.10) 计算 $\mathbf{M}$ 和 Z；
    根据公式 (5.8) 更新 $\mathbf{w}$；
until 收敛

输出: 全局种子点 $\mathbf{M}$ 和对象-簇隶属矩阵 $\mathbf{Q}$。

聚类（FC）方法相比效率较高。

5.4 实验设置与分析

本小节分为两个部分：基础实验部分和进阶实验部分，基础实验部分证明方法的有效性，考虑到联邦聚类中常见且更困难的异步通讯问题，进阶实验将突出所解决问题的挑战性。

5.4.1 基础实验设置

4 个实验类型：（1）可视化，（2）收敛性评估，（3）聚类性能评估，（4）消融实验，用于评估所提出的 FCCl 方法的竞争力。

5 个对比方法：包括一种传统方法，即 DK++[86]，和三种最先进的方法，即 FFCM-avg1、FFCM-avg2[3] 和 k-FED[1]，被选作比较。同时，我们还增加了一种名为联邦均值

表 5-2 基础实验数据集统计信息

No.	数据集	简称.	n	d	k^*
1	Synthetic Dataset 1	SD1	2300	2	4
2	Synthetic Dataset 2	SD2	2900	2	5
3	Drug Consumption	DC	1885	12	7
4	Avila	AL	10430	10	12
5	Abalone	AB	4178	8	29
6	Cancer	CC	570	31	2
7	Ecoli	EC	336	7	8
8	Seeds	SE	210	7	3
9	Parkinson	PA	195	22	2
10	Accent	AC	330	12	6
11	Sports Articles	SP	1000	59	2
12	Iris	IR	150	4	3
13	Segment	SG	2100	19	7

迁移（FMS）的联邦聚类方法，它基于原始的均值迁移方法[87] 并使用 k-means 聚类进行聚合。

13 个数据集：包括两个合成数据集和 11 个公共数据集，用于进行实验。数据集的统计信息见表5-2，所有公共数据集来自 UCI 机器学习库[88]。

3 个有效性指标：轮廓系数指数（SC）是一个传统且流行的内部指标，它同时计算聚类信息的密度和离散度 SC 的值范围是 $[-1,1]$。Calinski-Harabasz 指数（CH）是一个流行的指标，它计算聚类之间的距离与聚类内部距离的比率，值范围是 $(0,+\infty)$。需要注意的是，对于这两个指标来说，值越大表示聚类性能越好，并且它们对聚类数目不敏感。Bonferroni-Dunn（BD）检验[69] 也被应用于聚类结果的比较，用于计算关键的临界差异（CD）区间，以统计方式展示所提方法的优越性。

5.4.2 可视化

为了全面评估 FCCL 的聚类性能，图5-4展示了用于可视化的两个 2D 合成数据集。参数设置如下：对于合成数据集 1，$\mu_1=[1,1]^T$，$\mu_2=[1,5]^T$，$\mu_3=[5,5]^T$，$\mu_4=[5,1]^T$，$\sum=[0.1,0;0.1,0]$，每个聚类中的数据对象数量为 $G_1=1500$，$G_2=500$，$G_3=200$，$G_4=100$。相同的聚类用相同的颜色标记。四个分布被认为有四个真实标签。

我们使用 k-means 将数据集划分为三个子集，以创建非独立同分布（non-i.i.d）[84]

表 5-3 进阶实验数据集统计信息

No.	数据集	简称	样本数	属性数	簇数
1	Synthetic Dataset 1	SD1	2300	2	4
2	Synthetic Dataset 2	SD2	2900	2	5
3	Seeds	SE	210	7	3
4	Iris	IR	150	4	3
5	Avila	AL	10430	10	12
6	Abalone	AB	4177	7	29
7	Breast Cancer	CC	569	30	2
8	Accent	AC	329	12	6
9	Segment	SG	2100	19	7
10	Live	LI	7051	9	2
11	Parkinson	PA	197	22	2
12	Audit	AU	776	24	2
13	Transfusion	TF	748	4	2

数据，如图5-3(a)-5-3(c)所示，这些数据假定存储在三个客户端中。图5-3(d)显示了服务器中质心的分布和轨迹，以及 FCCL 的聚类结果。可以观察到，质心很好地拟合在每个真实标签的聚类中，并且在过程中一些质心在协作形成更好的聚类结果时被同质化。

5.4.3 聚类性能评估

为了进一步验证 FCCL 的有效性，我们还将 FCCL 与现有方法进行比较，在具有挑战性的非独立同分布（non-IID）场景中，使用 k-means 在合成和公共数据集上进行实验。具体来说，我们首先生成 20 个不同的数据分布集，使用 k-means 为每个数据集生成客户端数据。然后，每种方法都使用这 20 个数据集进行聚类。客户端数量设置为 5，并在此后续实验中使用。关于 SC 和 CH 的性能见表 5-4 和表 5-5。“ER” 表示方法报告错误。注意，k-FED 利用不同客户端之间的异质性，但非独立同分布数据的异质性可能会更严重，且不同客户端之间可能存在不平衡的数据分布，从而可能导致 k-FED 报告错误。最优和次优结果分别用加粗和下划线标出，平均排名行报告所有数据集的平均排名。

可以观察到，FCCL 在 SC 指数上普遍优于其他方法，表明其有效性。具体来说，FCCL 在 SC 指数上超越了几乎所有的数据集，在 SP 数据集上表现与其他方法相似，

表 5-4 在 13 个数据集上的 SC 性能的比较

数据集	FCCL	FFCM-avg1	FFCM-avg2	*k*-FED	DK++	FMS
SD1	**0.9576**	0.5063	0.5036	ER	0.5986	0.8204
SD2	**0.8478**	0.4773	0.4679	ER	0.6127	0.7455
DC	**0.5349**	0.1675	0.1476	0.2104	0.2884	0.1812
AL	**0.9750**	0.2721	0.2761	ER	0.2096	0.4056
AB	**0.5139**	0.3664	0.4863	ER	0.2314	-0.2242
CC	**0.5963**	0.2873	0.3051	0.3809	0.3778	0.4624
EC	**0.4580**	0.3442	0.4500	ER	0.2852	0.3537
SE	**0.5518**	0.4323	0.4323	0.3754	0.3229	0.3605
PA	**0.6284**	0.4419	0.4419	0.4517	0.2763	0.6016
AC	**0.6385**	0.2656	0.2358	ER	0.1831	0.0777
SP	0.5466	0.2800	0.2796	**0.5194**	0.3441	**0.5844**
IR	**0.6579**	0.5672	0.6119	0.4818	0.4955	0.4487
SG	0.5702	0.3819	0.3865	0.3117	0.3197	**0.3560**
平均排名	**1.0769**	3.8462	3.6923	4.8462	4.2308	3.3077

这表明服务器端有效地通过我们的竞争和协作学习过程，找到了质心的全局最优位置，从而有效地最小化了聚类内密度并最大化了聚类间的分散度。对于 CH 指数，某些数据集的表现可能不是最好的，如 AL、EC、SE 和 PA，但在这些数据集上 FCCL 始终表现最佳，这仍然证明了 FCCL 的竞争力。

我们还对平均排名中的结果进行了 BD 检验，并可视化了基于 CD 区间的测试结果，如图5-4所示。CD 的长度分别为 1.890 和 1.6407，在使用 α=0.05 和 α=0.1 时，对比了六种方法在 13 个数据集上的结果。可以看出，FCCL 显著优于大多数对比方法，尽管其中三个是最先进的方法。

5.4.4 消融实验

我们基于判别性的 CH 指数进行消融实验，评估聚类性能。为了验证 FCCL 核心组件的有效性，比较了三种不同版本的 FCCL。为了评估非独立同分布（non-IID）数据场景、服务器竞争过程和联邦环境的有效性，我们将它们替换为相应的独立同分布（IID）数据版本、服务器加权聚合版本以及原始版本，分别称为 FCCL-IID、FCCL-IID-Weight 和 CCL。FCCL-IID 是指每个客户端的数据是从中心数据中随机抽样的。在中央服务器中，FCCL-IID-Weighted 仅使用每个客户。

表 5-5 在 13 个数据集上的 CH 性能的比较

数据集	FCCL	FFCM-Avg1	FFCM-Avg2	*k*-FED	DK++	FMS
SD1	**19103.81**	3136.2	3140.1	ER	13933.31	17944.68
SD2	**18523.5**	1462.2	1474.6	ER	13187.23	17102.83
DC	**831.83**	402.31	349.58	480.988	683.35	18.3523
AL	7578	1607.5	1809.4	ER	**2790.2**	808.94
AB	**7910.1**	917.25	956.32	ER	3756.21	284.48
CC	**315.9**	104.05	107.41	290.533	282.03	82.42
EC	113.13	1124.8	**1159.8**	ER	101.49	102.48
SE	254.36	83.38	83.51	**261.215**	192.55	111.167
PA	64.99	79.45	79.45	68.4	40.21	**81.35**
AC	156.15	82.35	77.36	ER	112.26	67.893
SP	**455.33**	154.31	160.55	263.17	420.14	230.09
IR	**330.44**	65.58	66.27	292.93	315.16	210.75
SG	1620.8	765.2	1278.2	1122.85	1173.82	**215.883**
平均排名	**1.8462**	4.3462	3.5769	4.3077	2.8462	4.0769

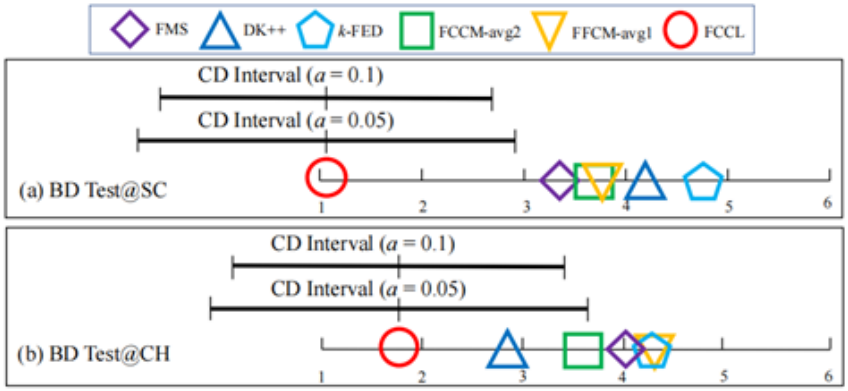


图 5-4 基于表5-4和表5-5的 BD 检验结果

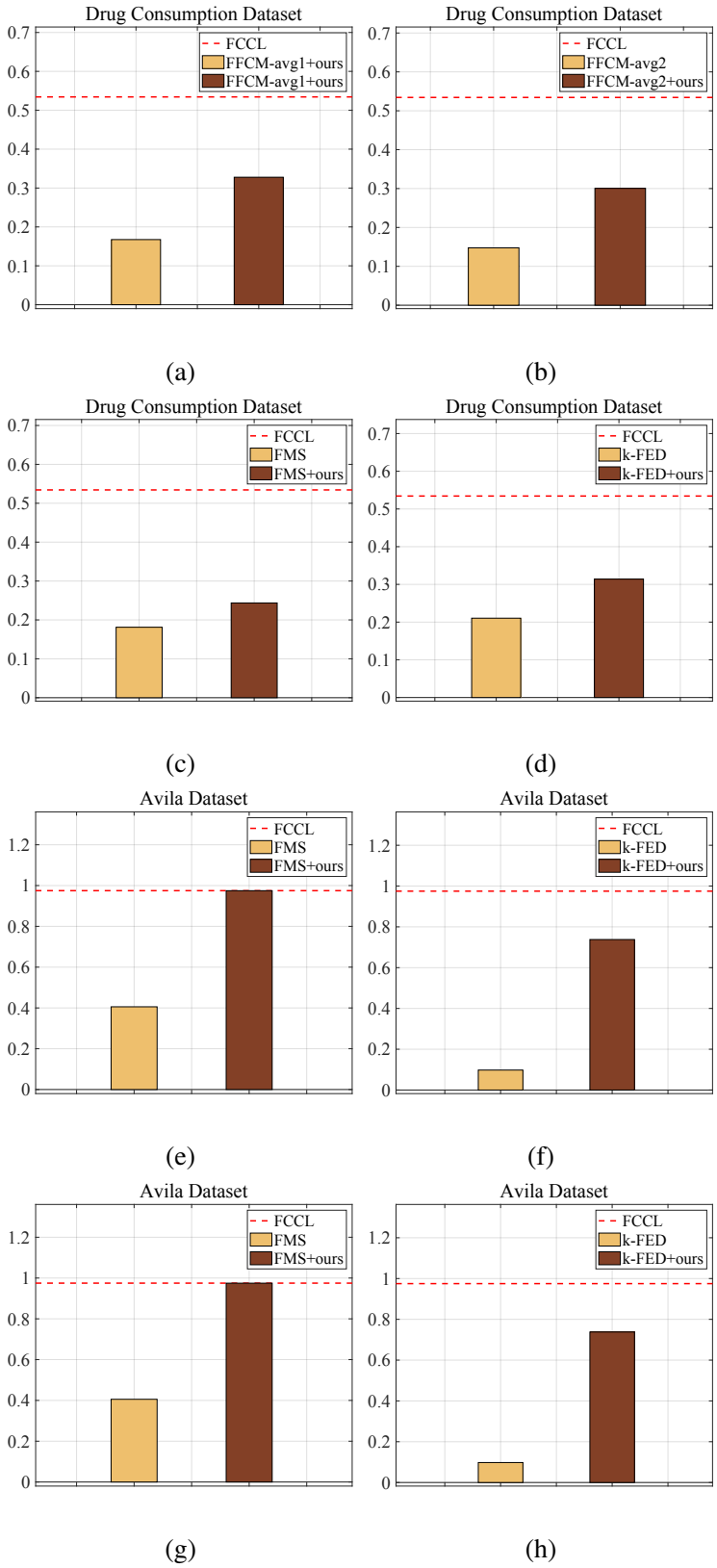


图 5-5 FCCL 不同版本在 Drug Consumption 和 Avila 数据集的聚类性能对比

表 5-6 通过 CH 指数在 13 个数据集上评价不同消融版本的 FCCL 的性能

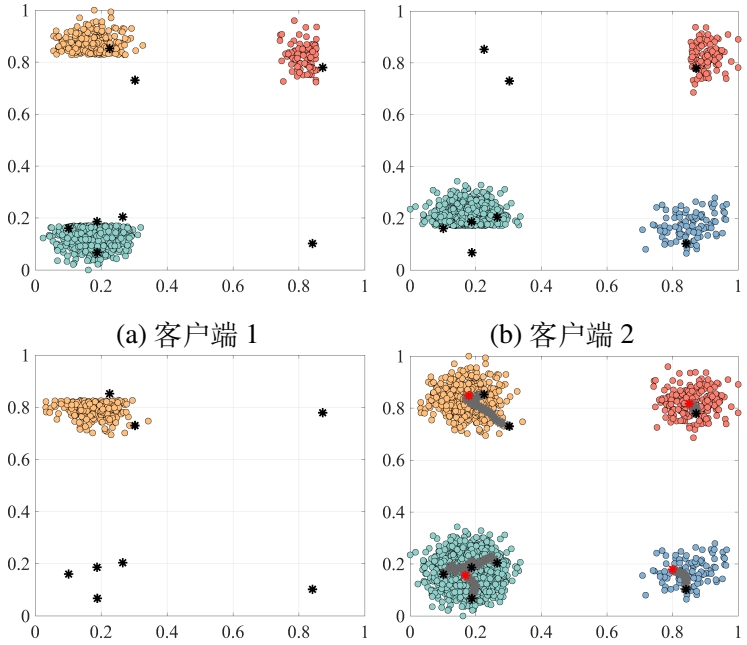
CH	FCCL	FCCL-IID	FCCL-IID-Weight	CCL
SD1	19947	19921	20556.3	19947
SD2	19696.8	19181.1	18415.42	16994.74
DC	**831.83**	**831.33**	519.071	459.33
AL	75.78	75.78	1651.1	**6331.7**
AB	7910.1	7422.3	2997.4	3684.6
CC	315.9	**326.9**	288.95	239.45
EC	113.13	121.07	116.43	**134.27**
SE	254.36	**271.28**	234.49	185.458
PA	**64.99**	65.94	64.58	64.125
AC	156.15	115.15	89.28	125.91
SP	455.33	336.14	356.25	**268.3**
IR	**330.44**	329.57	147.98	219.1
SG	1620.8	**1663.12**	949.33	336.82
平均排名	**1.8462**	2.1154	2.9231	3.1154

从表5-6可以观察到，FCCL、FCCL-IID、FCCL-IID-Weight 和 CCL 的总体表现按顺序下降，这直观地说明了我们方法在非 IID 条件下的竞争力、服务器竞争过程的有效性以及联邦学习环境的有效性。具体来说，FCCL 和 FCCL-IID 之间的差距很小，表明 FCCL 对于使用竞争学习的每个客户端的数据分布具有鲁棒性。FCCL-IID 通常优于 FCCL-IID-Weight，这直观地验证了 FCCL 核心组件的有效性，即中央服务器中的竞争过程。由于 FCCL-IID-Weight 和 CCL 的整体表现接近，这表明我们的联邦版本和原始版本的表现相似，且验证了我们的联邦学习环境的有效性。

5.4.5 进阶实验设置

6 种实验：（1）可视化：直观展示所提出的 AFCL 的学习过程；（2）收敛性评估：记录每次学习迭代的目标函数值，以说明 AFCL 的收敛性和效率；（3）聚类性能评估：我们将 AFCL 与传统和最先进的对比方法进行比较，以展示 AFCL 的优越性；由于篇幅限制，其余三项实验，即显著性检验、消融研究、执行时间评估以及更详细的实验设置，均在在线附录`https://github.com/Yunfan-Zhang/AFCL`中提供。

6 个对比方法：DK++[86] 是一种传统的分布式学习方法，符合 FL 的设置。5 个最先进的方法，即迭代学习方法 FFCM-avg1，FFCM-avg2[3]，和 FedSC[7]，以及单次学习



(a) 客户端 1　(b) 客户端 2

(c) 客户端 3　(d) 聚类过程

图 5-6 AFCL 服务器上的种子点及其轨迹

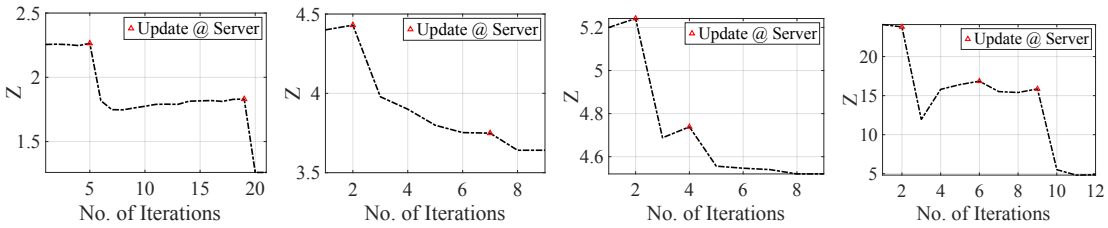
方法 k-FED[1] 和 Fed-SC[6]，被选作对比。它们的所有超参数（如有）均根据相应的源文献设置。

13 个数据集：包括两个高斯球面合成数据集和 11 个公共真实数据集，数据集来自 UCI 机器学习库[88]，其统计信息见表5-3。所有真实数据集通过删除缺失值的对象并使用最小-最大标准化进行了预处理。

3 个有效性指标：轮廓系数指数（SC）[12]，Calinski-Harabasz 指数（CH）[89]，以及 Bonferroni-Dunn（BD）检验[90] 被选为性能评估指标。SC 和 CH 的值分别位于 [-1,1] 和 $(0,+\infty)$ 区间，其中较高的值表示更好的聚类性能。这两个内部指标对聚类数目不敏感，有助于公平比较 AFCL 的聚类性能，AFCL 使用其学到的真实 k 和预设 $k*$，如表5-3所示。BD 检验用于对比方法性能排名进行检验，验证 AFCL 是否显著优于对比方法。

5.4.6 可视化

为了直观验证 AFCL 的有效性，我们将 SD1 划分为三个子集，创建三个极其非独立同分布（non-IID）的客户端，如图5-6(a)-5-6(c)所示。图5-6(d)显示了服务器上的全局数据分布和种子更新轨迹。可以观察到，即使这三个客户端具有完全不重叠的分布，AFCL 仍然能够适当地学习一组种子来表示全局的聚类分布。轨迹还表明，AFCL 的种子更新机制能够有效促进种子之间的交互，尽管存在由异步客户端上传的更新信息不



(a) SD1 数据集　(b) IR 数据集　(c) SE 数据集　(d) AL 数据集

图 5-7 AFCL 目标函数在 (a) SD1、(b) IR、(c) SE 和 (d) AL 数据集上的值

平衡的情况。经过一定数量的学习迭代后，冗余的种子被同质化，即它们重叠在几个显著聚类的中心。这直观地展示了 AFCL 在自主选择聚类数目方面的能力。

为了展示 AFCL 的收敛效率，我们在图5-7中绘制了四个数据集的目标函数值。从中可以观察到，AFCL 在大多数情况下在大约 10 次迭代后迅速收敛。此外，目标函数在服务器更新后总是经历一个陡峭的下降，这证明了所设计的种子交互机制非常有效。值得注意的是，由于仅允许在客户端和服务器之间传递有限的统计数据，因此不能保证严格的梯度下降，因此图5-7中的收敛曲线并不是单调递减的。这种效应对于联邦聚类（FC）是合理的，因为聚类目标可以视为在不同客户端和服务器上的异质性。

5.4.7 聚类性能评估

在非独立同分布（non-IID）和异步场景下，比较了 AFCL 和现有的联邦聚类（FC）方法的聚类性能。对于每个数据集，我们使用每种对比方法进行 20 次聚类，并报告平均性能。在 20 次实验中，对于每次实验，我们首先使用 $k = 5$ 实施 k-means，将整个数据集划分为五个子集，以模拟极端非重叠的非独立同分布（non-IID）客户端。然后，为了模拟客户端的异步参与，我们随机为每个客户端设置不同的参与概率，以控制它们在学习过程中每次迭代中的上传。由于 AFCL 不需要 k^*，种子的初始数量是在范围 $[k^*, 2k^*]$ 内随机选择的。最后，报告了 SC 和 CH 指标下的聚类性能。在上述设置下的结果见表 3-6 和表 3-7。最佳结果和次佳结果通过加粗和下划线高亮显示。‘Ave. Rank’行报告了不同方法在所有数据集上的平均排名。

可以观察到，AFCL 在一般情况下优于所有对比方法，表明其在异步联邦聚类中的优势。具体来说，AFCL 在大多数数据集上在 SC 指标上表现最好，除了 PA 数据集，在该数据集上 AFCL 仍然表现第二好。这是因为 AFCL 能够有效地最小化聚类内的不相似度，并最大化聚类间的分散度，从而搜索全局最优种子。对于 CH 指标，尽管 AFCL

表 5-7 SC 对所有 13 个数据集的聚类性能评估

数据集	DK++	*k*-FED	FFCM-avg1	FFCM-avg2	Fed-SC	FedSC	AFCL
SD1	0.5986±0.18	0.8494±0.00	0.5063±0.02	0.5036±0.02	0.8261±0.09	0.8539±0.00	**0.9714±0.00**
SD2	0.6127±0.11	0.7699±0.00	0.4773±0.03	0.4679±0.03	0.6005±0.14	0.7477±0.00	**0.8571±0.00**
SE	0.3229±0.00	0.3754±0.04	0.4323±0.05	0.4323±0.05	0.3619±0.00	0.3774±0.00	**0.5033±0.09**
IR	0.4955±0.01	0.4818±0.02	0.5672±0.02	0.6119±0.21	0.5384±0.04	0.5719±0.00	**0.6386±0.06**
AL	0.2096±0.02	0.0979±0.03	0.2721±0.09	0.2761±0.07	0.4422±0.05	0.3187±0.00	**0.6138±0.38**
AB	0.2314±0.02	0.1853±0.01	0.3664±0.34	0.4863±0.26	0.3009±0.03	0.3865±0.06	**0.5005±0.11**
CC	0.3778±0.00	0.3809±0.02	0.2873±0.05	0.3051±0.04	0.3672±0.04	0.3747±0.00	**0.5916±0.04**
AC	0.1831±0.02	0.0992±0.01	0.2656±0.13	0.2358±0.13	0.1376±0.02	0.1922±0.00	**0.4851±0.26**
SG	0.3197±0.02	0.3117±0.00	0.3819±0.11	0.3865±0.10	0.2677±0.04	0.2935±0.00	**0.6086±0.12**
LI	0.8241±0.00	0.8256±0.01	0.4805±0.16	0.4239±0.23	0.7980±0.10	0.8252±0.00	**0.8967±0.01**
PA	0.2763±0.00	0.4517±0.03	0.4419±0.15	0.4419±0.15	0.2443±0.00	**0.5138±0.00**	0.4903±0.11
AU	0.4046±0.05	0.3601±0.04	0.1440±0.08	0.2239±0.02	0.3411±0.07	0.4296±0.00	**0.5191±0.07**
TF	0.4935±0.00	0.5269±0.01	0.6402±0.16	0.6402±0.16	0.5172±0.25	0.5390±0.00	**0.6813±0.03**
平均排名	4.6154	4.2308	4.4231	4.1923	5.4615	4.0000	**1.0769**

表 5-8 CH 对所有 13 个数据集的聚类性能评估

数据集	DK++	*k*-FED	FFCM-avg1	FFCM-avg2	Fed-SC	FedSC	AFCL
SD1	13933.3423	18933.7121	3136.2342	3140.0656	18901.0798	18958.6529	**19482.8610**
SD2	13187.1700	16936.0195	1462.2227	1474.5426	1575.4188	16913.4103	**17200.3823**
SE	192.6124	**251.1952**	83.3635	83.3635	193.6615	206.0449	230.9555
IR	**315.2151**	232.9987	65.6174	66.3026	211.5583	232.9386	310.7035
AL	1524.2576	1559.4321	1607.4565	1809.4353	2260.6359	1524.2576	**4880.6220**
AB	3756.1887	3791.0247	3821.2626	3890.4190	3072.4473	3244.3141	**5906.3378**
CC	202.4573	**290.0258**	104.0924	107.4325	222.6533	231.4785	231.5775
AC	112.3225	80.9425	82.4193	77.3857	73.2801	99.9817	**140.7156**
SG	973.7952	1121.9421	765.2036	1278.2297	781.12178	1057.1972	**1541.6809**
LI	5013.7432	5526.4145	1050.4084	806.5331	4075.7079	3608.0879	**6090.9982**
PA	40.2442	68.3886	79.5121	79.5121	56.2801	**83.3207**	78.1141
AU	304.9842	251.1567	65.7637	107.1965	201.6834	**433.3965**	306.7529
TF	393.1126	540.1820	436.6372	436.6372	420.2884	518.4788	**651.6017**
平均排名	4.5000	3.0769	5.5769	4.8846	5.0000	3.4231	**1.5385**

在某些数据集上表现第二好，例如 SE、IR、CC 和 AU，但在这些数据集上，AFCL 的最佳对比方法表现有所不同，而 AFCL 在大多数情况下仍保持竞争力。更具体来说，在 AFCL 无法获得最佳性能的情况下，AFCL 与表现最佳的对比方法之间的性能差距通常很小，这证明了 AFCL 在不同数据集上的有效性和鲁棒性。

5.5 本章小结

本文提出了一种新的联邦聚类（FC）方法，称为 AFCL，用于挖掘异构数据下的全局聚类分布。AFCL 将联邦聚类（FC）推进到一个更具挑战性但更现实的场景，即客户端可以异步参与到客户端到服务器的上传中，所有客户端和服务器的分布都可以是极端非独立同分布（non-IID）的，而无需知道“真实”的聚类数量。AFCL 通过采用客户端到种子的信息融合框架来实现这一目标，框架使得种子点能够在服务器上协同工作，完成客户端的非 IID 分布，并自动学习去除冗余种子。同时，还设计了一个平衡机制，以缓解异步参与客户端上传的更新信息的不均匀性。因此，即使通信极其异步且客户端分布完全不同，AFCL 也能有效地基于从客户端接收到的聚合更新强度，勾画出全局的聚类分布。综合实验已证明了 AFCL 的有效性。尽管 AFCL 优越，但仍存在一些值得注意的潜在局限性。也就是说，我们假设 FC 仅适用于纯数值数据且客户端数量较少。下一个有前景的方向是将 FC 扩展到包含数值和分类属性的数据集，这些数据集分布在大量客户端上。

第六章　应用价值

在数据驱动决策的时代，聚类技术作为挖掘数据内在模式的核心工具，正以“场景适配性”为核心，在医疗与军事两大战略领域实现价值重构——前者需破解“隐私敏感 + 多模态复杂 + 少样本局限”的临床难题，后者则需应对“高维动态 + 异质关联 + 对抗干扰”的战场挑战。传统聚类方法在异质数据融合、信息提取完备性及动态适配性上存在显著局限：或因无法统一异质特征距离度量导致信息损失，或因局部信息提取不充分降低联邦聚类精度，或因缺乏自适应机制难以捕捉分布微妙变化。本项目构建的综合聚类系统，通过“统一异质距离度量”“完备信息提取”“自适应更新机制”三大技术突破，突破了传统方法的单一性局限，在医疗精准诊疗与军事智能决策中展现出显著的场景赋能能力。以下从医疗与军事两大领域展开具体分析。

6.1 医疗领域——多模态异质数据的精准聚类突破

医疗场景的核心需求是通过病例数据聚类实现疾病精准分型与诊疗方案优化，但其数据具有典型的多模态异质性：包含数值属性（如体温、白细胞计数）、分类属性（如症状类型、接触史）及影像/文本（如 CT 图像、主诉描述）。传统方法在异质数据处理上面临双重困境：一方面，数值与分类特征的距离度量割裂（如欧氏距离仅适用于数值特征，汉明距离仅适用于类别特征），导致多模态特征间的耦合信息严重丢失；另一方面，联邦医疗场景中（如多医院协作分析），现有方法仅上传局部簇中心等简化信息，难以完整保留全局分布特征，在非独立同分布数据场景下聚类精度显著下降。此外，新发疾病等少样本场景中，传统方法因参数固定无法适应分布动态漂移，易遗漏小样本病例亚型。以下是本作品的解决方案：

统一异质数据距离度量，降低融合损失：针对多模态特征，采用层次耦合编码策略，将数值、分类及影像特征统一为异质数据图。具体通过值级耦合（计算类别特征值的出现概率）、特征级耦合（基于条件概率分布捕捉特征间依赖）及对象级耦合（基于地球移动距离统一异质特征的距离计算），构建包含多模态关联的图结构。相较于传统方法仅用单一距离度量（如欧氏距离 + 汉明距离）导致的信息割裂，层次耦合编码策略通过图结构保留了特征间的潜在关联，例如将“持续高热”与“干咳”“乏力”等症状的关联关系显式编码，避免了因距离度量不统一导致的亚型误判。

数据信息完备提取，提升联邦聚类精度：针对联邦医疗数据的隐私敏感与非独立

同分布特性，设计客户端更新积累机制。各医院本地计算对象与种子点的差异信息（更新强度）并上传，而非仅传输簇中心。服务器端通过种子交互机制，融合多源更新强度，补全非重叠子簇的全局分布。

自适应更新机制，捕捉分布微妙变化：针对少样本、分布动态漂移场景（如新发传染病早期病例），引入平衡机制调整客户端上传频率权重。对于上传频繁的医院，降低其更新强度权重，缓解异步上传导致的分布偏差；同时通过种子点的动态竞争与协作，自动消除冗余种子，敏锐捕捉病例亚型的微妙变化。

6.2 军事领域——高维动态数据的智能决策赋能

军事场景的核心需求是通过战场数据聚类实现态势感知与装备优化，但其数据具有“高维性”“动态性”与“对抗干扰性”：雷达回波（高维数值）、地形类型（类别）、敌方轨迹（动态对抗）等多源数据交织。传统方法在高维异质数据处理中存在显著缺陷：高维数值特征（如雷达多频段回波）因噪声冗余易被忽略，类别特征（如地形类型）与数值特征的融合不足导致关联信息丢失；异步上传的多源数据（如前线传感器、侦察设备）因非独立同分布特性难以融合，聚类精度低下；敌方战术动态漂移（如突然转向、伪装）时，静态参数方法（如固定 k 值的谱聚类）无法快速调整簇结构，易误判威胁等级。本系统通过三大技术创新实现高效聚类：

统一异质数据距离度量，降低融合损失：针对高维数值（雷达回波）与类别（地形类型）特征，采用四元数图表示学习。将高维数值特征降维为四元数空间的多视图表示（实部、虚部 i、j、k），类别特征通过层次耦合编码嵌入图结构，利用四元数的哈密顿积实现异质特征的高效旋转与耦合。

数据信息完备提取，提升联邦聚类精度：针对多源异步上传的战场数据（如前线传感器、侦察设备），采用异步联邦聚类框架。客户端上传更新强度（包含对象与种子点的差异）而非局部簇中心，服务器端通过种子协作机制融合多源信息，补全非重叠子簇的全局分布。。

自适应更新机制，应对动态对抗变化：针对敌方战术动态漂移（如突然转向、伪装），设计自增长图与快速层次合并策略。自增长图动态分配密度敏感神经元，优先覆盖小目标分布区域（如“低慢小目标”的稀疏轨迹），避免传统微簇合并（$O(m^2)$ 复杂度）遗漏小簇；同时通过在线联邦聚类实时更新种子点（如增量 DBSCAN 捕捉轨迹漂

移)，仅传输变化量至指挥中心，动态调整全局簇中心。

从医疗领域的多模态异质数据精准分型，到军事领域的高维动态数据威胁预警，本聚类系统通过“统一异质距离度量”“完备信息提取”“自适应更新机制”三大技术突破，为两大战略领域提供了“精准、安全、高效”的聚类解决方案。未来，随着数据复杂性与对抗性的进一步提升，系统将持续优化多模态融合、边缘计算及动态适配能力，推动聚类技术在更深度场景中的价值释放。

第七章　结论与展望

在本章节中，我们将总结本文的研究贡献，包括提出的方法，解决的现实挑战。之后，我们也说明了本研究的未来发展方向与仍然有待解决的问题。

研究结论

随着聚类分析技术的发展，越来越多的领域将此项技术应用其中。分布式大数据聚类分析作为该研究领域中的一个难点集合，所涵盖主要难题包括：数据整合壁垒高、指标融合分析难以及非平衡簇检测弱等，使得分布式大数据聚类分析算法在现实中的落地充满挑战。因此本文围绕该背景下分布式大数据聚类分析任务展开研究，主要工作包括如下几点：

- **异质特征数据距离新度量**：针对信息融合损耗大，提出了一种新的属性图聚类方法。它利用四元数高效汉密尔顿积的优势，同时解决了限制聚类性能的过度平滑和过度支配问题。通过广义设计，形成了一个由可学习的 FVP 和 QGE 组成的表示学习模型，用于聚类友好的表示学习。

- **多源数据联邦聚类新策略**：针对联邦聚类精度低，本策略基于多粒度竞争学习的分类数据处理策略，通过自动学习不同粒度的对象划分，有效揭示了分类数据中复杂的嵌套多粒度簇效应；海融合多粒度结果，显著提升了聚类准确性。实验结果表明，该策略在多个真实分类数据集上性能卓越，在聚类准确率、调整兰德指数等指标上优于多种对比方法，且具有良好的可扩展性和稳定性。

- **动态环境持续学习新模型**：针对知识更新时效差，本架构通过采用异步更新平衡和漂移感知网络来实现这一目标，框架缓解了异步参与客户端上传的更新信息的不均匀性，使得种子点能够在服务器上协同工作，共同学习客户端的非独立同分布数据信息。因此，即使通信极其异步且客户端分布完全不同，本方法也能有效地基于从客户端接收到的聚合更新强度，持续增强模型性能，并勾画出全局的聚类分布。

未来研究展望

本文深入研究了复杂现实环境下分布式大数据聚类的关键技术，取得了一些突破性的成果，但是面对多变的实际环境，仍存在诸多具有挑战性的问题亟需解决：

- 通过异质特征数据距离新度量，虽然异质数据间的信息差异得到了极大的弥合，但其表征并没有专门聚焦于大数据聚类中，这表明方法对聚类精度的提升还有待更充分挖掘。

- 虽然使用多源数据联邦聚类新策略提升了检测精度。但仍然需要优化算法对复杂数据的分析效率。例如，探索将两种策略相结合，以处理同时包含多种数据类型且分布复杂的数据。

- 该动态环境赤血学习新模型虽然极大提升了适应非独立同分布和异步通讯场景的能力，仍然只适用于纯数值型数据，对异质数据分析任务的契合程度仍然有待研究。

在未来，随着科技的不断发展和深度学习技术的不断提升，联邦大数据聚类分析将在更多的领域中得到应用。我们更需要考虑更复杂场景下的情况，并对实时性提出更高地要求，这不仅需要设计更加高效的算法，也需要考虑系统的实现和硬件资源的限制。因此，未来的研究方向主要包括提高大数据聚类分析的准确率和鲁棒性以及实时性，使其更适用于实际应用场景，并通过深入研究来解决不断涌现地现实挑战。

参考文献

[1] DENNIS D K, LI T, SMITH V. Heterogeneity for the win: One-shot federated clustering [C]//2021 International Conference on Machine Learning. 2021: 2611-2620.

[2] PEDRYCZ W. Federated fcm: Clustering under privacy requirements[J]. IEEE Transactions on Fuzzy Systems, 2022, 30(8): 3384-3388.

[3] STALLMANN M, WILBIK A. Towards federated clustering: A federated fuzzy c-means algorithm (ffcm)[Z]. 2022.

[4] HU X, QIN J, SHEN Y, et al. An efficient federated multi-view fuzzy c-means clustering method[J]. IEEE Transactions on Fuzzy Systems, 2023: 1886-1899.

[5] DING S, LI C, XU X, et al. Horizontal federated density peaks clustering[J]. IEEE Transactions on Neural Networks and Learning Systems, 2023: 1-10.

[6] XIE S, WU Y, LIAO K. Fed-sc: One-shot federated subspace clustering over high-dimensional data[C]//2023 International Conference on Data Engineering. 2023: 2905-2918.

[7] QIAO D, DING C, FAN J. Federated spectral clustering via secure similarity reconstruction[C]//2024 Advances in Neural Information Processing Systems: Vol. 36. 2024: 58520-58555.

[8] CHEUNG Y M, ZENG H. Local kernel regression score for selecting features of high-dimensional data[J]. IEEE Transactions on Knowledge and Data Engineering, 2009, 21 (12): 1798-1802.

[9] WU J, YANG Y, LIU H, et al. Unsupervised graph association for person re-identification [C]//2019 International Conference on Computer Vision. 2019: 8321-8330.

[10] ZHANG Y, CHEUNG Y M, TAN K C. A unified entropy-based distance metric for ordinal-and-nominal-attribute data clustering[J]. IEEE Transactions on Neural Networks and Learning Systems, 2020, 31(1): 39-52.

[11] WANG B, YANG Y, WU J, et al. Self-similarity driven scale-invariant learning for weakly supervised person search[C]//2023 International Conference on Computer Vision. 2023: 1813-1822.

[12] ROUSSEEUW P J. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis[J]. Journal of Computational and Applied Mathematics, 1987, 20: 53-65.

[13] THORNDIKE R L. Who belongs in the family?[J]. Psychometrika, 1953, 18(4): 267-276.

[14] ZHANG Y, ZOU R, ZHANG Y, et al. Adaptive micro partition and hierarchical merging for accurate mixed data clustering[J]. Complex & Intelligent Systems, 2025, 11: 1-14.

[15] PENG M, WU Y, ZHANG Y, et al. Weighted density for the win: Accurate subspace density clustering[C]//2025 International Conference on Acoustics, Speech and Signal Processing. 2025: 1-5.

[16] ESTER M, KRIEGEL H P, SANDER J, et al. Density-based spatial clustering of applications with noise[C]//1996 International Conference on Knowledge Discovery and Data Mining. 1996: 1-5.

[17] CHEUNG Y M. A competitive and cooperative learning approach to robust data clustering[C]//Neural Networks and Computational Intelligence. 2004: 131-136.

[18] CAI S, ZHANG Y, LUO X, et al. Robust categorical data clustering guided by multigranular competitive learning[C]//2024 International Conference on Distributed Computing Systems. 2024: 288-299.

[19] HU L, JIANG M, LIU Y, et al. Significance-based categorical data clustering[Z]. 2022: 1-17.

[20] KIPF T N, WELLING M. Semi-supervised classification with graph convolutional networks[C]//Proceedings of the 5th International Conference on Learning Representations. 2017.

[21] WANG C, PAN S, HU R, et al. Attributed graph clustering: A deep attentional embedding approach[C]//Proceedings of the 28th International Joint Conference on Artificial Intelligence. 2019: 3670-3676.

[22] LIN D. An information-theoretic definition of similarity[C]//Proceedings of the 15th International Conference Machine Learning. 1998: 296-304.

[23] ZHANG Y, CHEUNG Y M. Exploiting order information embedded in ordered cate-

gories for ordinal data clustering[J]. Springer, 2018, 11177: 247-257.

[24] ZHANG Y, CHEUNG Y M. Exploiting order information embedded in ordered categories for ordinal data clustering[C]//Proceedings of the 24th International Symposium on Methodologies for Intelligent Systems: Vol. 11177. 2018: 247-257.

[25] IENCO D, PENSA R G, MEO R. Context-based distance learning for categorical data clustering[C]//Proceedings of the 8th International Symposium on Intelligent Data Analysis: Vol. 5772. 2009: 83-94.

[26] ZHANG Y, CHEUNG Y M. A new distance metric exploiting heterogeneous interattribute relationship for ordinal-and-nominal-attribute data clustering[J]. IEEE Transactions on Cybernetics, 2020, 52(2): 758-771.

[27] MANDROS P, KALTENPOTH D, BOLEY M, et al. Discovering functional dependencies from mixed-type data[C]//Proceedings of the 26th Conference on Knowledge Discovery and Data Mining. 2020: 1404-1414.

[28] ZHANG Y, CHEUNG Y M. An ordinal data clustering algorithm with automated distance learning[C]//Proceedings of the 34th AAAI Conference on Artificial Intelligence: Vol. 34. 2020: 6869-6876.

[29] ZHANG Y, CHEUNG Y. Graph-based dissimilarity measurement for cluster analysis of any-type-attributed data[J]. IEEE Transactions on Neural Networks and Learning Systems, 2023, 34(9): 6530-6544.

[30] QIAN Y, LI F, LIANG J, et al. Space structure and clustering of categorical data[J]. IEEE Transactions on Neural Networks Learning Systems, 2016, 27(10): 2047-2059.

[31] HAMERLY G, ELKAN C. Learning the k in k-means[C]//Proceedings of the 16th International Conference on Neural Information Processing Systems. 2003: 281-288.

[32] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. Imagenet classification with deep convolutional neural networks[C]//Proceedings of the 26th International Conference on Neural Information Processing Systems. 2012: 1106-1114.

[33] KIPF T N. Variational graph auto-encoders[Z]. 2016.

[34] VINCENT P, LAROCHELLE H, BENGIO Y, et al. Extracting and composing robust

features with denoising autoencoders[C]//Proceedings of the 25th International Conference on Machine Learning. 2008: 1096-1103.

[35] BOWMAN S R, VILNIS L, VINYALS O, et al. Generating sentences from a continuous space[Z]. 2015.

[36] PAN S, HU R, LONG G, et al. Adversarially regularized graph autoencoder for graph embedding[C]//Proceedings of the 27th International Joint Conference on Artificial Intelligence. 2018: 2609-2615.

[37] XU Y, ZHU X, XU H, et al. Quaternion convolutional neural networks[C]//Proceedings of the 15th European Conference on Computer Vision: Vol. 11212. 2018: 645-661.

[38] PARCOLLET T, ZHANG Y, MORCHID M, et al. Quaternion convolutional neural networks for end-to-end automatic speech recognition[C]//Proceedings of the 19th Conference of the International Speech Communication Association. 2018: 22-26.

[39] ZHENG Z, HUANG G, YUAN X, et al. Quaternion-valued correlation learning for few-shot semantic segmentation[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2023, 33(5): 2102-2115.

[40] GAUDET C J, MAIDA A S. Deep quaternion networks[C]//Proceedings of the International Joint Conference on Neural Networks. 2018: 1-8.

[41] ZHANG H, LI P, ZHANG R, et al. Embedding graph auto-encoder for graph clustering [J]. IEEE Transactions on Neural Networks and Learning Systems, 2022.

[42] YANG X, LIU Y, ZHOU S, et al. Cluster-guided contrastive graph clustering network [C]//Proceedings of the 37th AAAI Conference on Artificial Intelligence. 2023: 10834-10842.

[43] YANG X, TAN C, LIU Y, et al. Convert: Contrastive graph clustering with reliable augmentation[C]//Proceedings of the 31st ACM International Conference on Multimedia. ACM, 2023: 319-327.

[44] SEN P, NAMATA G, BILGIC M, et al. Collective classification in network data[J]. AI Magazine, 2008, 29(3): 93-106.

[45] BO D, WANG X, SHI C, et al. Structural deep clustering network[C]//Proceedings of

the 29th Web Conference. 2020: 1400-1410.

[46] YANG C, LIU Z, ZHAO D, et al. Network representation learning with rich text information[C]//Proceedings of the 24th International Joint Conference on Artificial Intelligence. 2015: 2111-2117.

[47] LIU Y. A survey of deep graph clustering: Taxonomy, challenge, and application[Z]. 2022.

[48] TU W, ZHOU S, LIU X, et al. Deep fusion clustering network[C]//Proceedings of the 35th AAAI Conference on Artificial Intelligence. 2021: 9978-9987.

[49] SHI J, MALIK J. Normalized cuts and image segmentation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000, 22(8): 888-905.

[50] ZHANG Y, CHEUNG Y. Learnable weighting of intra-attribute distances for categorical data clustering with nominal and ordinal attributes[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022, 44(7): 3560-3576.

[51] HUANG Z. Clustering large data sets with mixed numeric and categorical values[C]// Proceedings of the 1st Pacific-Asia Conference on Knowledge Discovery and Data Mining. 1997: 21-34.

[52] HAJ KACEM M A B, N' CIR C E B, ESSOUSSI N. Mapreduce-based k-prototypes clustering method for big data[C]//2015 IEEE International Conference on Data Science and Advanced Analytics (DSAA). 2015: 1-7.

[53] ALAMURI M, SURAMPUDI B R, NEGI A. A survey of distance/similarity measures for categorical data[C]//2014 International Joint Conference on Neural Networks (IJCNN). 2014: 1907-1914.

[54] LE S Q, HO T B. An association-based dissimilarity measure for categorical data[J]. Pattern Recognition Letters, 2005, 26(16): 2549-2557.

[55] AHMAD A, DEY L. A method to compute distance between two categorical values of same attribute in unsupervised learning for categorical data set[J]. Pattern Recognition Letters, 2007, 28(1): 110-118.

[56] IENCO D, PENSA R G, MEO R. From context to distance: Learning dissimilarity for

categorical data clustering[J]. ACM Transactions on Knowledge Discovery from Data, 2012, 6(1): 1-25.

[57] HUANG Z. A fast clustering algorithm to cluster very large categorical data sets in data mining[J]. DMKD, 1997, 3(8): 34-39.

[58] GUHA S, RASTOGI R, SHIM K. Rock: A robust clustering algorithm for categorical attributes[J]. Information Systems, 2000, 25(5): 345-366.

[59] JIA H, CHEUNG Y M. Subspace clustering of categorical and numerical data with an unknown number of clusters[J]. IEEE Transactions on Neural Networks and Learning Systems, 2017, 29(8): 3308-3325.

[60] MOUSAVI E, SEHHATI M. A generalized multi-aspect distance metric for mixed-type data clustering[J]. Pattern Recognition, 2023, 138: 109353.

[61] OSKOUEI A G, BALAFAR M A, MOTAMED C. Fkmawcw: Categorical fuzzy k-modes clustering with automated attribute-weight and cluster-weight learning[J]. Chaos, Solitons & Fractals, 2021, 153: 111494.

[62] ZHANG Y, CHEUNG Y. Graph-based dissimilarity measurement for cluster analysis of any-type-attributed data[J]. IEEE Transactions on Neural Networks and Learning Systems, 2022, 34(9): 6530-6544.

[63] JIA H, CHEUNG Y M, LIU J. A new distance metric for unsupervised learning of categorical data[J]. IEEE Transactions on Neural Networks and Learning Systems, 2016, 27(5): 1065-1079.

[64] JIAN S, CAO L, LU K, et al. Unsupervised coupled metric similarity for non-iid categorical data[J]. IEEE Transactions on Knowledge and Data Engineering, 2018, 30(9): 1810-1823.

[65] ZHANG Y, CHEUNG Y M. A new distance metric exploiting heterogeneous interattribute relationship for ordinal-and-nominal-attribute data clustering[J]. IEEE Transactions on Cybernetics, 2022, 52(2): 758-771.

[66] BANABILAH S, ALOQAILY M, ALSAYED E, et al. Federated learning review: Fundamentals, enabling technologies, and future applications[J]. Information Processing &

Management, 2022, 59(6): 103061.

[67] ZHANG C, XIE Y, BAI H, et al. A survey on federated learning[J]. Knowledge-Based Systems, 2021, 216: 106775.

[68] ZHOU W, LI P, HAN Z, et al. Privacy-preserving federated learning via disentanglement [C]//2023 International Conference on Information and Knowledge Management. 2023: 3606-3615.

[69] MA J, ZHANG Q, LOU J, et al. Privacy-preserving tensor factorization for collaborative health data analysis[C]//2019 International Conference on Information and Knowledge Management. 2019: 1291-1300.

[70] NELUS A, GLITZA R, MARTIN R. Unsupervised clustered federated learning in complex multi-source acoustic environments[C]//2021 European Signal Processing Conference. 2021: 1115-1119.

[71] CHUNG J, LEE K, RAMCHANDRAN K. Federated unsupervised clustering with generative models[C]//2022 International Workshop on Trustable, Verifiable and Auditable Federated Learning. 2022: 1-9.

[72] FISHER R A. Iris[Z]. 1988.

[73] KUMAR H H, KARTHIK V, NAIR M K. Federated k-means clustering: A novel edge ai based approach for privacy preservation[C]//2020 International Conference on Cloud Computing in Emerging Markets. 2020: 52-56.

[74] SCHUBERT E, SANDER J, ESTER M, et al. Dbscan revisited, revisited: Why and how you should (still) use dbscan[J]. ACM Transactions on Database Systems, 2017, 42(3): 1-21.

[75] CHEUNG Y M. On rival penalization controlled competitive learning for clustering with automatic cluster number selection[J]. IEEE Transactions on Knowledge and Data Engineering, 2005, 17(11): 1583-1588.

[76] JIA H, CHEUNG Y M, LIU J. Cooperative and penalized competitive learning with application to kernel-based clustering[J]. Pattern Recognition, 2014, 47(9): 3060-3069.

[77] AHALT S C, KRISHNAMURTHY A K, CHEN P, et al. Competitive learning algorithms

for vector quantization[J]. Neural Networks, 1990, 3(3): 277-290.

[78] ZOU R, ZHANG Y, ZHANG Y, et al. Federated clustering with unknown number of clusters[C]//2024 International Conference on Data-driven Optimization of Complex Systems. 2024: 671-677.

[79] HU L, JIANG M, LIU X, et al. Significance-based decision tree for interpretable categorical data clustering[J]. Information Sciences, 2025, 690: 121588.

[80] ZHOU X, WANG X. Memory and communication efficient federated kernel k-means[J]. IEEE Transactions on Neural Networks and Learning Systems, 2022: 7114-7125.

[81] HUANG L, LI Z, SUN J, et al. Coresets for vertical federated learning: Regularized linear regression and k-means clustering[C]//2022 Advances in Neural Information Processing Systems. 2022: 29566-29581.

[82] CHEUNG Y M, JIA H. Categorical-and-numerical-attribute data clustering based on a unified similarity metric without knowing cluster number[J]. Pattern Recognition, 2013, 46(8): 2228-2238.

[83] ACAR A, AKSU H, ULUAGAC A S, et al. A survey on homomorphic encryption schemes: Theory and implementation[J]. ACM Computing Surveys, 2018, 51(4): 1-35.

[84] LI Y, WANG S, CHI C Y, et al. Differentially private federated clustering over non-iid data[J]. IEEE Internet of Things Journal, 2023: 6705-6721.

[85] ROS F, RIAD R, GUILLAUME S. Pdbi: A partitioning davies-bouldin index for clustering evaluation[J]. Neurocomputing, 2023, 528: 178-199.

[86] BAHMANI B, MOSELEY B, VATTANI A, et al. Scalable k-means++[J]. Proceedings of the VLDB Endowment, 2012, 5(7).

[87] CHENG Y. Mean shift, mode seeking, and clustering[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1995, 17(8): 790-799.

[88] ASUNCION A, NEWMAN D. Uci machine learning repository[Z]. 2007.

[89] CALINSKI T, HARABASZ J. A dendrite method for cluster analysis[J]. Communications in Statistics-Theory and Methods, 1974, 3(1): 1-27.

[90] DEMSAR J. Statistical comparisons of classifiers over multiple data sets[J]. The Journal

of Machine Learning Research, 2006, 7: 1-30.

[91] GHOSH A, CHUNG J, YIN D, et al. An efficient framework for clustered federated learning[C]//2020 Advances in Neural Information Processing Systems. 2020: 19586-19597.

[92] WEI K, LI J, DING M, et al. Federated learning with differential privacy: Algorithms and performance analysis[J]. IEEE Transactions on Information Forensics and Security, 2020, 15: 3454-3469.

附录 A　第三方论文评价

报告编号：202521C0725459

科 技 查 新 报 告

项目名称：多源异质大数据联合分析

委 托 人：学校

委托日期：二〇二五年三月十八日

查新机构：中国科学院上海科技查新咨询中心
（科技查新专用章）

查新完成日期：二〇二五年三月二十八日

中 华 人 民 共 和 国 科 学 技 术 部

二〇一六年制

查新项目名称	中文：多源异质大数据联合分析				
	英文：Joint Clustering Analysis Methods for Multi-source Heterogeneous Attribute Data				
查新机构	名　　称	中国科学院上海科技查新咨询中心			
	通信地址	上海市岳阳路319号31号楼C座底楼		邮编	200031
	负 责 人	江洪波	联系人	刘　剑	
	电　　话	021-54922920	传　真	021-54922934	
	电子信箱	chaxin@sibs.ac.cn	网　址	http://www.chaxin.ac.cn	

一、查新目的

成果鉴定查新 国内外查新

二、查新项目的科学技术要点

现实应用中的多源异质大数据分析存在分布完整性低、对齐程度弱、检测准度低这三个痛点问题，是实现高效高精度的多源异质大数据联邦聚类分析需解决的系统性问题。该项目设计了一种分布式异质特征数据聚类分析算法框架，在该框架中计算与分析分别在中央服务节点（疾控中心）和本地节点（如各级医疗单位和疾控数据节点）进行。中央服务节点主要实现与本地聚类信息的持续交流和多本地节点信息的整合聚类分析与增量式更新。本地节点采用异质数据度量新方法，将异质指标特征转为同质信息，为后续实施精准聚类分析奠定基础。基于此，对数据进行细粒度划分，获得本地的多粒度簇分布结构，并通过超复空间深度聚类，搭建异质特征之间的信息通路，促进不同指标的信息融合，获得精准的本地簇分布。由此同时解决上述三大关键科学问题，提升数据整合能力、聚类精度和鲁棒性，极大地推动聚类在复杂多源场景中的落地应用。

三、查新点与查新要求

查新点：

1、异步信息自组织隐私传输机制，提出了一种新的无监督联邦学习方法，在无需依赖“真实”簇数量的前提下，可以从异步通信的客户端学习分布知识，内部的平衡机制也可以自适应地平衡不同客户端对服务器的贡献，在非独立同分布这种真实的复杂数据分布情况下的聚类表现突出。

2、微簇自适应划分与层次融合自收敛机制，针对非平衡异常分布簇的分析难题，提出了一种新的多粒度竞争惩罚学习机制，用于自收敛地探索数据的嵌套多粒度簇，并且设计了多粒度簇聚合编码表征策略以融合多粒度聚类结果，实现对簇规模非平衡数据的多分布层次信息表征，进而实现高效精准的聚类。

3、多层次耦合四元数异质图编码策略，针对异质特征数据的复杂结构，提出了一个新颖的四元数表征学习框架，结合了异质特征数据的先验知识对数据进行预编码，并且将四元数超复空间旋转运算应用于无监督表征学习，实现异质特征数据预编码的高效解耦，从而提升表征学习模型拟合能力和泛化能力，进而实现了精准的端到端异质特征数据聚类。

查新要求：

查国内外有无相同或类似文献与研究报道，并作对比分析及新颖性判断。

四、文献检索范围及检索策略

Delphion (http：//www.delphion.com)	earliest — 2025 年 3 月
Current Contents Connect	1998 — 2025 年 3 月
Derwent Innovations Index	1963 — 2025 年 3 月
Web of Science	1899 — 2025 年 3 月
Aerospace & High Technology Database	1962 — 2025 年 3 月
ANTE： Abstracts in New Technologies and Engineering	1981 — 2025 年 3 月
Conference Papers Index	1982 — 2025 年 3 月
NTIS： National Technical Information Service	1964 — 2025 年 3 月
NetLibrary（http：//www.netlibrary.com）	2001 — 2025 年 3 月
EiCompendex	1969 — 2025 年 3 月
IEEE/IEE Electronic Library（IEL）	
Computer and Information Systems Abstracts	1981 — 2025 年 3 月
Electronics and Communications Abstracts	1981 — 2025 年 3 月
中国学术会议论文数据库	1986 — 2025 年 3 月
中国学位论文数据库	1986 — 2025 年 3 月
中国科技成果数据库（CSTAD）	1980 — 2025 年 3 月
中国企业、公司及产品数据库	1988 — 2025 年 3 月
中国专利文献数据库	1985 — 2025 年 3 月
中国知识产权网 (http：//www.cnipr.com)	1985 — 2025 年 3 月
中文科技期刊数据库	1989 — 2025 年 3 月
中国科技经济新闻数据库	1992 — 2025 年 3 月
清华同方科技期刊全文数据库	1994 — 2025 年 3 月
中国科技成果网(http：//www.nast.org.cn)	
中国标准全文数据库	
中国科技在线(www.chinatecn.com.cn)	

检索策略：

1. 多源异质 and 大数据分析 and 隐私传输 and (异步信息 or 无监督联邦学习 or 簇数量 or 异步通信 or 客户端学习 or 平衡机制 or 服务器 or 数据分布 or 聚类表现)
2. 多源异质 and 大数据分析 and 微簇自适应划分 and (层次融合自收敛机制 or 异常分布簇 or 多粒度竞争惩罚 or 自收敛 or 嵌套多粒度簇 or 聚合编码 or 多分布层次信息)
3. 多源异质 and 大数据分析 and 异质图编码 and (多层次耦合 or 四元数 or 特征数据 or 学习框架 or 异质特征数据 or 预编码 or 超复空间旋转 or 无监督表征学习 or 拟合能力 or 泛化能力 or 特征数据聚类)
4. Multi source heterogeneity and big data analysis and privacy transmission and (asynchronous information or unsupervised federated learning or cluster quantity or asynchronous communication or client learning or balance mechanism or server or data distribution or clustering performance)
5. Multi source heterogeneity and big data analysis and micro cluster adaptive partitioning and (hierarchical fusion self convergence mechanism or abnormal distribution clusters or multi granularity competition penalty or self convergence or nested multi granularity clusters or aggregation encoding or multi distribution hierarchical information)
6. Multi source heterogeneity and big data analysis and heterogeneous graph encoding and (multi-level coupling or quaternion or feature data or learning framework or heterogeneous feature data or precoding or hyper complex space rotation or unsupervised representation learning or fitting ability or generalization ability or feature data clustering)

五、检索结果

按照前述的主题词或关键词进行多种逻辑组配，在前面所列的国内外相关数据库和时段内，采用计算机检索和人工检索结合的方法进行检索，共查得一般相关文献 16 篇，现摘录如下：

国内文献：

1. 标题：一种基于大数据的计算机网络安全分析方法及系统

发明人：陈威，崔宇飞.

申请人：青岛誉名科技有限公司.

公开号：CN118555117A

公开日：2024-08-27

摘要：公开了一种基于大数据的计算机网络安全分析方法，涉及网络安全领域，包括：采集用户的行为数据和交互数据；对采集的用户数据集进行预处理；采用联邦学习架构，以预处理后的用户数据集为训练样本，在用户端本地训练机器学习模型，用于用户状态分析；将训练后的机器学习模型的参数上传至云服务器，在云服务进行全局聚合，得到全局机器学习模型；根据全局机器学习模型和交互数据，采用基于异常检测的无监督学习方法，对用户的计算机网络行为进行安全分析。针对现有技术中存在的网络安全效率低，通过用户端本地 BiLSTM 神经网络模型对用户状态进行实时分析，同时，利用联邦学习架构，在云端进行全局聚合等，提高了分析的效率。

2. 标题：无监督多源域自适应场景下的个性化联邦学习

著者：史彩娟，孔凡跃，郑远帆，赵琳，张昆.

机构：华北理工大学人工智能学院

出处：第十七届全国信号和智能信息处理与应用学术会议

摘要：无监督多源域自适应场景下，客户端之间的数据异构性加剧导致联邦学习过程中模型聚合困难和客户端模型性能倒退；另外，联邦学习全局模型泛化性能差，不能为客户端用户提供个性化解决方案。因此，提出了一种代理模型辅助的个性化联邦学习算法(PFLADM)，为每个客户端分别构建一个个性化模型来缓解数据异构的影响。首先，为客户端副模型设计恢复模块(RM)来缓解联邦学习客户端模型倒退问题；然后，为客户端副模型设计交换增强模块(EEM)，进一步提升模型的泛化性能，满足客户端的个性化任务需求。所提算法 PFLADM 在无监督多源域自适应场景下的两个数据集上进行了实验，与基准算法相比，在 SVHN 源域上 PFLADM 准确率提升了 7.8 个百分点，在 Quickdraw 源域上 PFLADM 准确率提升了 19.4 个百分点，源域的模型性能均得到了不同程度的提升，表明了所提算法 PFLADM 的有效性。

3. 标题：机器学习模型层的无监督联邦学习

发明人：弗朗索瓦丝·博费，沈启财，约翰·沙尔克威克.

申请人：谷歌有限责任公司.

公开号：CN116134453A

公开日：2023-05-16

摘要：该文公开的实施方式针对全局机器学习(“ML”)模型层的无监督联邦训练，该 ML 模型层在联邦训练之后可以与附加层组合，从而产生组合的 ML 模型。处理器可以：检测捕获客户端设备用户的口述话语的音频数据；使用本地 ML 模型处理音频数据以生成预测输出；使用在客户端设备本地的无监督学习基于预测输出来生成梯度；将梯度传输到远程系统；基于梯度来更新全局 ML 模型层的权重；在更新权重之后，在远程系统上远程使用监督学习训练组合的 ML 模型，所述组合的 ML 模型包括更新的全局 ML 模型层和附加层；将组合的 ML 模型传输到客户端设备；并使用组合的 ML 模型在客户端设备上进行预测。

4. 标题：面向大数据的聚类方法及其应用研究

著者：汪宜东.

出处：厦门大学 2015 硕士论文

摘要：近些年来，随着计算机科学与技术的快速发展，在很多行业中产生了越来越多的海量数据信息。聚类作为数据挖掘的一个非常受关注的分支学科，在这种情况下得到了长足的发展，一系列经典的聚类算法被研究者提出，但目前能应用于大数据聚类的算法不多，Apache Mahout 推出的聚类算法只有 5 种，其中有 4 种基于 Kmeans 算法开发的，Spark 官方推出的聚类算法目前只有 Kmeans。一些效果较好的聚类算法，它们的时间复杂度比较高，开发出适应大数据聚类的难度较大。传统的 Kmeans 可以用于大数据聚类，但其迭代过程涉及到多次的 HDFS 文件系统的读写操作也非常费时。该文通过引入聚类特征树，获得微簇中心点集，利用 Maximin 算法选取初始聚类中心点集，提出了基于层次和划分的 BM2Kmeans 算法，同时利用微簇中心进行微簇融合，提出了基于层次和密度的 BMCMCluster 算法。前一种算法能够实现快速搜索到较好且稳定的初始聚类中心点集，从而实现高效的大数据聚类，但需要指定聚类类别数;后一种算法也能够实现快速、高效的大数据聚类，且聚类类别数不需指定，算法会通过微簇融合的方式形成大的聚簇，能够发现任意形状聚簇。该文研究以油气勘探领域的数据为实验数据，对基于 Hadoop 平台实现的两种针对该文提出的聚类算法通过实验验证，并对实验结果进行分析，通过可视化的方式将这两种聚类算法的聚类结果表达出来。通过实验，可以看出基于层次和划分的集成聚类算法 BM2Kmeans 的聚类效果要优于传统的 Kmeans 大数据聚类算法，并对基于层次和密度的集成聚类算法 BMCMCluster 的聚类结果进行可视化展示。

5. 标题：基于分布式数据流的大数据分类模型和算法

著者：毛国君，胡殿军，谢松燕.

机构：中央财经大学

出处：计算机学报，2017，40(1)

摘要：大数据是需求驱动的概念。随着数据库系统的普及和因特网服务的扩张，企业或者个人可用的数据正在膨胀，已有的技术很难满足大数据时代的数据分析需求，因此需要探索新的理

论和方法来支撑大数据的应用。虽然大数据的4V属性已经被广泛讨论，但是它们大多描述的仍然是大数据的表象，所以很难从中抽象出统一的数据格式，因而进一步寻找可用于数据格式化的技术特征是必要的。面向于以分布式和流动性为主要技术特征的大数据应用需求，文中以分布式数据流为数据表达载体，在此基础上设计对应的大数据分类模型和挖掘算子。同时针对大数据的分类挖掘需要解决的关键问题来构建关键步骤对应的算法。理论上证明了文中给出的微簇合并技术和样本数据重构方法的合理性。文中提出的基于分布式数据流的大数据的分类模型及算法不仅能大幅度地减少网络节点间的通讯代价，而且可以获得平均10%左右的全局挖掘精度的提升(对比已有的典型算法DS-means);虽然时间花费略高于DS-means，但是两者在不同的数据容量测试下相差很小、且时间攀升趋势相当。

6. 标题：基于微簇的在线网络异常检测方法

著者：肖三，杨雅辉，沈晴霓.

机构：北京大学

出处：计算机工程与应用，2013，49(6)

摘要：针对大流量骨干网的在线网络异常检测是目前网络安全研究的热点之一，提出一种网络异常检测方法，有效在线处理大数据流，利用密度聚类算法把大数据流转换成微簇，通过微簇提高处理效率，定时调用孤立点检测算法发现攻击行为。方法具有不需线下训练、能发现任意行为模式、支持大数据流、可以平衡检测精度与系统资源要求、处理效率高等优点。实验表明，原型系统在20s完成2000年LLS DDOS 1.0数据集分析，检测率为82%，误报率为6%，效果与K-means相当。

7. 标题：基于异质数据融合的目标辨识技术研究

著者：高全伟.

出处：西安电子科技大学2020硕士论文

摘要：异质时空数据融合旨在综合不同类型传感器上获取的具有互补性的数据，是提升数据应用能力的重要手段，在现代安防监控中发挥关键作用。由于其自动的特征学习能力与在大数据下的优良性能，近年来深度学习在安防监控中的应用日益增多，但现有研究大多集中于单类型数据。在实际安防系统中，常获取多种类型传感器在不同时间点的时序数据，需要深度学习方法能够对异质时空数据进行自动综合与处理。围绕该问题，该文面向安防监控中的毫米波雷达与光学传感器，探索基于深度学习的多源异质时空数据的关联、配准、融合，以及目标检测、识别与跟踪方法，以提高雷摄系统(毫米波雷达-光学摄像头系统)的目标辨识精度。主要研究内容及成果如下：1.针对常用数据关联方法对于应用场景鲁棒性不足的缺陷，提出了一种基于在线深度学习的数据关联方法。首先，设计深度相机标定网络，利用传统相机标定方法初始化网络参数；然后，设计一种基于最近邻与交并比相结合的数据关联方法，对雷达目标与光学目标进行数据关联；最后，利用雷摄系统中的数据流在线训练相机标定网络。由于该网络利用了在线深度学习，因此具

有实时性、高效性的特点，可以有效应对用于不同场景时鲁棒性不足的问题。在自行构造的数据集上进行了相机标定与数据关联实验，实验结果表明：方法能快速对异质数据的相关性做出判断，相比于传统相机标定方法定位误差减少 13.04%，数据关联准确率提升 6.01%。2.针对多源目标检测中异质数据未能充分利用的问题，提出了一种基于注意力机制的雷摄目标检测模型 RC-Det(RadarCamerasystem Detection)。首先，对光学图像初步检测，并利用相机标定网络确定雷达目标在图像中的范围；接着，构建雷达与光学图像映射模块，并将图像初步检测结果与相机标定结果送入映射模块，分别产生光学、雷达显著图；然后，设计基于卷积自编码的双定位融合模块，并将上述显著图送入其中，以确定目标在图像中的范围；最后，利用图像目标检测模型对该范围图像进行精确检测。在自行构建的数据集中，将该模型分别与单源目标检测网络以及传统数据融合算法进行多组对比实验，结果表明，RC-Det 模型的准确度相比现有单源目标检测网络至少提升 9.08%，相比传统数据融合算法提升 6.91%，充分验证了网络的有效性。3.为了充分利用异质时空数据在时序上的关联信息，提出了基于长短时记忆网络(LSTM，Long Short-Term Memory)的雷摄目标跟踪模型 RC-Track(Radar Camera systemTracking)。首先，基于 LSTM 构建时序目标的预测网络，根据目标在图像中的变化趋势，预测其下一时刻位置与大小；其次，利用相机标定网络与雷达映射模块生成雷达显著图；接着，将上一时刻预测信息送入光学映射模块，产生光学显著图；最后，将上述显著图送入到双融合定位模块，根据其结果对目标所在范围进行精确检测，并将结果返回预测模块。在自行构建的数据集中，多组对比实验结果表明：RC-Track 模型的准确率相比光学目标跟踪模型至少提升 6.62%，相比于传统时序异质融合的目标跟踪方法提升 4.7%。

8. 标题：基于多维度特征融合的云工作流任务执行时间预测方法

著者：李慧芳，黄姜杭，徐光浩，夏元清.

机构：北京理工大学

出处：自动化学报，2023，49(1)

摘要：任务执行时间估计是云数据中心环境下工作流调度的前提。针对现有工作流任务执行时间预测方法缺乏类别型和数值型数据特征的有效提取问题，提出了基于多维度特征融合的预测方法。首先，通过构建具有注意力机制的堆叠残差循环网络，将类别型数据从高维稀疏的特征空间映射到低维稠密的特征空间，以增强类别型数据的解析能力，有效提取类别型特征;其次，采用极限梯度提升算法对数值型数据进行离散化编码，通过对稠密空间的输入向量进行稀疏化处理，提高了数值型特征的非线性表达能力;在此基础上，设计多维异质特征融合策略，将所提取的类别型、数值型特征与样本的原始输入特征进行融合，建立基于多维融合特征的预测模型，实现了云工作流任务执行时间的精准预测;最后，在真实云数据中心集群数据集上进行了仿真实验。实验结果表明，相对于已有的基准算法，该方法具有较高的预测精度，可用于大数据驱动的云工作流任务执行时间预测。

9. 标题：基于深度残差自编码器的无监督聚类算法

著者：张浩，陆彦辉.

机构：郑州大学信息工程学院

出处：计算机仿真，2023，40(1)

摘要：随着社会数字化程度的加深，数据的类型和维度不断增长，数据逐渐呈现出高维特性。在高维数据下，传统无监督聚类算法聚类效率低下，维度灾难导致其性能不佳。随着深度学习的发展，自编码器技术在降维任务上取得了长足的进步。提出了基于深度残差自编码器的无监督聚类方法——ResDAE-KMeans++。上述方法在无监督训练的深度残差自编码器基础上，应用KMeans++在低维特征空间中自主聚类。相较其他无监督聚类算法，应用非线性的残差自编码器编码后的特征空间使得聚类速度显著提升的同时，准确率也得到了进一步提高。在Iris、Wine、MNIST数据集上与其它主流无监督算法进行对比，实验结果表明，ResDAE-KMeans++算法在对比其它聚类算法存在有明显优势。

国外文献：

10. TI: Asynchronous Transmission Schemes for Digital Information

AU：Mine H , Hasegawa T , Koga Y .

AN：Department of Applied Mathematics and Physics, Kyoto University, Kyoto, Japan

SO：IEEE transactions on communication technology, 2003, 18(5):562-568.

该文介绍了数字信息的异步传输方案，提出了一些用于数字信息异步传输的二进制信号到三进制信号的编码方案，介绍并讨论了微分三元归零信号在异步传输中的应用，其中每个信号都与特定的分离信号相关联，其次介绍并讨论了连续信号相似的编码方案。在这些方案中，信号可能具有较大的码间干扰，从理论上计算了斐波那契编码信号的功率谱，在所有情况下任何一个错误都可能导致一系列错误，为了防止由错误引起的这些错误，需要实现稳定的块或帧同步。

11. TI: Long code state information transmission method in asynchronous mobile communication system

发明人：P Jehon，パク ジェホン，I Jonwon，イ ジョンウォン，I Yuro，イ ユロ

公开号：JP 特開 2001-251666

该专利公开了一种异步移动通信系统中的长码状态信息传输方法，通过将长码状态信息从异步无线网络传输到异步终端，在从异步无线网切换到同步无线网的情况下消除呼叫中断。

12. TI: A Nested Partitioning Algorithm for Adaptive Meshes on Heterogeneous Clusters

AU：Sundar H , Ghattas O .

AN：University of Utah

SO：ICS '15: Proceedings of the 29th ACM on International Conference on Supercomputing, 2015.

该文介绍了异构簇上自适应网格的嵌套划分算法，同构系统的分区只需要平衡节点级工作负载与单节点性能，而支持加速器的系统的分区需要一个嵌套的分区方案，以确保最佳的节点内和

节点间负载平衡，在异构的 Intel R Xeon Phi 加速超级计算机（Stampede）上证明了所提出的分区方案的有效性，使用 7 阶离散化实现了高达 5.78 倍的加速，使用 15 阶离散化相对于基线纯 MPI 实现实现了 6.88 倍的速度，展示了强和弱的缩放结果以及单个节点的性能，以说明加速器支持的科学计算的优点和局限性。

13. TI: Clustering web documents using hierarchical representation with multi-granularity

AU：Huang F , Zhang S , He M ,et al.

AN：Fujian Normal University

SO：World Wide Web, 2014, 17(1):105-126.

该文介绍了使用具有多粒度的层次表示对 web 文档进行聚类，基于粒度计算和文章结构理论，提出了一种由五层数据表示和两阶段聚类过程组成的多粒度层次表示模型（HRMM），在 HRMM 中引入了基于本体的策略和基于容差粗糙集的策略，通过使用粒度计算，HRMM 可以更高效、更有效地捕获文档中隐藏的结构知识，从而生成更高质量的 web 文档集群，HRMM、具有容差粗糙集策略的 HRMM 和具有本体论的 HRMM 在 F-Score 方面都明显优于 VSM 和代表性的非 VSM 算法 WFP。

14. TI: Position encoding for heterogeneous graph neural networks

AU：Zeng X

AN：Guangdong Provincial Key Lab. of Intellectual Property & Big Data (China)

SO：Proceedings of SPIE, 2022, 12258(000):8.

该文介绍了异构图神经网络的位置编码，可以通过图上的拓扑信息来生成有用的位置特征，并提出了异构图神经网络（HGNN）的通用框架，称为位置编码（PE）。首先，PE 利用现有的节点嵌入方法来获取图上的隐式语义，并生成低维节点嵌入，对于每个与任务相关的目标节点，PE 生成相应的采样子图，其中使用节点嵌入来计算相对位置，并将位置编码为可以直接使用或作为附加特征的位置特征。具有位置特征的子图集可以很容易地与所需的图神经网络（GNN）或 HGNN 相结合，以学习目标节点的表示。

15. TI: Self-supervised Heterogeneous Graph Neural Network Based onDeep andBroad Neighborhood Encoding

AU：Li C, Song Q, Fu J, et al.

AN：Shandong University of Science and Technology

SO：Blockchain and Web3 Technology Innovation and Application Exchange Conference.Springer, Singapore, 2025.

该文介绍了基于深度和广度邻域编码的自监督异构图神经网络，HGNN-DB 旨在有效地捕捉深层和广泛邻域的特征，引入了一种深度邻域编码器，该编码器使用距离加权策略来捕获目标节点的深度特征，利用单层图卷积网络作为宽邻域编码器来聚合目标节点的宽特征，还采用了一种协作对比算法来学习两种邻域信息视图之间的互补性和潜在不变性。

16. TI: Adaptive micro partition and hierarchical merging for accurate mixed data clustering

AU：Zhang Y, Zou R, Zhang Y, et al.

AN：Guangdong University of Technology

SO：Complex & Intelligent Systems, 2025, Vol 11, Issue 1, pp.1-14.

异构属性数据（也称为混合数据）以具有数值和分类值的属性为特征，在各种场景中经常出现。由于标注成本高，聚类已成为分析未标记混合数据的有利技术。为了解决复杂的现实世界聚类任务，本文提出了一种基于邻域粗糙集理论的自适应微划分和分层合并（AMPHM）聚类方法和一种新的分层合并机制。具体来说，我们提出了一种在数值和分类属性上统一的距离度量，以利用邻域粗糙集将数据对象划分为细粒度的紧凑集群。然后，我们逐渐合并当前最相似的集群，以避免将不同的对象合并到相似的集群中。结果表明，所提出的方法突破了由预设的搜索聚类数量 k 和聚类分布偏差带来的聚类性能瓶颈，因此能够对包含数值和分类属性的各种组合的数据集进行聚类。将所提出的 AMPHM 与各种数据集上最先进的对应物进行广泛的实验评估，证明了其优越性。

六、查新结论

依据与查新委托人签定的“科技查新合同”的有关要求，针对“多源异质大数据联合分析”的课题，我们利用国内外数据库进行了查新检索，共检索到文献16篇。

经阅读、分析对比得到以下结论：

国内外相关文献中，主要内容有：文献1公开了一种基于大数据的计算机网络安全分析方法，将训练后的机器学习模型的参数上传至云服务器，在云服务进行全局聚合，得到全局机器学习模型，根据全局机器学习模型和交互数据，采用基于异常检测的无监督学习方法，对用户的计算机网络行为进行安全分析；文献2介绍了无监督多源域自适应场景下的个性化联邦学习，无监督多源域自适应场景下，客户端之间的数据异构性加剧导致联邦学习过程中模型聚合困难和客户端模型性能倒退；文献3公开了机器学习模型层的无监督联邦学习，使用本地ML模型处理音频数据以生成预测输出；使用在客户端设备本地的无监督学习基于预测输出来生成梯度；将梯度传输到远程系统；基于梯度来更新全局ML模型层的权重；文献4介绍了面向大数据的聚类方法及其应用研究，通过引入聚类特征树，获得微簇中心点集，利用Maximin算法选取初始聚类中心点集，提出了基于层次和划分的BM2Kmeans算法，同时利用微簇中心进行微簇融合，提出了基于层次和密度的BMCMCluster算法；文献5介绍了基于分布式数据流的大数据分类模型和算法，微簇合并技术和样本数据重构方法的合理性，基于分布式数据流的大数据的分类模型及算法不仅能大幅度地减少网络节点间的通讯代价，而且可以获得平均10%左右的全局挖掘精度的提升(对比已有的典型算法DS-means)；文献6介绍了基于微簇的在线网络异常检测方法，利用密度聚类算法把大数据流转换成微簇，通过微簇提高处理效率，定时调用孤立点检测算法发现攻击行为；文献7介绍了基于异质数据融合的目标辨识技术研究，异质时空数据融合旨在综合不同类型传感器上获取的具有互补性的数据，是提升数据应用能力的重要手段，在现代安防监控中发挥关键作用，设计一种基于最近邻与交并比相结合的数据关联方法，对雷达目标与光学目标进行数据关联，利用雷摄系统中的数据流在线训练相机标定网络；文献8介绍了基于多维度特征融合的云工作流任务执行时间预测方法，通过构建具有注意力机制的堆叠残差循环网络，将类别型数据从高维稀疏的特征空间映射到低维稠密的特征空间，以增强类别型数据的解析能力，有效提取类别型特征；文献9介绍了基于深度残差自编码器的无监督聚类算法，提出了基于深度残差自编码器的无监督聚类方法——ResDAE-KMeans++，应用KMeans++在低维特征空间中自主聚类；文献10介绍了数字信息的异步传输方案，提出了一些用于数字信息异步传输的二进制信号到三进制信号的编码方案，介绍并讨论了微分三元归零信号在异步传输中的应用，其中每个信号都与特定的分离信号相关联，其次介绍并讨论了连续信号相似的编码方案；文献11公开了一种异步移动通信系统中的长码状态信息传输方法，通过将长码状态信息从异步无线网络传输到异步

终端，在从异步无线网切换到同步无线网的情况下消除呼叫中断；文献 12 介绍了异构簇上自适应网格的嵌套划分算法，同构系统的分区只需要平衡节点级工作负载与单节点性能，而支持加速器的系统的分区需要一个嵌套的分区方案，以确保最佳的节点内和节点间负载平衡，在异构的 Intel R Xeon Phi 加速超级计算机上证明了所提出的分区方案的有效性，使用 7 阶离散化实现了高达 5.78 倍的加速，使用 15 阶离散化相对于基线纯 MPI 实现实现了 6.88 倍的速度；文献 13 介绍了使用具有多粒度的层次表示对 web 文档进行聚类，基于粒度计算和文章结构理论，提出了一种由五层数据表示和两阶段聚类过程组成的多粒度层次表示模型（HRMM），在 HRMM 中引入了基于本体的策略和基于容差粗糙集的策略，通过使用粒度计算，HRMM 可以更高效、更有效地捕获文档中隐藏的结构知识；文献 14 介绍了异构图神经网络的位置编码，可以通过图上的拓扑信息来生成有用的位置特征，并提出了异构图神经网络（HGNN）的通用框架，称为位置编码（PE）；文献 15 介绍了基于深度和广度邻域编码的自监督异构图神经网络，HGNN-DB 旨在有效地捕捉深层和广泛邻域的特征，引入了一种深度邻域编码器，编码器使用距离加权策略来捕获目标节点的深度特征，利用单层图卷积网络作为宽邻域编码器来聚合目标节点的宽特征，还采用了一种协作对比算法来学习两种邻域信息视图之间的互补性和潜在不变性；文献 16 本委托单位提出了一种基于邻域粗糙集理论的自适应微划分和分层合并（AMPHM）聚类方法和一种新的分层合并机制。

文献分析对比表明：

国内外见有基于大数据的计算机网络安全分析方法，将训练后的机器学习模型的参数上传至云服务器，在云服务进行全局聚合，得到全局机器学习模型，根据全局机器学习模型和交互数据，采用基于异常检测的无监督学习方法，对用户的计算机网络行为进行安全分析的报道，但本委托项目查新点 1“异步信息自组织隐私传输机制，提出了一种新的无监督联邦学习方法，在无需依赖‘真实’簇数量的前提下，可以从异步通信的客户端学习分布知识，内部的平衡机制也可以自适应地平衡不同客户端对服务器的贡献，在非独立同分布这种真实的复杂数据分布情况下的聚类表现突出”，在以上国内外文献检索中，未见相同文献报道；见有面向大数据的聚类方法及其应用研究，通过引入聚类特征树，获得微簇中心点集，利用 Maximin 算法选取初始聚类中心点集，提出了基于层次和划分的 BM2Kmeans 算法，同时利用微簇中心进行微簇融合，提出了基于层次和密度的 BMCMCluster 算法的报道，但本委托项目查新点 2“微簇自适应划分与层次融合自收敛机制，针对非平衡异常分布簇的分析难题，提出了一种新的多粒度竞争惩罚学习机制，用于自收敛地探索数据的嵌套多粒度簇，并且设计了多粒度簇聚合编码表征策略以融合多粒度聚类结果，实现对簇规模非平衡数据的多分布层次信息表征，进而实现高效精准的聚类”，在以上国内外文献检索中，未见相同文献报道；见有基于异质数据融合的目标辨识技术研

究，异质时空数据融合旨在综合不同类型传感器上获取的具有互补性的数据，是提升数据应用能力的重要手段，在现代安防监控中发挥关键作用，设计一种基于最近邻与交并比相结合的数据关联方法的报道，但本委托项目查新点3“多层次耦合四元数异质图编码策略，针对异质特征数据的复杂结构，提出了一个新颖的四元数表征学习框架，结合了异质特征数据的先验知识对数据进行预编码，并且将四元数超复空间旋转运算应用于无监督表征学习，实现异质特征数据预编码的高效解耦，从而提升表征学习模型拟合能力和泛化能力，进而实现了精准的端到端异质特征数据聚类”，在以上国内外文献检索中，未见相同文献报道。

综上所述，关于本查新项目“多源异质大数据联合分析”的3个查新点：

1、异步信息自组织隐私传输机制，提出了一种新的无监督联邦学习方法，在无需依赖“真实”簇数量的前提下，可以从异步通信的客户端学习分布知识，内部的平衡机制也可以自适应地平衡不同客户端对服务器的贡献，在非独立同分布这种真实的复杂数据分布情况下的聚类表现突出。

2、微簇自适应划分与层次融合自收敛机制，针对非平衡异常分布簇的分析难题，提出了一种新的多粒度竞争惩罚学习机制，用于自收敛地探索数据的嵌套多粒度簇，并且设计了多粒度簇聚合编码表征策略以融合多粒度聚类结果，实现对簇规模非平衡数据的多分布层次信息表征，进而实现高效精准的聚类。

3、多层次耦合四元数异质图编码策略，针对异质特征数据的复杂结构，提出了一个新颖的四元数表征学习框架，结合了异质特征数据的先验知识对数据进行预编码，并且将四元数超复空间旋转运算应用于无监督表征学习，实现异质特征数据预编码的高效解耦，从而提升表征学习模型拟合能力和泛化能力，进而实现了精准的端到端异质特征数据聚类。

在以上国内外文献检索中，未见相同文献报道。因此，本查新项目具有新颖性。

查新员（签字）：王佳

查新员职称：馆员

审核员（签字）：

审核员职称：研究员
（国家级审核员）

（科技查新专用章）

2025 年 3 月 28 日

七、查新员、审核员声明

1、 本报告陈述的内容均以公开文献为依据。

2、 我们按照《科技查新技术规范》GB/T 32003-2015 进行查新和审核，并作出上述查新结论。

查新员（签字）：王俊迪　　审核员（签字）：安勇

2025 年 3 月 28 日　　2025 年 3 月 28 日

八、附件清单

相关文献（文摘或全文）16 篇。

九、备注

1、 本查新报告无查新员和审核员签名无效；

2、 本查新报告无查新机构的“科技查新专用章”无效；

3、 本查新报告涂改无效；

4、 其他需要备注的内容。

推 荐 信

本科生张云帆同学申报的“多源异质大数据联合分析”项目，针对大数据复杂分析场景中的多源异质数据共享与联合分析问题，深入开展了联邦聚类分析技术的研究。提出了统一的距离度量空间和表征策略，对异质数据进行信息融合增强。设计了多粒度层次划分和自组织竞争机制，在无监督且数据脱敏的信息匮乏情形下实现了高精度、高效、以及可持续的多源异质数据联邦聚类分析。对无监督联邦学习和异质数据分析的研究具有前沿性和前瞻性，项目所提出的方案为解决该领域的关键基础性技术难题提供了创新思路。部分研究成果已发表于 SCI 一区期刊和 CCF-A 类会议，充分证实了其科学性和创新性。因此，本人郑重推荐该项目参加第十九届“挑战杯”全国大学生课外学术科技作品竞赛。

推荐人：

D1.推荐者情况及对作品的说明

说明：1．由推荐者本人填写；

2．推荐者必须具有高级专业技术职称，并是与申报作品相同或相关领域的专家学者或专业技术人员（教研组集体推荐亦可）；

3．推荐者填写此部分，即视为同意推荐；

4．推荐者所在单位签章仅被视为对推荐者身份的确认。

<table>
<tr><td rowspan="4">推荐者情况</td><td>姓　名</td><td>蔡宏民</td><td>性别</td><td>男</td><td>年龄</td><td>45</td><td>职称</td><td>教授</td></tr>
<tr><td>工作单位</td><td colspan="7">华南理工大学未来技术学院</td></tr>
<tr><td>通讯地址</td><td colspan="3">广州市番禺区兴业大道东 777 号</td><td colspan="2">邮政编码</td><td colspan="2">511442</td></tr>
<tr><td>单位电话</td><td colspan="3">020-81181674</td><td colspan="2">住宅电话</td><td colspan="2"></td></tr>
<tr><td>推荐者所在单位签章</td><td colspan="8">同意推荐
（签章）　　年　月　日</td></tr>
<tr><td>请对申报者申报情况的真实性作出阐述</td><td colspan="8">学生团队有丰富的相关研究经历，在该领域知名期刊和会议发表了系列论文，成果在三甲医院进行了部署应用并取得了相应证明，上述情况经核查真实可靠。</td></tr>
<tr><td>请对作品的意义、技术水平、适用范围及推广前景作出您的评价</td><td colspan="8">该作品以真实复杂数据分析环境的应用需求为导向，以打破数据孤岛效应，突破当前的异构特征数据聚类分析精度为目标，研究并提出了系列方法。包括异构特征数据聚类算法、无监督联邦学习方法、深度表征模型等。成果关联紧密且成体系，共同构成了联邦异构数据分析方法体系。成果的理论性和科学性较强，研究的问题具有前瞻性，很好地拓展和丰富了数据科学和机器</td></tr>
</table>

	学习研究领域。已发表 CCF-A 类顶级会议论文以及 SCI 期刊论文多篇，充分证实了该成果的科学价值。此外，该作品在许多重要行业有良好应用前景，如医疗数据分析、工业异常检测、推荐系统等。有望撬动更多无标签数据资源，提升行业应用中的数据分析效能。
其它说明	该参赛作品成果形式丰富，包括论文、专利、应用成果等。且绝大多数论文均为该团队本科生以第一作者身份发表，并已编纂为论文集，成果系统性强。
学校组织协调机构确认盖章	年　月　日
校主管领导签字或学校确认盖章	（学校盖章）　年　月　日

该项目论文受领域内高水平学者正面引用与评价：

- 香港浸会大学人工智能讲席教授，教育部长江学者讲座教授张晓明【IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2025, DOI: 10.1109/ICASSP49660.2025.10889806】
- 电子科技大学杨波教授【Information Processing & Management, 2025, DOI: 10.1016/j.ipm.2025.104076】

附录 B　论文原文列表

Learning Self-Growth Maps for Fast and Accurate Imbalanced Streaming Data Clustering

[redacted] Zexi Tan [redacted]

***Abstract*—Streaming data clustering is a popular research topic in data mining and machine learning. Since streaming data is usually analyzed in data chunks, it is more susceptible to encounter the dynamic cluster imbalance issue. That is, the imbalance ratio of clusters changes over time, which can easily lead to fluctuations in either the accuracy or the efficiency of streaming data clustering. Therefore, we propose an accurate and efficient streaming data clustering approach to adapt the drifting and imbalanced cluster distributions. We first design a Self-Growth Map (SGM) that can automatically arrange neurons on demand according to local distribution, and thus achieve fast and incremental adaptation to the streaming distributions. Since SGM allocates an excess number of density-sensitive neurons to describe the global distribution, it can avoid missing small clusters among imbalanced distributions. We also propose a fast hierarchical merging strategy to combine the neurons that break up the relatively large clusters. It exploits the maintained SGM to quickly retrieve the intra-cluster distribution pairs for merging, which circumvents the most laborious global searching. It turns out that the proposed SGM can incrementally adapt to the distributions of new chunks, and the Self-grOwth map-guided Hierarchical merging for Imbalanced data clustering (SOHI) approach can quickly explore a true number of imbalanced clusters. Extensive experiments demonstrate that SOHI can efficiently and accurately explore cluster distributions for streaming data.**

***Index Terms*—Cluster analysis, streaming data, imbalanced data, self-organizing map, drift adaptation, efficient algorithms.**

I. Introduction

STREAMING data, specifically the datasets with flowing-in or updates of its objects over time, is prevalent in various fields, such as market research, health big data analysis, and the internet of things [1]–[3]. Due to the lack of readily available labels for streaming data, clustering that gathers similar data objects into a certain number of groups becomes indispensable for data analysis. Cluster analysis of streaming data is often conducted on a chunk-by-chunk basis to ensure sufficient statistical information [4]. However, the non-uniform co-occurrence of data objects from different distributions at the same time, along with shifts in data distributions over time [5], will frequently lead to the emergence of imbalanced clusters, i.e., clusters with very different numbers of objects. Fig. 1 shows the causes of such a problem, which puts forward a new clustering challenge, i.e., how to accurately explore and efficiently adapt to the imbalanced cluster distributions of streaming data chunks.

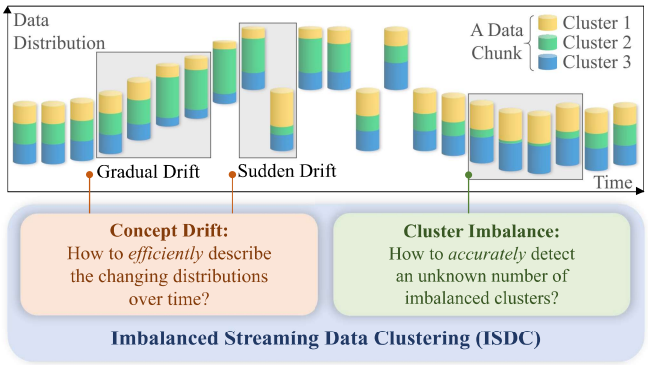


Fig. 1: Key problems in imbalanced streaming data clustering.

Imbalanced streaming data introduces more complex and challenging issues to clustering, which are manifested into two key aspects: 1) Uncertainty in the number of clusters, and 2) Laborious detection of imbalanced clusters. On the one hand, due to the very different and changing sizes of clusters, it is difficult to dynamically determine an appropriate number of clusters, leading to unsatisfactory clustering accuracy. On the other hand, existing imbalanced data clustering solutions [6] usually involve two phases: i) partition all n objects into m-scale microclusters ($m \gg k^*$, where k^* is the true number of clusters), and ii) perform m^2-scale merging of the microclusters to form final clusters. Such a time-consuming $O(m^2)$-complex process prevents the existing imbalanced data clustering approaches from efficient streaming data analysis, while the fast clustering techniques struggle in exploring imbalanced clusters due to the overlook of relatively small clusters. Subsequently, we analyze existing relevant clustering solutions from the perspective of Imbalanced Streaming Data Clustering (ISDC).

Partitional clustering algorithms demonstrate a certain level of capability in addressing ISDC. In this stream, the most classic method would be the k-means [7] algorithm. Thanks to its simplicity and efficiency, it can be applied to the analysis of streaming data with linear complexity. However, both k-means and most of its variants (e.g., [8], [9]) tend to produce balanced clusters [10]. To specifically address the imbalance issue, the Multi-Center (MC) clustering algorithm [11] has been proposed to partition the whole dataset into many small subclusters, and then successively merge the closer pairs to avoid neglecting relatively small clusters. As its performance is very sensitive to initialization, a self-adaptive clustering method called SMCL [6] has been proposed, which more robustly generates a proper number of seed points through competitive learning [12]–[14] to describe the data distribution for exploring imbalanced clusters. A novel rough set-based approach M3W [15] has also been proposed to generate a hierarchical data distribution structure that informatively facilitates the formation of small subclusters with clear boundaries. However, similar to MC and SMCL, it involves relatively laborious computation due to the m^2-scale search of many (m-scale) seed points, making it unsuitable for streaming data.

Density-based clustering methods determine object-cluster affiliation according to the local density of objects [16], and they demonstrate a certain degree of capability in handling imbalanced data. For instance, Density Peaks Clustering (DPC) [17] identifies data objects with relatively higher local density as cluster centers, and can thus explore smaller clusters with relatively prominent local density. To achieve more reasonable density quantification, an algorithm called FKNN-DPC [18] has been proposed, employing fuzzy weighted k-nearest neighbor as a density measure. With a more appropriate density measure, FKNN-DPC achieves better performance in detecting imbalanced clusters in comparison with the original DPC. Later, an approach called Local Density Peaks for Imbalanced data (LDPI) [19] has been proposed, adopting an adaptive subcluster construction scheme, which forms more subclusters than true clusters to enhance the detection of imbalanced clusters. However, LDPI has an object-wise quadratic time complexity, thus outside the consideration of efficient ISDC.

To specifically achieve efficient cluster analysis, fast clustering solutions have emerged. One of the most conventional methods is StreamKM++ [20], which integrates a software acceleration architecture and k-means++ [21] to dynamically update cluster centers to incrementally fit streaming data distribution. The efficiency of density-based clustering has also been improved by adapting DPC to streaming data, which demonstrates superior clustering performance compared to other streaming data clustering methods [16]. However, dynamic DPC is not robust to hyper-parameters when dealing with non-stationary streaming data. To further address this issue, AMD-DPC [22] was proposed, which adopts a graph-based data structure for local density updates. This algorithm saves computation cost while remaining accurate, showing potential for real-time big data processing. By inheriting the efficiency of partitional clustering and the unbiased cluster discovery capability of density-based clustering, a fast and accurate clustering algorithm called IGMTT [23] has been presented. However, the above-mentioned fast algorithms have yet to consider imbalanced clusters, and may thus yield unsatisfactory clustering results in the ISDC tasks.

To the best of our knowledge, most conventional clustering assumes that the true number of clusters k^* is given by the users based on prior knowledge of the data [4], [10]. However, for ISDC, where the distribution of the current new data chunks may continue to change, it is difficult to obtain prior knowledge about k^* in advance. Combining all the above analysis, it can be concluded that existing methods only consider one of the factors of imbalance or streaming, and most of them do not take into account the case of unknown k^*. Therefore, there is an urgent need to achieve fast and accurate ISDC without knowing k^*.

This paper, therefore, proposes a fast and accurate approach called **S**elf-gr**O**wth maps-guided **H**ierarchical merging for **I**mbalanced data clustering (SOHI, pronounced 'so high!') for ISDC. First, to realize an efficient distribution description for data chunks, we propose a Self-Growth Map (SGM) learning algorithm that can quickly and incrementally adapt the distribution via connected neurons generated in need. The maps use a 3-neuron triangle as the basic geometry unit for growth. Compared to the self-organizing map that adopts pre-sized grids, our SGM, which grows in a triangular manner, enables flexible distribution exploration and can incrementally adapt to the changing distributions. Based on the distribution described by the abundant map neurons, relatively small clusters can be effectively captured. Subsequently, further exploration of the imbalanced clusters is performed by hierarchically merging the adjacent data objects corresponding to the neurons. The topological structure of SGM is fully utilized to accelerate the laborious hierarchical merging by serving as a retrieval structure, allowing only closely connected neurons to merge. According to our analysis, such a map retrieval-based acceleration considerably improves the time complexity. In comparison with the state-of-the-art counterparts, SOHI demonstrates its superiority in ISDC as it sufficiently improves time complexity, while still being competitive in clustering accuracy. Comprehensive experiments have been conducted to illustrate its promising performance. The main contributions are summarized into four-fold:

1) A new paradigm called SOHI is proposed for ISDC. The distribution information provided by the learned SGM is thoroughly exploited to relieve the trade-off between efficiency and accuracy in ISDC.
2) We propose self-growing maps named SGM with triangles as the basic growth unit. Such a design avoids the missing of small clusters, and can flexibly adapt to streaming chunks incrementally.
3) To ensure an efficient imbalanced cluster detection, a fast hierarchical merging mechanism is designed to fully utilize the similarity of local distributions reflected by the SGM, thus considerably avoiding meaningless searches.
4) An imbalanced streaming data chunk generator is presented to simultaneously simulate the changing of cluster number and cluster size in real ISDC scenarios. It ensures a more convincing evaluation and can be a universal experimental tool in studying ISDC.

The rest of this paper is organized as follows: Section II reviews the related work. Section III presents the problem statement and preliminaries, and Section IV introduces the proposed SOHI with time complexity analysis. Section V showcases experimental results with in-depth observations. Finally, we conclude the whole work in Section VI.

II. Related Work

This section overviews the research topics relevant to this paper, including streaming data clustering, imbalanced data clustering, and distribution learning techniques.

A. Streaming Data Clustering

Data stream clustering partitions a series of data chunks into compact clusters, which places higher requirements on efficiency than static data clustering. Therefore, the streaming data clustering method strives to strike a balance between accuracy and efficiency [4]. Many of these algorithms are variations of traditional clustering methods like the partitional k-means [10] (e.g., [13], [20], [24], [25]), density-based DBSCAN (e.g.,[26]–[28]), and hierarchical clustering [29]. Based on the brief introduction in Section I, we know that they either introduce approximation-based acceleration from the computational level, such as StreamKM++ [20], or replace the original modules involving laborious computation with more efficient alternatives from the algorithmic level, e.g., IGMTT [23]. However, they inevitably introduce hyper-parameters that are difficult to tune and may cause unstable performance.

Since constructing and merging microclusters by measuring the Euclidean distances between high-dimensional data objects is challenging, OSRC [30] utilizes low-dimensional projection to effectively select an appropriate number of representative data objects, but may be sensitive to the selection of hyper-parameters. To circumvent this, the work proposed in [31] uses the Davies-Bouldin Index (DBI) to guide the optimization of clustering. However, its scalability is limited by the DBI computation based on static data. Accordingly, the work [32] develops incremental Xie-Beni (XB) and DBI indices to monitor the streaming clustering process of k-means type algorithms. To extend the above methods to process multi-view data, a multi-view support vector domain description model [33] has been proposed to capture cluster evolution and discover arbitrarily shaped clusters with limited computing resources. Nevertheless, streaming data clustering remains a challenging issue due to the unavoidable trade-off between efficiency and accuracy.

B. Imbalanced Data Clustering

Imbalanced data clustering, where the scale varies for different clusters, has attracted much more attention in real data mining applications [34]. In addition to the density-based clustering methods introduced in Section I, which naturally has a certain ability to handle imbalanced clusters, using many prototypes to capture the micro distributions is also considered one of the most effective ways to specifically avoid missing small clusters. Such a principle has been commonly adopted by the works [11], [35]–[38]. The undersampling strategy is also one of the solutions for imbalanced clustering, but it often faces the difficulties of gradient explosion and insufficient learning experience of positive data objects. Recently, [39] proposes an informative undersampling and boundary expansion strategy to deal with it. However, they typically use a pre-defined number of prototypes, which makes the clustering performance highly sensitive to different datasets with various distributions. Hence, their recent variants propose to adaptively generate prototypes for distribution description.

A method called SMCL [6] achieves incremental prototype learning by gradually adding seed points driven by a competitive learning mechanism to prevent the issue of dead units [12]. However, this approach lacks robustness to noise and is extremely computationally expensive due to the recursive seed points generation and merging. To address this, LDPI [19] designs an initial subcluster generation scheme, improving the clustering method of DPC [17] by automatically identifying noise points and initial subcluster centers. According to the nearest-neighbor principle, the remaining objects are classified as subcluster centers to represent local micro distributions. MCNS [40] further introduces a measure based on the reconstruction rate to select the appropriate number of clusters, enhancing convergence speed while ensuring accuracy. However, all the above-mentioned methods seek the optimal solution through iterative searching on n-scale prototypes, resulting in quadratic-level time complexity.

C. Distribution Learning

Common unsupervised distribution learning approaches include: 1) representation learning [41]–[43] that learns to project the data objects from the original distance space into a more cluster-discriminative space; 2) data summarization [44], [45] that uses a set of prototypes to describe the data distribution; and 3) Self-Organizing Map (SOM) [46]–[48] that trains a low-dimensional map to simultaneously realize dimensionality reduction and distribution description on datasets. Since the latter two types are more efficiency-promising under the scenario of ISDC, we further discuss them below.

Among the summarization-based methods, data bubbles [49] and its variants [50], [51] summarize data distribution by randomly initializing a set of prototypes to incorporate nearby data objects into groups (i.e., data bubbles). In general, their performance is sensitive to the compression rate and the initialization of prototypes. Later, the work in [52] relieves the sensitivity issue by specifically training prototypes to fit the distribution. To further preserve the embedded hierarchies of more complex data distribution, hierarchical summarization approaches [23], [24], [37] have been proposed to partition the data into microclusters, and then construct dendrograms for multi-granular distribution summarization.

SOM [46] that trains neurons constrained by mesh topology in a competitive learning way has been proposed to summarize high-dimensional data into a low-dimensional map for unbiased any-shape cluster detection [47], [53]. To improve the adaptability of SOM to represent more complex data distributions, more advanced SOMs [54]–[56] that introduce asymmetric connection weights, dynamic learning

rates, and adaptive neighborhood relationships, respectively, have been proposed in recent years. Given that conventional SOMs initialize fix-sized and grid-connected neurons, they are inflexible in distribution exploration of streaming data. To address this issue, growing self-organizing maps [23], [57] have been developed to generate neurons on demand to fit data distributions. The growing hierarchical map [58] has been proposed to recursively grow and refine the coarse-grained neurons to represent complex data distributions. Most recently, the method proposed in [59] further generalizes SOM to a sparse dictionary of prototypes with flexible free-to-update/remove neighboring relationships, thus facilitating efficient and accurate online streaming data classification. However, the above-mentioned approaches either operate in a supervised scenario, or do not specifically address the imbalance of distributions, preventing them from tackling the ISDC problem. Furthermore, SOM has also been incorporated into deep learning framework [60]. Although it facilitates a convenient end-to-end clustering, the data scale requirement for model training and the inability to adapt to concept drift limit its usage in ISDC.

III. Problem Statement

The primary cause of the unsatisfactory streaming clustering performance is the difficulty in exploring imbalanced clusters, known as the Imbalanced Streaming Data Clustering (ISDC) problem demonstrated in Fig. 1. Assuming data objects are continuously generated or collected, which can be more formally denoted as a data stream $X_\Gamma = \{X^\iota\}_{\iota=1}^{N}$ comprising N data chunks $X^\iota \in \mathbb{R}^{n\times d}$ formed at different time-stamps ι. Generally, data streams are considered to be unbounded (i.e., $N \to \infty$). Each data chunk consists of n objects denoted as a collection of n vectors $X^\iota = \{\mathbf{x}_1^\iota, \mathbf{x}_2^\iota, \ldots, \mathbf{x}_n^\iota\}$, where $\mathbf{x}_j^\iota \in \mathbb{R}^d$ represents the j-th data object within X^ι.

For ISDC, assume the objects of a chunk X^ι can be distributed to k^ι clusters denoted as $C^\iota = \{C_1^\iota, C_2^\iota, \ldots, C_{k^\iota}^\iota\}$ with the corresponding cluster centers $S^\iota = \{\mathbf{s}_1^\iota, \mathbf{s}_2^\iota, \ldots, \mathbf{s}_{k^\iota}^\iota\}$. A j-th cluster C_j^ι is a subset of X^ι. The conventional cluster objective is to partition X^ι into C^ι that minimizes:

$$SSQ(X^\iota, S^\iota) = \sum_{j=1}^{k^\iota} \sum_{\mathbf{x}\in C_j^\iota} \|\mathbf{x} - \mathbf{s}_j^\iota\|_2, \tag{1}$$

which is the sum of squared distances [61], and $\|\cdot\|_2$ denotes the L-2 norm.

When data object generation is biased towards certain clusters, the cluster-imbalanced data chunks arise, where the number of objects in the relatively large cluster can significantly exceed that in the relatively small one, which can be reflected by an Imbalance Ratio (IR):

$$IR = \frac{\max(\text{card}(C_1^\iota), \text{card}(C_2^\iota), ..., \text{card}(C_{k^\iota}^\iota))}{\min(\text{card}(C_1^\iota), \text{card}(C_2^\iota), ..., \text{card}(C_{k^\iota}^\iota))}, \tag{2}$$

where $\text{card}(\cdot)$ is a function that counts the number of elements of a set, and IR is actually the ratio between the sizes of the largest and the smallest clusters. Intuitively, $IR = 1$ indicates an extremely balanced case, while larger IR indicates a more severe cluster imbalance. Since most existing clustering algorithms implicitly assume balanced clusters and perform static data clustering, how to timely capture the key smaller clusters in ISDC is the core problem to be tackled.

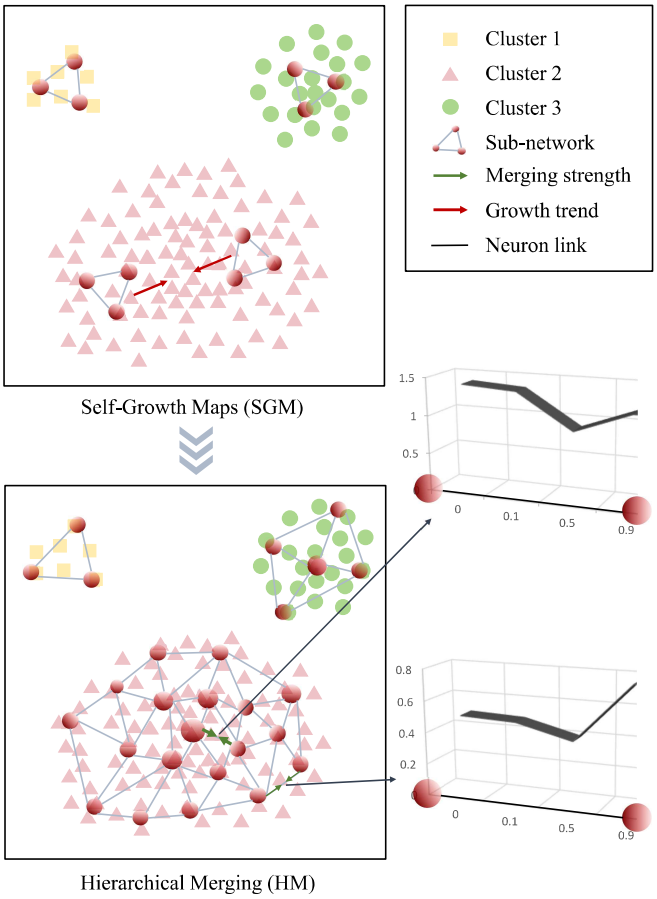


Fig. 2: Overview of the proposed SOHI framework.

IV. Proposed Method

This section provides a detailed description of the proposed SOHI approach for ISDC. SOHI contains two main steps: 1) Adaptive chunk distribution representation by Self-Growth Maps (SGM), and 2) Hierarchical Merging (HM) for imbalanced cluster exploration. The overview of the SOHI approach is shown in Fig. 2. SGM first initializes multiple subnetworks and lets them grow to fit the local object distribution (upper left in Fig. 2). When the overall data distribution can be sufficiently represented by the SGM, the neurons are hierarchically merged by the HM (lower left in Fig. 2) to explore an optimal number of imbalanced clusters based on the density gaps between merged neurons (lower right in Fig. 2).

A. SGM: Self-Growth Maps

For ISDC, it is crucial to achieve fast microcluster exploration for distribution description. Most conventional SOM-based methods initialize and train a complete map to represent the data distribution through its neurons, where excessive neurons can appear between adjacent clusters, making them indistinguishable. More specifically, the neurons are usually

treated as microcluster centers and will guide the merging of microclusters to form prominent clusters. If redundant neurons between adjacent clusters are created, clusters might be mistakenly merged, severely degrading the clustering performance. Growing Cell Structures (GCS) [62], an incremental SOM, is promising as it dynamically adapts to the changing distributions of streaming data. However, it attempts to maintain a complete map with all neurons connected, which may still incur the redundant neuron effect. Although it provides a threshold-based strategy for redundant neuron elimination, the threshold tuning is still non-trivial.

Inspired by GCS, the Self-Growth Map (SGM) is proposed to train multiple subnetworks for rapid data distribution fitting. A network A consisting of T subnetworks is initialized:

$$A = \bigcup_{l=1}^{T} A_l, \tag{3}$$

where T is typically set at a value larger than the optimal number of clusters k^* to avoid missing relatively small clusters. To ensure that each subnetwork can appropriately fit the local data distribution without overlapping, rapid Poisson disk sampling is employed for generating initial subnetworks [63] (each of which is a basic triangle 3-neuron network), which are allowed to grow independently. Planar triangle structures are adopted by the growing network rather than the high-dimensional hypertetrahedra as in the original GCS for an efficient purpose. Given a chunk X^ι, subnetworks, e.g., A_l, are trained to fit the local object distribution by:

$$\mathbf{z}_{l,s}^{\text{new}} = \mathbf{z}_{l,s}^{\text{old}} + \epsilon_b(\mathbf{x}_j^\iota - \mathbf{z}_{l,s}^{\text{old}}), \tag{4}$$

if $\mathbf{z}_{l,s}$ is the Best Matching Neuron (BMN) of object $\mathbf{x}_j^\iota$ determined by:

$$\mathbf{z}_{l,s} = \underset{\mathbf{z}_{i,r}\in A}{\operatorname{argmin}} \|\mathbf{x}_j^\iota - \mathbf{z}_{i,r}\|_2. \tag{5}$$

Note that $\mathbf{z}_{l,s}$ is a d-dimensional vector representing the s-th neuron of A_l, as the training of neurons is driven by the d-dimensional data objects from X^ι. Moreover, to ensure a smooth and efficient update of the BMNs, a small learning rate ϵ_b is adopted in Eq. (4). To ensure a structural update of the subnetwork, the update provided by $\mathbf{x}_j^\iota$ is also propagated to the neurons that are directly connected with the BMN:

$$\mathbf{z}_{l,c}^{\text{new}} = \mathbf{z}_{l,c}^{\text{old}} + \epsilon_\Omega(\mathbf{x}_j^\iota - \mathbf{z}_{l,c}^{\text{old}}), \tag{6}$$
$$\text{s.t. } \mathbf{z}_{l,c} \in \Omega(\mathbf{z}_{l,s}),$$

where $\Omega(\mathbf{z}_{l,s})$ represents the set of 1-hop adjacent neurons of $\mathbf{z}_{l,s}$, and ϵ_Ω is the corresponding learning rate. Since only the BMN and its 1-hop neighbors are trained w.r.t. each object $\mathbf{x}_j^\iota$, an efficient adaptation to new data distributions can be achieved without involving all the subnetwork's neurons. By randomly selecting data objects from X^ι for the above training, the network gradually fits the distribution of X^ι.

Due to the distribution complexity, mapping too many distant data objects to a neuron will lead to an improper distribution representation. Such an effect can be reflected by the accumulated BMN distance, which we call BMN inadaptability. For each training input $\mathbf{x}_j^\iota$, its corresponding BMN inadaptability is updated by:

$$\tau_{\mathbf{z}_{l,s}}^{\text{new}} = \tau_{\mathbf{z}_{l,s}}^{\text{old}} + \|\mathbf{x}_j^\iota - \mathbf{z}_{l,s}\|_2. \tag{7}$$

In the same subnetwork A_l, the neurons other than the BMN typically exhibit lower inadaptability with respect to the current input. Therefore, their inadaptability values are slightly decreased, with the reduction rate governed by a parameter α:

$$\tau_{\mathbf{z}_{l,q}}^{\text{new}} = \tau_{\mathbf{z}_{l,q}}^{\text{old}} - \alpha\tau_{\mathbf{z}_{l,q}}^{\text{old}}, \tag{8}$$
$$\text{s.t. } \mathbf{z}_{l,q} \neq \mathbf{z}_{l,s}.$$

When A has been trained to adapt to ρ data objects, it is necessary to evaluate whether the network needs to create new neurons or merge nearby subnetworks. A reasonable evaluation method is to compare the distance between the neurons with the highest and second highest inadaptability, e.g., $\mathbf{z}_{i,v}$ and $\mathbf{z}_{j,g}$, which belong to two different subnetworks, given by:

$$\mathbf{z}_{i,v} = \underset{\mathbf{z}_{i,s}\in A}{\operatorname{argmax}} \tau_{\mathbf{z}_{i,s}} \quad \text{and} \quad \mathbf{z}_{j,g} = \underset{\mathbf{z}_{j,s}\in A\setminus\{A_i\}}{\operatorname{argmax}} \tau_{\mathbf{z}_{j,s}}. \tag{9}$$

If the distance between $\mathbf{z}_{i,v}$ and $\mathbf{z}_{j,g}$ is larger than that between $\mathbf{z}_{i,v}$ and its furthest adjacent neuron $\mathbf{z}_{i,q}$, i.e.,

$$\|\mathbf{z}_{i,v} - \mathbf{z}_{j,g}\|_2 > \|\mathbf{z}_{i,v} - \mathbf{z}_{i,q}\|_2, \tag{10}$$
$$\text{s.t. } \mathbf{z}_{i,q} = \underset{\mathbf{z}_{i,g}\in\Omega(\mathbf{z}_{i,v})}{\operatorname{argmax}} \|\mathbf{z}_{i,g} - \mathbf{z}_{i,v}\|_2,$$

then it suggests that $\mathbf{z}_{i,v}$ may suffer from under-adaptation and be insufficient to properly represent its associated objects. To address this, a new neuron $\mathbf{z}_{i,h} = (\mathbf{z}_{i,v} + \mathbf{z}_{i,q})/2$ is created and connected to the common neighbors of $\mathbf{z}_{i,v}$ and $\mathbf{z}_{i,q}$, preserving the basic triangle topology of the network.

To facilitate sustainable map growth, an inadaptability value $\tau_{\mathbf{z}_{i,h}}^{\text{new}}$ should be assigned to the new neuron $\mathbf{z}_{i,h}$:

$$\tau_{\mathbf{z}_{i,h}}^{\text{new}} = \sum_{\mathbf{z}_{i,c}\in\Omega(\mathbf{z}_{i,h})} \frac{\zeta_{\mathbf{z}_{i,c}}^{\text{old}} - \zeta_{\mathbf{z}_{i,c}}^{\text{new}}}{\zeta_{\mathbf{z}_{i,c}}^{\text{old}}} \tau_{\mathbf{z}_{i,c}}^{\text{old}}, \tag{11}$$

which is the sum of the reduced inadaptability of its neighboring neurons. The reduced amount of inadaptability is reflected by the receptive field of a neuron:

$$\zeta_{\mathbf{z}_{i,c}} = \frac{1}{\text{card}(\Omega(\mathbf{z}_{i,c}))} \sum_{\mathbf{z}_{i,t}\in\Omega(\mathbf{z}_{i,c})} \|\mathbf{z}_{i,c} - \mathbf{z}_{i,t}\|_2, \tag{12}$$

which is the average distance between $\mathbf{z}_{i,c}$ and its neighboring neurons. As the new $\mathbf{z}_{i,h}$ splits up the receptive fields from its neighboring neurons, the inadaptability of each neighboring neuron $\mathbf{z}_{i,c}$ is updated with a corresponding reduction as:

$$\tau_{\mathbf{z}_{i,c}}^{\text{new}} = \tau_{\mathbf{z}_{i,c}}^{\text{old}} - \frac{\zeta_{\mathbf{z}_{i,c}}^{\text{old}} - \zeta_{\mathbf{z}_{i,c}}^{\text{new}}}{\zeta_{\mathbf{z}_{i,c}}^{\text{old}}} \tau_{\mathbf{z}_{i,c}}^{\text{old}}, \tag{13}$$
$$\text{s.t.} \quad \mathbf{z}_{i,c} \in \Omega(\mathbf{z}_{i,h}).$$

If the distance between $\mathbf{z}_{i,v}$ and $\mathbf{z}_{j,g}$ is not larger than the distance between $\mathbf{z}_{i,v}$ and $\mathbf{z}_{i,q}$, it implies that the two corresponding subnetworks A_i and A_j are too close, and a merging procedure should be launched. Specifically, an

Algorithm 1: IGCM: Inadaptability measure-Guided neuron Creation or subnetwork Merging

Input: Network A.
Output: Updated network A with created neurons or merged subnetworks.

Find $\mathbf{z}_{i,v}$ and $\mathbf{z}_{j,g}$ with the largest and second largest inadaptability by Eq. (9);
Find the furthest adjacent neuron $\mathbf{z}_{i,q}$ of $\mathbf{z}_{i,v}$ from A_i;
if $\|\mathbf{z}_{i,v} - \mathbf{z}_{j,g}\|_2 > \|\mathbf{z}_{i,v} - \mathbf{z}_{i,q}\|_2$ **then**
    Create $\mathbf{z}_{i,h}$ with its inadaptability $\tau_{\mathbf{z}_{i,h}}$ by Eq. (11);
    Update inadaptabilities of $\Omega(\mathbf{z}_{i,h})$ by Eq. (13);
    Create edges to connect $\mathbf{z}_{i,h}$ to $\mathbf{z}_{i,v}$ and $\mathbf{z}_{i,q}$;
else
    Connect $\mathbf{z}_{i,v}$ and $\mathbf{z}_{i,q}$ to $\mathbf{z}_{j,g}$, merge A_i and A_j;
end

edge is created to connect $\mathbf{z}_{i,v}$ and $\mathbf{z}_{i,q}$ to $\mathbf{z}_{j,g}$, maintaining the triangular topology within and across subnetworks. The complete algorithm of inadaptability-guided neuron creation and subnetwork merging is summarized in Algorithm 1.

Remark 1. **Merits of Subnetworks:** The multi-subnetwork design naturally circumvents the thorny elimination of redundant neurons located at the distribution boundary of clusters. Moreover, it allows for the flexible removal of subnetworks that do not fit the data distribution in a new data chunk without affecting the structure of the other subnetworks. In addition, parallel computing can be performed to accelerate the training of the independent subnetworks before their merging.

Remark 2. **Rationality of the IGCM algorithm:** Subnetwork merging is conducted by considering the neurons with high inadaptability and their inter-subnetwork distance. Inadaptability filters most neurons that can well represent the corresponding objects, and the designed merging process will not merge two nearby neurons with relatively low inadaptability. This is because the lower inadaptability indicates a clear distribution boundary between the two microclusters.

Through the SGM training, a network A containing Q neurons organized in a certain number of subnetworks is obtained. Each neuron can be treated as a microcluster center to partition the data chunk into Q microclusters. A is also utilized for quickly retrieving the neighboring neurons, which is significant in accelerating the hierarchical microcluster merging process in the next subsection.

B. HM: Hierarchical Merging for Imbalanced Clustering

With the distribution knowledge of the current data chunk obtained through SGM training, the microclusters are merged to obtain a proper number of imbalanced clusters. The topological structure of the neurons is exploited to accelerate the merging process. Since clusters corresponding to non-adjacent neurons are less likely to be merged, the topology is utilized as a retrieval structure that only allows the merging of adjacent neurons, thereby significantly avoiding the laborious traversing of all the possible microcluster pairs during the merging.

To judge the merging of two clusters, the concept of density gap is introduced to describe the prominence of their boundary. Based on this, global compactness and global separability can be derived as measures to monitor the merging process and guide the selection of the optimal number of clusters. Specifically, Q microclusters $\{G_1, G_2, ..., G_Q\}$ obtained through SGM are the initial microclusters. During their merging, the current k clusters composed of a certain number of microclusters are denoted as $\Phi(k) = \{C_1, C_2, ..., C_k\}$, with their centers denoted as $S = \{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_k\}$. Note that these cluster centers are not necessarily original neurons, as each cluster may consist of data objects belonging to multiple neurons' corresponding microclusters. Given two clusters C_i and C_j, their merging is considered by projecting all their objects onto the 1-D space passing through their centers $\mathbf{s}_i$ and $\mathbf{s}_j$:

$$\mathbf{x}' = \frac{(\mathbf{x} - \bar{\mathbf{s}})^T(\mathbf{s}_i - \mathbf{s}_j)}{\|\mathbf{s}_i - \mathbf{s}_j\|_2}, \tag{14}$$

where $\bar{\mathbf{s}} = (\mathbf{s}_i + \mathbf{s}_j)/2$. To characterize the 1-D Gaussian mixture probability density distribution of C_i and C_j, their objects are further mapped onto the 1-D space with respective cluster centers as 0.5 and -0.5:

$$M(u; i, j) = \frac{\text{card}(C_i) f(u|0.5, \sigma_i^2) + \text{card}(C_j) f(u|-0.5, \sigma_j^2)}{\text{card}(C_i) + \text{card}(C_j)}, \tag{15}$$

where $f(u|0.5, \sigma_i^2)$ describes the probability density distribution of C_i, with variance σ_i^2 computed from the objects in C_i after mapping them to the 1-D space crossing $\mathbf{s}_i$ and $\mathbf{s}_j$. Then the density gap between C_i and C_j can be defined as:

$$m_{ij} = \frac{1}{\min\limits_{u \in U} M(u; i, j)}, \tag{16}$$

where $U = \{-0.5, -0.49, \ldots, 0.5\}$ is a traverse set of u with a step size of 0.01. Since a smaller $\min_{u \in U} M(u; i, j)$ reflects a more prominent distribution density gap between two clusters, then the corresponding m_{ij} will be larger, indicating that C_i and C_j are more unsuitable for merging. Conversely, C_i and C_j are selected for merging if their m_{ij} reaches the global minimum among all the cluster pairs in $\Phi(k)$, which can be defined as the global compactness:

$$\theta_k = \min_{C_i, C_j \in \Phi(k)} m_{ij}. \tag{17}$$

It reflects the compactness of the clusters that will be merged currently in $\Phi(k)$. A lower θ_k indicates that the new cluster formed by merging will be more compact, and thus a lower θ_k is preferred by the typical clustering objective. After the merging, a new cluster is formed as $C_v = \{C_i, C_j\}$ and added to $\Phi(k)$, while the original C_i and C_j are removed:

$$\Phi(k) \setminus \{C_i, C_j\} \quad \text{and} \quad \Phi(k) \cup C_v. \tag{18}$$

Then a new status $\Phi(k-1)$ arises as the number of clusters becomes $k-1$. The merging stops when all clusters are merged into one cluster, i.e., $k = 1$.

An optimal k can be selected by evaluating the merging process. Since θ_k may keep monotonic increasing due to the gradual merging of adjacent microclusters with lower density

Algorithm 2: SOHI: Self-grOwth maps-guided Hierarchical merging for Imbalanced data clustering

Input: Data chunk X^{ι}, learning rates ϵ_{Ω}, ϵ_b, reduction coefficient α, and number of nearest neighbors s.
Output: $\hat{k}^*$ clusters $\{C_1, C_2, ..., C_{\hat{k}^*}\}$.

1. **SGM: Self-Growth Maps:**
   1.1 Initialize network A with T subnetworks;
   1.2 Train A according to Eqs. (4)-(6) by randomly selecting objects from X^{ι};
   1.3 Update A with created neurons or merged subnetworks according to Algorithm 1.
2. **HM: Hierarchical Merging:**
   2.1 Calculate density gap of adjacent microclusters according to Eqs. (14)-(16);
   2.2 Compute global compactness according to Eq. (17) and merge all the clusters through A-guided retrieval;
   2.3 Compute global separability according to Eq. (19);
   2.4 Obtain $\hat{k}^*$ clusters according to Eq. (20).

gaps, it is incompetent in determining the optimal number of clusters. Therefore, a global separability is also introduced:

$$\omega_k = \max_{C_i \in \Phi(k)} \left(\sum_{\mathbf{x}_j \in C_i} \frac{N_s(\mathbf{x}_j) \setminus \{N_s(\mathbf{x}_j) \cap C_i\}}{s} \right), \quad (19)$$

where s refers to the number of global nearest neighbors of an object $\mathbf{x}_j \in C_i$, and $N_s(\mathbf{x}_j) \setminus \{N_s(\mathbf{x}_j) \cap C_i\}$ represents the number of objects that are the s-nearest neighbors of $\mathbf{x}_j$ but are not included in the cluster C_i. It is intuitive that if a cluster C_i is located far away from the other clusters, a small $(N_s(\mathbf{x}_j) \setminus \{N_s(\mathbf{x}_j) \cap C_i\})/s$ is yielded as most s-nearest neighbors of $\mathbf{x}_j$ may be included in its cluster C_i. The cluster size of C_i is not introduced to this measure to ensure equal status to imbalanced clusters, thus avoiding overlooking relatively small clusters in ISDC. Accordingly, a lower ω_k indicates a higher separability of the current clusters, which is preferred by the typical clustering objective. As merging progresses, ω_k decreases because more inseparable adjacent clusters are merged to form larger prominent clusters.

By collectively considering the global compactness and global separability, a proper number of clusters can be automatically determined as:

$$\hat{k}^* = \underset{k \in \{1,2,...,Q\}}{\arg\min} \left(\frac{\theta_k}{\max(\theta_1, \theta_2, ..., \theta_Q)} + \frac{\omega_k}{\max(\omega_1, \omega_2, ..., \omega_Q)} \right), \quad (20)$$

where the two denominators are utilized to normalize θ_k and ω_k into the same scale to ensure their comparability, and form a trade-off between them. Consequently, the optimal number of clusters k^* is estimated as the knee point [6], i.e., $\hat{k}^*$, of the composite curve of θ_k and ω_k.

C. Overall Algorithm and Complexity Analysis

The complete SOHI is summarized as Algorithm 2. SGM is first implemented to grow and merge subnetworks to represent the distribution of the imbalanced dataset. Then, the microclusters corresponding to the neurons are adaptively merged using HM, and an optimal number of clusters is obtained by evaluating the merging process. The time and space complexity of SOHI is analyzed in the following.

Theorem 1. **Time Complexity:** Given an n-object chunk X^{ι}, and its Q-neuron SGM. The time complexity of SOHI is $O(nQ^2d)$.

Proof. Time complexity of SGM: The growth of SGM is driven by all the n objects. For each object, the time complexity is analyzed as follows: The distances between the object and all Q neurons are computed to find the BMN, which incurs a complexity of $O(Qd)$, where d denotes the feature dimensionality. The BMN and its B adjacent neurons are updated, costing $O(Bd)$, where B is the branching factor of the network. The inadaptability of the BMN and at most Q neurons in the same subnetwork are also updated, resulting in a complexity of $O(d+Q)$. Therefore, updating the neurons and their inadaptabilities for all n objects results in a time complexity of $O(nBd+nQ+nd)$. Since the neuron creation or subnetworks merging of A is triggered after every ρ objects are input, Algorithm 1 will be implemented $\frac{n}{\rho}$ times in total. The time complexity for each implementation is analyzed as follows: Identifying the neurons with the largest and second-largest inadaptabilities, e.g., $\mathbf{z}_{i,v}$ and $\mathbf{z}_{j,g}$, requires $O(Q)$ time (line 1 of Algorithm 1), as the inadaptability of each neuron has been prepared. Then, finding the furthest adjacent neuron $\mathbf{z}_{i,q}$ among the B neighbors of $\mathbf{z}_{i,v}$ takes $O(Bd)$ time (line 2). Computing the distances for judging the condition (line 3) takes $O(2d)$ time. Therefore, $O(Q+Bd+2d)$ is required to determine whether a new neuron should be created or subnetworks should be merged. If a new neuron $\mathbf{z}_{i,h}$ is to be created (lines 3–6), it takes $O(2d+Bd)$ to initialize the neuron as the mean of the two adjacent neurons and to update the inadaptability according to its B neighbors. The B neighbors of the new neurons should also be updated, each of which involves $O(Bd)$ operations. Thus, the total cost for B neighbor updating is $O(B^2d)$. Hence, the complexity for a new neuron creation is $O(2d+Bd+B^2d)$. If subnetwork merging is triggered (lines 8) to connect three neurons, a negligible computational complexity is involved. Consequently, the worst-case time complexity for one iteration of Algorithm 1 is $O(Q+Bd+2d+2d+Bd+B^2d)$, and for all $\frac{n}{\rho}$ implementations, it will be $O(\frac{n}{\rho}Q+\frac{n}{\rho}B^2d)$. In summary, the overall time complexity of the SGM process is $O(nQ+nd+nB^2d)$.

Time complexity of HM: The SGM topology is leveraged to accelerate the merging, and we analyze the complexity in two main stages: 1) IntrA-Subnetwork Merging (IASM), and 2) IntEr-Subnetwork Merging (IESM). In IASM, at most Q microclusters should be considered within the same subnetwork for merging. Each merge estimates a density distribution based on at most n objects from two candidate clusters with complexity $O(nd)$. The two candidates are chosen based on the density gap between all possible pairs of clusters. Since there are at most Q microclusters, and for each microcluster, only the adjacent B microclusters are considered for merging, the complexity is thus $O(nQBd)$. Searching for the clus-

ter pair with the smallest gap involves complexity $O(QB)$. Since there are at most Q merges, the time complexity is $O(Q(nd + nQBd + QB)) = O(nQd + nQ^2Bd + Q^2B)$, which can be simplified to $O(nQ^2Bd)$. For IESM, merging the current clusters corresponding to the subnetworks should be conducted at most T times. Each cluster contains at most n objects, and each merge requires up to T^2 density gap computations with complexity $O(nT^2d)$. The overall IESM complexity is thus $O(nT^3d)$. In addition to the above two stages, each merge also involves calculating global separability. This process is analyzed as follows: The most similar s neighbors of each object should be identified by computing pairwise similarities of n objects, which incurs a complexity of $O(n^2d)$. Then the global separability can be computed by traversing all n objects and their s neighbors, which takes $O(ns)$ complexity. Therefore, the overall complexity for global separability computation is $O(n^2d + ns)$. Since at most Q merges are conducted to merge all Q microclusters, the overall complexity for the separability computation is $O(Q(n^2d + ns)) = O(n^2Qd + nQs)$. An alternative efficient way for computing the separability is to treat neurons as objects. Specifically, it requires finding the s neighbors of each of the Q neurons based on the $Q \times Q$ similarity matrix, incurring a time complexity of $O(Q^3d + Q^2s)$ for Q merges in total. In summary, the total time complexity of the HM process is $O(nQ^2Bd + nT^3d + Q^3d + Q^2s)$, which can be simplified to $O(nQ^2Bd)$, given that $Q \ll n$, and both T and s are small constants in most cases.

The overall time complexity of SOHI, combining the complexity of SGM and HM, is $O(nQ + nd + nB^2d + nQ^2Bd)$. Since B is also a small constant, the time complexity of SOHI can be simplified to $O(nQ^2d)$.

□

Theorem 2. **Space Complexity:** Given an n-object chunk X^ι, and its Q-neuron SGM. The space complexity of SOHI is $S(nd + nQ)$.

Proof. Space complexity of SGM: The growth of SGM involves the storage of X^ι as an $n \times d$ matrix. The SGM is with Q neurons described by d-dimensional vectors, and each neuron is connected to at most B neighbors, where B is the branching factor. Thus, the SGM can be described by a $Q \times d$ matrix and a $Q \times B$ adjacency matrix for neuron and linkage description, respectively. The object-neuron similarity is stored in an $n \times Q$ matrix. A Q-dimensional vector is also required to record the inadaptability of the Q neurons. Therefore, the overall space complexity of SGM is $S(nd + Qd + QB + nQ + Q)$.

Space complexity of HM: Based on X^ι, SGM, and the object-neuron similarity matrix stored in the SGM phase, HM needs additional space to store the density gaps between at most $Q(Q-1)/2$ pairs of clusters during merging. Global separability computation at the object level needs the storage of s neighbors for each of the n objects. Accordingly, the space complexity required by HM is $S(Q^2 + ns)$.

The overall space complexity of SOHI is thus $S(nd + Qd + QB + nQ + Q + Q^2 + ns)$, which can be simplified to $S(nd + nQ)$, given that $Q \ll n$, and B and s are small constants in most cases. □

Algorithm 3: TLRS: Two-Layer Random Sampling

Input: The whole original dataset X, the maximum imbalance ratio IR, true number of clusters k^*, true clustering partition C.
Output: Data chunk X^ι and its clustering partition C^ι.

Randomly set cluster number as an integer : $2 \le k^\iota \le k^*$; Initialize a imbalance ratio set IR^ι as empty arrays;
for $i \leftarrow 1$ *to* $k^\iota - 1$ **do**
  $IR_i^\iota \leftarrow \text{DiscreteUniform}([1, IR])$;
end
$IR^\iota \leftarrow$ sort IR^ι in ascending order;
$\text{card}(C_1^\iota) \leftarrow \text{card}(C_1)$;
for $i \leftarrow 2$ *to* k^ι **do**
  if $IR_{i-1}^\iota \cdot \mathit{card}(C_{i-1}) \le \mathit{card}(C_i)$ **then**
    $\text{card}(C_i^\iota) \leftarrow IR_{i-1}^\iota \cdot \text{card}(C_{i-1})$;
  else
    $\text{card}(C_i^\iota) \leftarrow \text{card}(C_i)$;
  end
end
C_i^ι data objects are randomly taken from the i-th cluster and form X^ι.

TABLE I: Statistical information of the eleven datasets. d, n, k^*, and IR represent the numbers of attributes, data objects, true clusters, and the imbalance ratio, respectively.

No.	Dataset	Abbrev.	d	n	IR	k^*
1	Gaussian	GA	2	2000	19.87	4
2	IDS2	ID	2	3200	10.00	5
3	Abalone	AB	8	4177	689.00	28
4	Car Evaluation	CE	6	1728	18.62	4
5	Haberman's Survival	HS	3	306	2.78	2
6	Heart Failure	HF	12	299	2.11	2
7	Land Mines	LM	3	338	1.09	5
8	Page Blocks	PB	10	5473	175.43	5
9	Raisin	RA	7	900	1.00	2
10	Seeds	SE	7	210	1.00	3
11	Wholesale Customers	WC	7	440	6.72	3

V. Experiments

Three experiments, i.e., efficiency evaluation, clustering accuracy evaluation, and ablation study, have been conducted on eleven datasets by comparing ten counterparts, including eight existing methods and two ablated versions of SOHI.

A. Experimental Settings

Eleven datasets, including two synthetic and nine real datasets with varying sizes, dimensions, and distribution types, are utilized for the experiments. Their statistics are provided in Table I. ID and GA [11] are synthesized by applying a mixture of bivariate Gaussian density functions [6]. All real datasets are obtained from the UCI Machine Learning Repository [64]. Min-max normalization is adopted to pre-process each feature into the identical value domain $[0, 1]$.

A streaming data chunk generation algorithm named Two-Layer Random Sampling (TLRS) described in Algorithm 3 is designed to more realistically validate the performance of clustering methods on the ISDC problem. A selected dataset serves

TABLE II: Clustering performance of different methods evaluated by the ARI (↑), NMI (↑) and DBI (↓) metrics. The results marked in **Orange** and **Gray** colors indicate the best and second-best results on each dataset, respectively.

Dataset	Metric	BIRCH	StreamKM++	CPCL	SMCL	IGMTT	DenSOINN	LDPI	M3W	SOHI (ours)
GA	ARI	0.2360±0.00	0.0098±0.01	-0.0002±0.01	0.9252±0.06	0.5725±0.22	0.3234±0.01	0.0226±0.07	0.8264±0.13	0.9667±0.01
	NMI	0.0053±0.00	0.0122±0.00	0.0114±0.00	0.8685±0.08	0.6744±0.12	0.4707±0.00	0.0480±0.12	0.8406±0.07	0.9447±0.02
	DBI	25.0641±2.85	0.7303±0.14	0.9589±0.16	0.5321±0.05	0.9871±0.35	1.3108±4.82	0.9864±0.28	3.6246±1.91	0.4879±0.01
ID	ARI	0.0008±0.00	0.0077±0.00	0.0118±0.01	1.0000±0.00	0.6611±0.10	0.0055±0.00	0.5959±0.00	0.8528±0.00	0.9930±0.01
	NMI	0.0043±0.00	0.0085±0.00	0.0168±0.00	1.0000±0.00	0.7668±0.06	0.0097±0.00	0.6983±0.00	0.8367±0.00	0.9901±0.01
	DBI	52.6033±13.26	0.2214±0.00	0.8440±0.05	0.2084±0.00	0.7807±0.16	50.8885±8.07	0.2198±0.00	1.4958±0.01	0.2194±0.01
AB	ARI	0.0027±0.00	0.0610±0.17	0.0031±0.01	-0.2316±0.01	0.0928±0.04	0.0001±0.00	0.0010±0.01	0.2153±0.03	0.1431±0.03
	NMI	0.0862±0.01	0.1741±0.21	0.0795±0.03	0.2140±0.07	0.2535±0.05	0.0650±0.01	0.0289±0.00	0.4147±0.05	0.2819±0.07
	DBI	0.8233±0.04	0.7427±0.08	0.8835±0.45	0.9685±0.16	1.6856±0.18	4.4526±1.97	0.6181±0.13	2.2018±1.66	0.6918±0.26
CE	ARI	-0.0004±0.00	0.0847±0.00	-0.0076±0.02	-0.0895±0.37	0.0069±0.02	-0.0010±0.00	0.0092±0.02	–	0.1138±0.07
	NMI	0.0052±0.00	0.0139±0.00	0.0120±0.01	0.0424±0.24	0.0652±0.01	0.0052±0.00	0.0243±0.01	–	0.0837±0.06
	DBI	20.8972±1.61	1.4144±0.01	3.0599±0.71	3.5717±1.43	1.7767±0.10	20.8279±1.58	1.8903±0.21	–	1.9068±0.38
HS	ARI	-0.0028±0.01	0.0465±0.02	0.0059±0.03	-0.0166±0.01	0.0018±0.01	0.0066±0.01	-0.0022±0.03	-0.0101±0.02	0.0613±0.03
	NMI	0.0046±0.00	0.0198±0.01	0.0140±0.01	0.0077±0.00	0.0071±0.00	0.0105±0.00	0.0202±0.02	0.0127±0.01	0.0251±0.01
	DBI	9.5495±1.91	0.9385±0.01	1.3509±0.26	0.8764±0.05	1.5378±0.56	15.8149±3.38	0.3927±0.15	0.8223±0.01	0.7448±0.16
HF	ARI	0.0039±0.01	0.0333±0.02	0.0200±0.03	0.0058±0.01	0.0046±0.02	0.0041±0.00	–	0.0035±0.00	0.0346±0.03
	NMI	0.0049±0.00	0.0268±0.02	0.0106±0.01	0.0038±0.00	0.0047±0.00	0.0080±0.00	–	0.0389±0.00	0.0117±0.01
	DBI	27.3766±19.03	0.3158±0.26	0.6165±0.03	1.9381±0.80	2.2353±0.14	46.5634±19.15	–	1.7595±0.00	0.5804±0.11
LM	ARI	0.0010±0.00	0.0130±0.01	0.0585±0.01	0.0419±0.02	0.0147±0.02	0.0028±0.00	0.0059±0.01	0.0189±0.03	0.1066±0.07
	NMI	0.0179±0.00	0.0311±0.01	0.1037±0.01	0.1660±0.04	0.0446±0.04	0.0182±0.00	0.0405±0.02	0.1262±0.06	0.2033±0.05
	DBI	17.8274±2.17	0.9159±0.03	0.8714±0.02	0.9369±0.04	0.9819±0.04	18.4434±2.68	0.8165±0.12	1.8742±0.34	0.9554±0.31
PB	ARI	-0.0011±0.00	0.0112±0.01	-0.0059±0.01	-0.0323±0.01	0.1224±0.04	0.0034±0.00	0.0103±0.01	0.0315±0.03	0.1336±0.08
	NMI	0.0123±0.00	0.0282±0.01	0.0076±0.00	0.0382±0.01	0.1936±0.04	0.0170±0.00	0.0131±0.01	0.2376±0.01	0.1491±0.04
	DBI	5.5314±0.74	0.9143±0.02	0.6875±0.09	1.1289±0.04	1.0646±0.20	17.8086±1.16	0.4487±0.11	1.174±0.10	0.5160±0.27
RA	ARI	0.0009±0.01	0.0361±0.02	0.0070±0.01	0.5173±0.22	0.2135±0.10	0.0137±0.01	0.0053±0.01	0.0642±0.00	0.5319±0.02
	NMI	0.0027±0.00	0.0083±0.01	0.0124±0.00	0.4568±0.22	0.2024±0.08	0.0105±0.00	0.0020±0.00	0.1757±0.00	0.3979±0.01
	DBI	21.2803±9.64	0.5071±0.00	0.5644±0.01	0.9011±0.14	1.6233±0.57	84.9281±22.33	0.3169±0.00	5.3499±0.00	0.3607±0.11
SE	ARI	0.0023±0.01	0.0648±0.02	0.0848±0.01	0.6047±0.14	0.3277±0.13	0.0039±0.01	0.0103±0.06	0.3040±0.12	0.8684±0.05
	NMI	0.0260±0.00	0.0600±0.01	0.1218±0.00	0.6790±0.09	0.4457±0.11	0.0236±0.00	0.0344±0.04	0.4591±0.05	0.7920±0.07
	DBI	13.4789±2.73	0.8175±0.06	0.9336±2.73	0.8401±0.11	1.6062±0.45	12.0563±2.43	0.6264±0.03	1.2176±0.13	0.6531±0.09
WC	ARI	0.0004±0.00	0.0205±0.01	-0.0017±0.01	0.0535±0.02	0.0420±0.01	0.0032±0.00	-0.0025±0.01	-0.0033±0.00	0.0036±0.00
	NMI	0.0088±0.00	0.0283±0.01	0.0141±0.01	0.0501±0.01	0.0418±0.02	0.0124±0.00	0.0125±0.00	0.0179±0.00	0.0473±0.01
	DBI	12.1115±1.56	0.8360±0.08	1.1122±0.11	0.5299±0.06	1.0990±0.12	11.1380±1.84	1.2536±0.08	0.6666±0.07	0.3799±0.11

as the basis for generating data chunks. Through controlling the imbalance ratio and number of imbalanced clusters, TLRS can generate arbitrary-sized data chunks with k^* imbalanced clusters to include various imbalance states of streaming data while preserving the original data distribution. The TLRS serves to enhance the diversity of experimental data, allowing a more convincing ISDC performance evaluation under various imbalanced scenarios.

The eight counterparts include the advanced automatic k-selection algorithms (CPCL [14], M3W [15]), advanced algorithms designed for streaming data (StreamKM++ [20], BIRCH [24]), and state-of-the-art methods suitable for static imbalanced data (SMCL [6], IGMTT [23], DenSOINN [13], LDPI [19]). For StreamKM++ and BIRCH, the number of clusters needs to be specified in advance. In contrast, CPCL, SMCL, IGMTT, DenSOINN, LDPI, M3W and our method can adaptively determine the number of clusters without prior specification. Hyperparameters of our method are set as follows: the initial number of subnetworks is set at $T = 15$, the number of objects ρ for triggering Algorithm 1 is set at $\rho = 100$ recommended by [62], the number of nearest neighbors is set at $s = 10$, and the learning rates ϵ_b, ϵ_Ω, and α are set at 0.6, 0.02, and 0.005, respectively. The other parameters for all the compared methods are set following the recommendations in the source literature.

Three metrics, i.e., Adjusted Rand Index (ARI), Normalized Mutual Information (NMI), and Davies-Bouldin Index (DBI), are utilized for evaluation. ARI indicates the agreement in clustering that would be expected by random chance, which is a discriminative index with a range of $[-1, 1]$. NMI reflects the correlation between the clustering results and the given labels from the perspective of information theory, and its value range is $[0, 1]$. For both ARI and NMI, the larger their values, the better the clustering performance is. DBI is derived from the principles of internal consistency and density gap in cluster analysis. It evaluates the quality of clustering by measuring the distinction between adjacent clusters while also considering the tightness within clusters. A lower DBI value indicates better clustering, as it suggests that data objects within clusters are more compact and there is a higher degree of density gap between different clusters. The DBI range is $[0, \infty)$, where 0 represents a perfect clustering effect, i.e., clusters are very tight internally and completely separable from the others. The experiments are programmed using Python 3.11 and implemented using a workstation with 16GB RAM and 2.4GHz AMD R9 7940HX CPU.

B. *Efficiency Evaluation*

The impact of data size on the running time of clustering methods is studied by plotting their running time with the increase of data size in Fig. 3. It can be seen that the proposed SOHI has a similar running time as that of StreamKM++, BIRCH, and CPCL, thanks to low complexity, while the group of SMCL and DenSOINN is with heavy computational cost due to their polynomial time complexity.

TABLE III: Comparison of ablated SOHI variants. The symbol "✓" indicates that the corresponding component is not ablated. The symbol "‡" represents the expected performance degradation compared to SOHI (i.e., the first row on each dataset).

Dataset	MS	SGM	HM	ARI	NMI	DBI
GA	✓	✓	✓	0.9674±0.01	0.9228±0.02	0.4969±0.02
		✓	✓	0.8207±0.25‡	0.7996±0.20‡	0.5174±0.08‡
			✓	0.7298±0.07‡	0.6746±0.05‡	1.0782±0.18‡
ID	✓	✓	✓	0.9958±0.09	0.9889±0.05	0.4154±0.07
		✓	✓	0.8357±0.03‡	0.8832±0.02‡	0.5835±0.08‡
			✓	0.6516±0.00‡	0.7029±0.00‡	0.7464±0.00‡
AB	✓	✓	✓	0.0386±0.00	0.0910±0.00	0.5494±0.11
		✓	✓	0.0385±0.00‡	0.0907±0.00‡	1.1043±0.04‡
			✓	0.0048±0.00‡	0.0347±0.00‡	0.6926±0.00‡
CE	✓	✓	✓	0.1087±0.03	0.0832±0.02	1.6209±0.02
		✓	✓	0.0258±0.08‡	0.0223±0.04‡	1.8348±0.10‡
			✓	0.0151±0.00‡	0.0072±0.00‡	2.4099±0.00‡
HS	✓	✓	✓	0.1357±0.04	0.0647±0.02	0.3658±0.03
		✓	✓	0.1103±0.01‡	0.0468±0.00‡	0.6296±0.18‡
			✓	0.0143±0.00‡	0.0051±0.00‡	1.6586±0.02‡
HF	✓	✓	✓	0.0475±0.00	0.0206±0.01	0.1242±0.01
		✓	✓	0.0053±0.00‡	0.0016±0.00‡	0.7219±0.00‡
			✓	0.0135±0.00‡	0.0016±0.00‡	0.5659±0.00‡
LM	✓	✓	✓	0.1001±0.04	0.1937±0.08	0.8503±0.04
		✓	✓	-0.0016±0.02‡	0.0074±0.02‡	1.3347±0.06‡
			✓	-0.0039±0.00‡	0.0025±0.00‡	1.1030±0.00‡
PB	✓	✓	✓	0.1055±0.02	0.1047±0.02	0.1565±0.06
		✓	✓	0.0891±0.02‡	0.0979±0.02‡	1.6877±0.25‡
			✓	-0.0127±0.00‡	0.0276±0.00‡	0.9413±0.00‡
RA	✓	✓	✓	0.4033±0.08	0.3444±0.02	0.2574±0.08
		✓	✓	0.3673±0.01‡	0.2911±0.02‡	0.6584±0.02‡
			✓	0.1758±0.00‡	0.2554±0.00‡	0.5323±0.00‡
SE	✓	✓	✓	0.7121±0.20	0.7154±0.19	0.5450±0.02
		✓	✓	0.4052±0.03‡	0.4996±0.01‡	0.7902±0.00‡
			✓	0.0334±0.01‡	0.0743±0.02‡	1.6588±0.13‡
WC	✓	✓	✓	0.0447±0.00	0.0419±0.01	0.1707±0.03
		✓	✓	0.0295±0.01‡	0.0074±0.00‡	1.9827±0.35‡
			✓	-0.0106±0.00‡	0.0052±0.00‡	1.2073±0.00‡

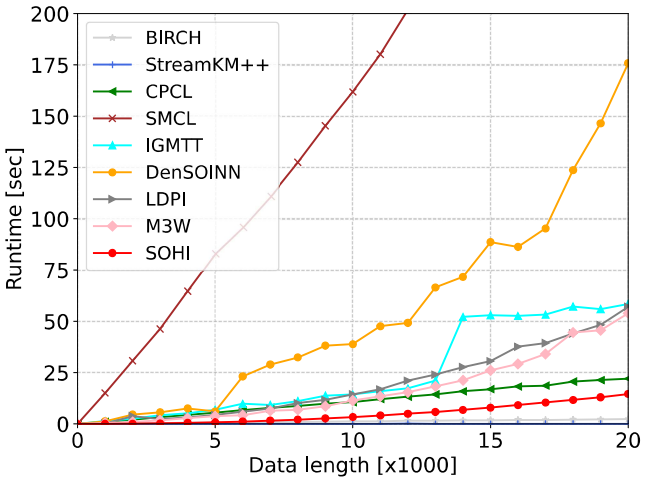


Fig. 3: Effect of increasing data size on running time.

C. Clustering Accuracy Evaluation

The clustering performance of different clustering methods is compared on all the datasets, where each dataset is generated with 10 chunks, and the average experimental results are reported. The best and second-best-performing methods on each dataset are highlighted in orange and gray, respectively.

From Table II, it can be observed that the proposed SOHI performs the best in general, winning or being the runner-up in 28 out of 33 comparisons. In comparison with the fast clustering methods, i.e., StreamKM++, BIRCH, and IGMTT, our SOHI demonstrates its superiority in accuracy, as it is competent in detecting imbalanced clusters, while the three fast methods do not adopt a mechanism to specially take the imbalanced issue into account. The results of LDPI on HF dataset and the results of M3W on CE dataset are not reported, because they wrongly group all the data objects into one cluster. LDPI's criteria for selecting initial subclusters is highly sensitive to the data distributions. It fails in clustering the HF dataset because a part of the objects is distributed with an extremely high density than the other parts. This results in only one subcluster being initialized by the LDPI. As for M3W, it fails on the CE dataset because the data objects are distributed in a sparse and relatively uniform way. Even the smallest number of neighbors suggested by M3W is still too large, leading to the merging of all the initialized cores to form a single cluster. In addition, since M3W tends to partition data objects into a larger number of smaller clusters, it is reasonable that M3W achieves superior clustering performance on AB dataset with many (i.e., $k^* = 28$) clusters and performs well in terms of the internal DBI index. In general, SOHI demonstrates higher clustering accuracy in comparison with the fast algorithms for streaming data processing, while being extremely competitive in accuracy compared to the state-of-the-art algorithms proposed for static data clustering.

Then we evaluate the performance of different methods in adapting to consecutive imbalanced streaming data chunks. 50 chunks are generated using TLRS, and the ARI, NMI, and DBI performance of the methods per chunk (time-stamp) is shown in Figs. 4 - 6, respectively. It can be observed that SOHI still significantly outperforms the other methods in general, which conforms with the observations of Table II. Since the static data-oriented CPCL, M3W, SMCL, DenSOINN, and LDPI are also executable on streaming data chunks by treating each chunk as a static dataset, their ISDC performance is also reported. In summary, the proposed SOHI is superior in terms of clustering accuracy in handling imbalanced streaming data chunks with various imbalance ratios.

D. Ablation Study

To more specifically validate SOHI's effectiveness, we conduct ablation experiments to compare SOHI and its ablated variants. Since the merits of SOHI mainly stem from the initialization of Multiple Subnetworks (MS), the learning of SGM, and the HM module that obtains a proper number of clusters, the ablated variants are formed as follows: 1) To ablated MS, a single network is trained without MS initialization and subnetwork fusion; 2) To ablate SGM, the network's

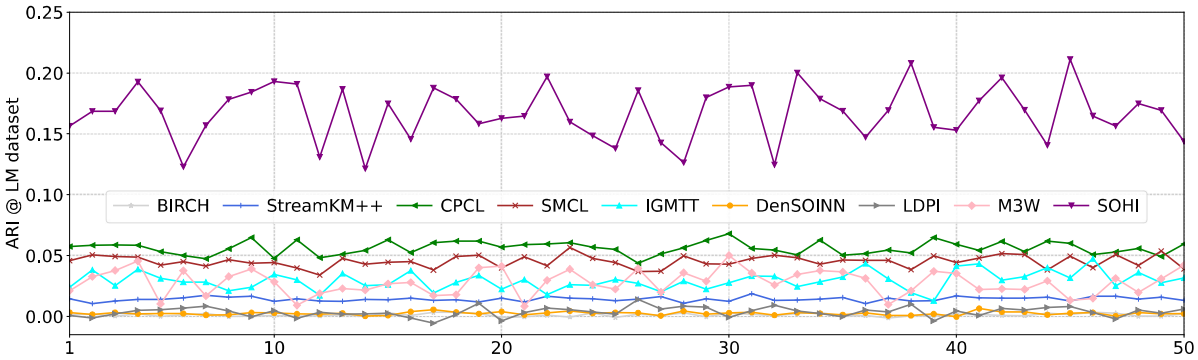


Fig. 4: ARI performance on 50 streaming chunks of LM dataset.

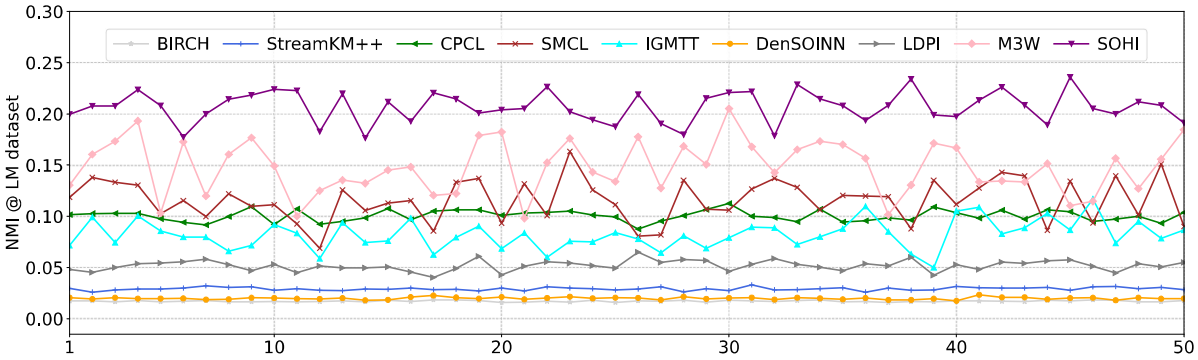


Fig. 5: NMI performance on 50 streaming chunks of LM dataset.

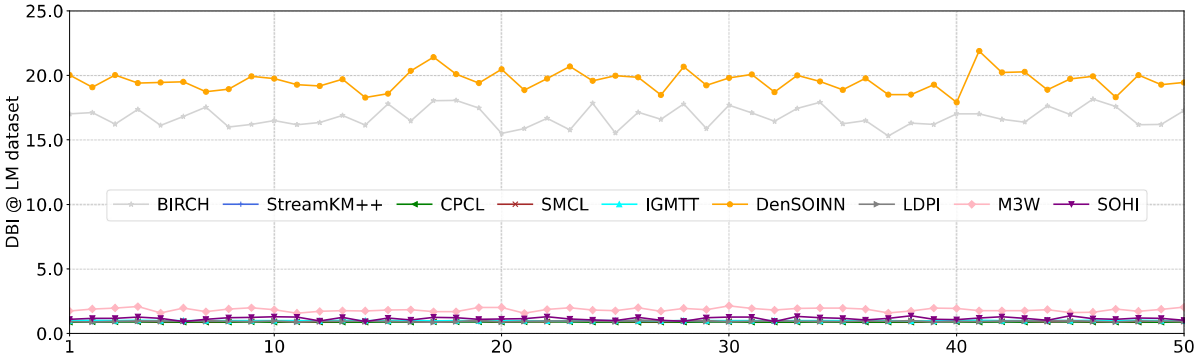


Fig. 6: DBI performance on 50 streaming chunks of LM dataset.

growth is restricted by preventing new neuron addition; 3) Since the effectiveness of HM is in the efficiency perspective, the HM module is treated as a module in this experiment. Because the ablated versions cannot handle streaming data, the ablation study is conducted on each whole static dataset. This is why the ablation study results in Table III are not exactly the same as the chunk-wise results in Table II.

It can be observed that the ablation of any module of SOHI leads to a decrease in its clustering performance, indicating that each module contributes to achieving good clustering performance. More specifically, the MS ablation has a smaller impact, while the SGM ablation results in more significant accuracy differences. This is because even if MS is replaced by a single network, a considerable number of neurons can still represent the data distribution. However, for the SGM module, when it is restricted to grow, the limited number of neurons cannot finely describe the data distribution, thus severely influencing the following subnetwork fusion and cluster merging. In short, SOHI consistently produced the best clustering results across all ablation versions, confirming the effectiveness of the key components.

VI. Conclusion

An accurate and efficient ISDC method called SOHI is proposed. It adaptively trains growing neuron maps named SGM to achieve a topological representation of data distribution with rich local density information. The structure is

proven to be: 1) efficient in adapting to new data distributions by incrementally updating its neurons, 2) effective in describing relatively small cluster distributions, and 3) efficient in providing retrieval information for microcluster merging. In the process of hierarchically merging microclusters to explore imbalanced clusters, the density distribution reflected by SGM is utilized to make a fine judgment on whether two microclusters should be merged. Such a process also guides the selection of the final appropriate number of clusters. To facilitate convincing experimental evaluation, we also propose a streaming data chunk generator that can simulate various extreme situations in real streaming data scenarios. Extensive experiments, including clustering accuracy and efficiency evaluation on streaming and static data, ablation studies, etc., have been conducted. By comparing with the state-of-the-art methods on various datasets, the proposed method is proven to be superior in both accuracy and efficiency for ISDC.

ACKNOWLEDGMENTS

REFERENCES

[1] M. M. Gaber, A. Zaslavsky, and S. Krishnaswamy, "Mining data streams: A review," *SIGMOD Rec*, vol. 34, no. 2, pp. 18–26, 2005.

[2] L. Zhao, Z. Chen, Y. Yang, L. Zou, and Z. J. Wang, "ICFS clustering with multiple representatives for large data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 3, pp. 728–738, 2019.

[3] L. Zhao, Y. Zhang, Y. Ji, A. Zeng, F. Gu, and X. Luo, "Heterogeneous drift learning: Classification of mix-attribute data with concept drifts," in *Proceedings of the IEEE International Conference on Data Science and Advanced Analytics*, IEEE, 2022, pp. 1–10.

[4] A. Zubaroğlu and V. Atalay, "Data stream clustering: A review," *Artificial Intelligence Review*, vol. 54, no. 2, pp. 1201–1236, 2021.

[5] L. Wang, H. Zhu, J. Meng, and W. He, "Incremental local distribution-based clustering using bayesian adaptive resonance theory," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 11, pp. 3496–3504, 2019.

[6] Y. Lu, Y.-m. Cheung, and Y. Y. Tang, "Self-adaptive multiprototype-based competitive learning approach: A k-means-type algorithm for imbalanced data clustering," *IEEE Transactions on Cybernetics*, vol. 51, no. 3, pp. 1598–1612, 2021.

[7] J. A. Hartigan and M. A. Wong, "A k-means clustering algorithm," *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, vol. 28, no. 1, pp. 100–108, 1979.

[8] S. C. Ahalt, A. K. Krishnamurthy, P. Chen, and D. E. Melton, "Competitive learning algorithms for vector quantization," *Neural Networks*, vol. 3, no. 3, pp. 277–290, 1990.

[9] L. Xu, A. Krzyzak, and E. Oja, "Rival penalized competitive learning for clustering analysis, rbf net, and curve detection," *IEEE Transactions on Neural Networks*, vol. 4, no. 4, pp. 636–649, 1993.

[10] A. M. Ikotun, A. E. Ezugwu, L. Abualigah, B. Abuhaija, and J. Heming, "K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data," *Information Sciences*, vol. 622, pp. 178–210, 2023.

[11] J. Liang, L. Bai, C. Dang, and F. Cao, "The K-means-type algorithms versus imbalanced data distributions," *IEEE Transactions on Fuzzy Systems*, vol. 20, no. 4, pp. 728–745, 2012.

[12] Y.-m. Cheung, "On Rival penalization controlled competitive learning for clustering with automatic cluster number selection," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 11, 2005.

[13] B. Xu, F. Shen, and J. Zhao, "A density-based competitive data stream clustering network with self-adaptive distance metric," *Neural Networks*, vol. 110, pp. 141–158, 2019.

[14] H. Jia, Y.-m. Cheung, and J. Liu, "Cooperative and penalized competitive learning with application to kernel-based clustering," *Pattern Recognition*, vol. 47, pp. 3060–3069, 2014.

[15] M. Du, J. Zhao, J. Sun, and Y. Dong, "M3W: Multistep three-way clustering," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 4, pp. 5627–5640, 2022.

[16] R. Liu, H. Wang, and X. Yu, "Shared-nearest-neighbor-based clustering by fast search and find of density peaks," *Information Sciences*, vol. 450, pp. 200–226, 2018.

[17] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, 2014.

[18] J. Xie, H. Gao, W. Xie, X. Liu, and P. W. Grant, "Robust clustering by detecting density peaks and assigning points based on fuzzy weighted k-nearest neighbors," *Information Sciences*, vol. 354, pp. 19–40, 2016.

[19] W. Tong, Y. Wang, and D. Liu, "An adaptive clustering algorithm based on local-density peaks for imbalanced data without parameters," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 4, pp. 3419–3432, 2023.

[20] M. R. Ackermann, M. Märtens, C. Raupach, K. Swierkot, C. Lammersen, and C. Sohler, "Streamkm++ a clustering algorithm for data streams," *Journal of Experimental Algorithmics*, vol. 17, pp. 2–1, 2012.

[21] D. Arthur and S. Vassilvitskii, "K-means++: The advantages of careful seeding," in *Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms*, 2007, pp. 1027–1035.

[22] D. Amagata, "Scalable and accurate density-peaks clustering on fully dynamic data," in *Proceedings of the IEEE International Conference on Big Data*, 2022, pp. 445–454.

[23] Y.-m. Cheung and Y. Zhang, "Fast and accurate hierarchical clustering based on growing multilayer topology training," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, pp. 876–890, 2019.

[24] T. Zhang, R. Ramakrishnan, and M. Livny, "Birch: A new data clustering algorithm and its applications," *Data Mining and Knowledge Discovery*, vol. 1, pp. 141–182, 1997.

[25] C. C. Aggarwal, P. S. Yu, J. Han, and J. Wang, "A framework for clustering evolving data streams," in *Proceedings of the VLDB Conference*, 2003, pp. 81–92.

[26] F. Cao, M. Estert, W. Qian, and A. Zhou, "Density-based clustering over an evolving data stream with noise," in *Proceedings of the SIAM International Conference on Data Mining*, 2006, pp. 328–339.

[27] Y. Chen and L. Tu, "Density-based clustering for real-time stream data," in *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2007, pp. 133–142.

[28] M. Peng, Y. Wu, Y. Lu, M. Li, Y. Zhang, and Y.-m. Cheung, "Weighted density for the win: Accurate subspace density clustering," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, IEEE, 2025, pp. 1–5.

[29] P. P. Rodrigues, J. Gama, and J. Pedroso, "Hierarchical clustering of time-series data streams," *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, no. 5, pp. 615–627, 2008.

[30] J. Chen, S. Yang, C. Fahy, Z. Wang, Y. Guo, and Y. Chen, "Online sparse representation clustering for evolving data streams," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 36, no. 1, pp. 525–539, 2025.

[31] K.-s. Zhang, L. Zhong, L. Tian, X.-y. Zhang, and L. Li, "Dbiecm-an evolving clustering method for streaming data clustering," *Amse Journals-Amse Iieta*, vol. 60, no. 1, pp. 239–254, 2017.

[32] M. Moshtaghi, J. C. Bezdek, S. M. Erfani, C. Leckie, and J. Bailey, "Online cluster validity indices for performance monitoring of streaming data clustering," *International Journal of Intelligent Systems*, vol. 34, no. 4, pp. 541–563, 2019.

[33] L. Huang, C.-D. Wang, H.-Y. Chao, and P. S. Yu, "Mvstream: Multi-view data stream clustering," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 9, pp. 3482–3496, 2020.

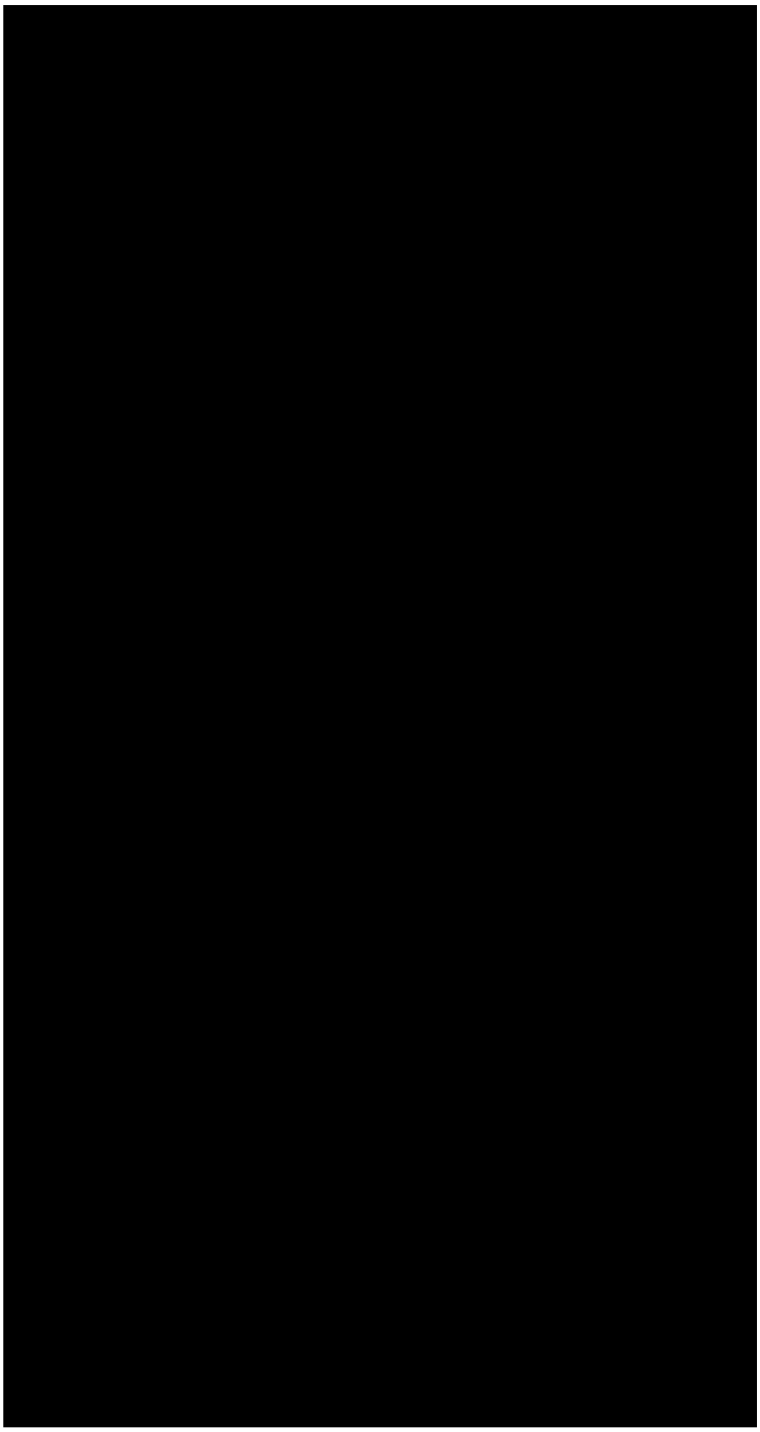
[34] J. Zhang, H. Tao, and C. Hou, "Imbalanced clustering with theoretical learning bounds," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 9, pp. 9598–9612, 2023.

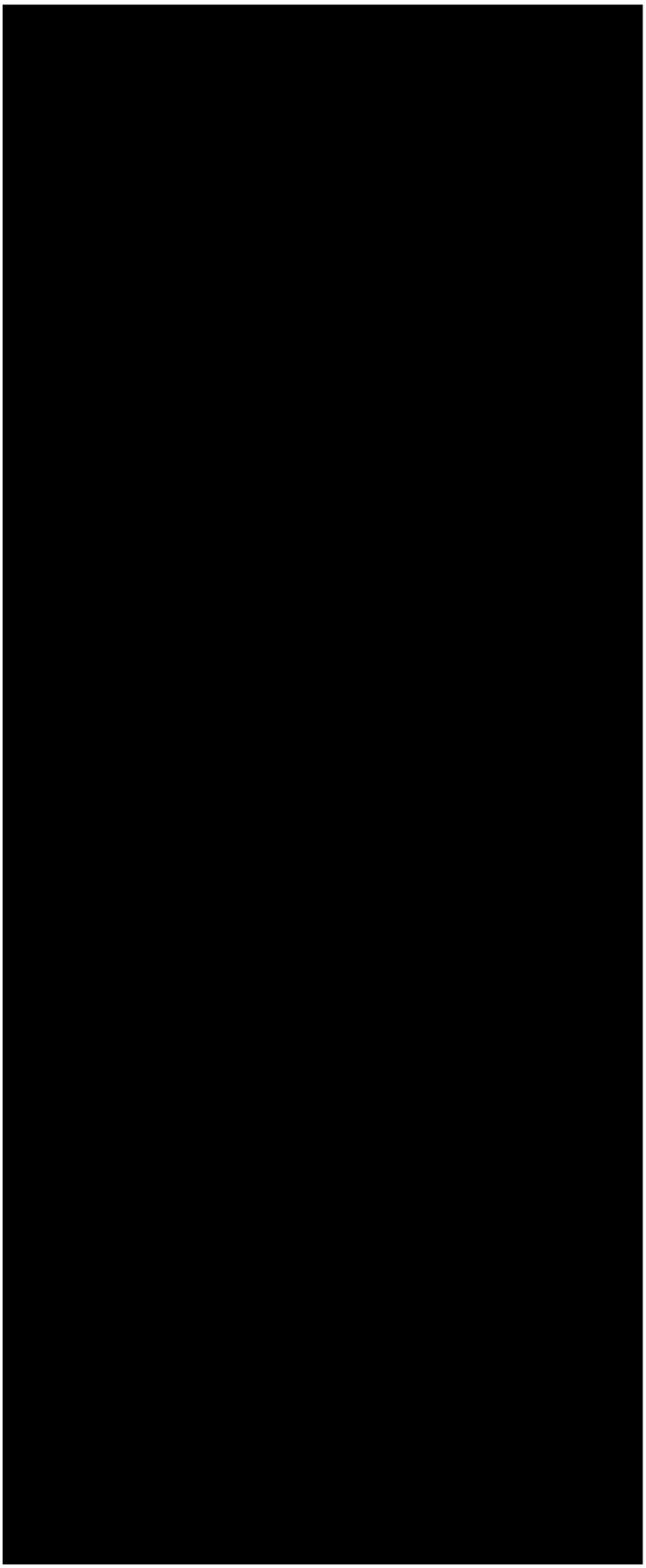
[35] M. Liu, X. Jiang, and A. C. Kot, "A multi-prototype clustering algorithm," *Pattern Recognition*, vol. 42, no. 5, pp. 689–698, 2009.

[36] Y. Zhang, R. Zou, Y. Zhang, Y. Zhang, Y.-m. Cheung, and K. Li, "Adaptive micro partition and hierarchical merging for accurate mixed data clustering," *Complex & Intelligent Systems*, vol. 11, no. 1, pp. 1–14, 2025.

[37] S. Cai, Y. Zhang, X. Luo, Y.-m. Cheung, H. Jia, and P. Liu, "Robust categorical data clustering guided by multi-granular competitive

learning," in *Proceedings of the IEEE International Conference on Distributed Computing Systems*, 2024, pp. 288–299.
[38] Y. Zhang, X. Luo, Q. Chen, R. Zou, Y. Zhang, and Y.-m. Cheung, "Towards unbiased minimal cluster analysis of categorical-and-numerical attribute data," in *Proceedings of the International Conference on Pattern Recognition*, Springer, 2024, pp. 254–269.
[39] Z. A. Huang, Y. Sang, Y. Sun, and J. Lv, "Neural network with a preference sampling paradigm for imbalanced data classification," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 7, pp. 9252–9266, 2024.
[40] D. Li, S. Zhou, T. Zeng, and R. H. Chan, "Multi-prototypes convex merging based k-means clustering algorithm," *IEEE Transactions on Knowledge and Data Engineering*, vol. 36, no. 11, pp. 6653–6666, 2023.
[41] J. Chen, Y. Ji, R. Zou, Y. Zhang, and Y.-m. Cheung, "Qgrl: Quaternion graph representation learning for heterogeneous feature data clustering," in *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2024, pp. 297–306.
[42] Y. Zhang, M. Zhao, Y. Chen, Y. Lu, and Y.-m. Cheung, "Learning unified distance metric for heterogeneous attribute data clustering," *Expert Systems with Applications*, p. 126 738, 2025.
[43] S. Feng, M. Zhao, Z. Huang, Y. Ji, Y. Zhang, and Y.-m. Cheung, "Robust qualitative data clustering via learnable multi-metric space fusion," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, IEEE, 2025, pp. 1–5.
[44] M. Ahmed, "Data summarization: A survey," *Knowledge and Information Systems*, vol. 58, pp. 249–273, 2018.
[45] Z. R. Hesabi, Z. Tari, A. Goscinski, A. Fahad, I. Khalil, and C. Queiroz, "Data summarization techniques for big data—a survey," *Handbook on Data Centers*, pp. 1109–1152, 2015.
[46] T. Kohonen, "The self-organizing map," *Neurocomputing*, vol. 21, pp. 1–6, 1998.
[47] J. Vesanto and E. Alhoniemi, "Clustering of the self-organizing map," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 11, no. 3, pp. 586–600, 2000.
[48] X. Luo, Y. Zhang, Y. Ji, P. Liu, and T. Xiao, "Efficient topology-driven clustering for imbalanced streaming biomedical data analysis," in *Proceedings of the IEEE International Conference on Bioinformatics and Biomedicine*, IEEE, 2024, pp. 2262–2267.
[49] M. M. Breunig, H.-P. Kriegel, P. Kröger, and J. Sander, "Data bubbles: Quality preserving performance boosting for hierarchical clustering," in *Proceedings of the ACM SIGMOD Conference on Management of data*, 2001, pp. 79–90.
[50] M. M. Breunig, H.-P. Kriegel, and J. Sander, "Fast hierarchical clustering based on compressed data and optics," *Principles of Data Mining and Knowledge Discovery*, pp. 232–242, 2002.
[51] S. Nassar, J. Sander, and C. Cheng, "Incremental and effective data summarization for dynamic hierarchical clustering," in *Proceedings of the ACM SIGMOD Conference on Management of Data*, 2004, pp. 467–478.
[52] Y. Zhang, Y.-m. Cheung, and Y. Liu, "Quality preserved data summarization for fast hierarchical clustering," in *Proceedings of the International Joint Conference on Neural Networks*, 2016, pp. 4139–4146.
[53] J. A. F. Costa and M. L. de Andrade Netto, "Clustering of complex shaped data sets via kohonen maps and mathematical morphology," in *Proceedings of SPIE - The International Society for Optical Engineering*, vol. 4384, 2001, pp. 16–27.
[54] D. Olszewski, "Asymmetric k-means clustering of the asymmetric self-organizing map," *Neural Processing Letters*, vol. 43, no. 1, pp. 231–253, 2016.
[55] A. A. Hameed, B. Karlik, M. S. Salman, and G. Eleyan, "Robust adaptive learning approach to self-organizing maps," *Knowledge-Based Systems*, vol. 171, pp. 25–36, 2019.
[56] Y. Zhang, M. Simsek, and B. Kantarci, "Empowering self-organized feature maps for ai-enabled modeling of fake task submissions to mobile crowdsensing platforms," *IEEE Internet of Things Journal*, vol. 8, no. 3, pp. 1334–1346, 2020.
[57] B. Fritzke, "Growing self-organizing networks—history, status quo, and perspectives," *Kohonen Maps*, pp. 131–144, 1999.
[58] A. Rauber, D. Merkl, and M. Dittenbach, "The growing hierarchical self-organizing map: Exploratory analysis of high-dimensional data," *IEEE Transactions on Neural Networks*, vol. 13, no. 6, pp. 1331–1341, 2002.
[59] D. N. Coelho and G. A. Barreto, "A sparse online approach for streaming data classification via prototype-based kernel models," *Neural Processing Letters*, vol. 54, pp. 1679–1706, 2022.
[60] S. Li, F. Liu, L. Jiao, P. Chen, and L. Li, "Self-supervised self-organizing clustering network: A novel unsupervised representation learning method," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–15, 2022.
[61] R. Zhu, Z. Wang, Z. Ma, G. Wang, and J.-H. Xue, "Lrid: A new metric of multi-class imbalance degree based on likelihood-ratio test," *Pattern Recognition Letters*, vol. 116, pp. 36–42, 2018.
[62] B. Fritzke, "Growing cell structures—a self-organizing network for unsupervised and supervised learning," *Neural Networks*, vol. 7, no. 9, pp. 1441–1460, 1994.
[63] R. Bridson, "Fast poisson disk sampling in arbitrary dimensions," *SIGGRAPH sketches*, vol. 10, no. 1, 2007.
[64] M. Kelly, R. Longjohn, and K. Nottingham, *The uci machine learning repository*, https://archive.ics.uci.edu.





Quaternion Cross-Modality Spatial Learning for Multi-Modal Medical Image Segmentation

Junyang Chen, [redacted]

***Abstract*—Recently, the Deep Neural Networks (DNNs) have had a large impact on imaging process including medical image segmentation, and the real-valued convolution of DNN has been extensively utilized in multi-modal medical image segmentation to accurately segment lesions via learning data information. However, the weighted summation operation in such convolution limits the ability to maintain spatial dependence that is crucial for identifying different lesion distributions. In this paper, we propose a novel Quaternion Cross-modality Spatial Learning (Q-CSL) which explores the spatial information while considering the linkage between multi-modal images. Specifically, we introduce to quaternion to represent data and coordinates that contain spatial information. Additionally, we propose Quaternion Spatial-association Convolution to learn the spatial information. Subsequently, the proposed De-level Quaternion Cross-modality Fusion (De-QCF) module excavates inner space features and fuses cross-modality spatial dependency. Our experimental results demonstrate that our approach compared to the competitive methods perform well with only 0.01061 M parameters and 9.95G FLOPs.**

***Index Terms*—Multi-modal medical image, Quaternion, Spatial dependency, Cross-modality.**

Manuscript received 17 April 2023; revised 15 July 2023, 8 September 2023, 11 December 2023, and 17 December 2023; accepted 19 December 2023. Date of publication 25 December 2023; date of current version 7 March 2024. This work was supported in part by the Key Areas Research and Development Program of Guangzhou under Grant 2023B01J0029, in part by Science and Technology Research in Key Areas in Foshan under Grant 2020001006832, in part by the Guangdong Provincial Key Laboratory of Cyber-Physical System under Grant 2020B1212060069, in part by the National Science Foundation of China under Grant U21A20478, in part by the Guangdong Basic, Applied Basic Research Foundation under Grant 2023A1515012534, and in part by the Science and Technology Program of Guangzhou [redacted]

Junyang Chen [redacted]

I. Introduction

In THE last decade, there have been many excellent works emerging in the medical image segmentation area. Representative work is U-Net [1], which leverages an encoder-decoder design to effectively capture features in single medical image sequences. As this field has evolved, there has been a growing collection of multi-modal 3D medical images, such as those available in the BraTS dataset. These multi-modal images offer a wealth of information for lesion detection, surpassing the information of individual image sequences. For instance, T2 and Flair are suitable for detecting tumors with perineural edema, while T1 and T1ce are better for detecting tumor core without peritumoral edema in gliomas. Recently, U-Net-like networks [2], [3], [4] use the real-valued 3D convolution to directly handle all of the modality inputs, making a lot of success.

Specifically, V-Net [2] utilizes real-valued 3D convolution to learn lesion under a U-Net architecture. Meanwhile, SegResNet [3] also extensively incorporates real-valued 3D convolution with VAE part to construct image. These approaches have demonstrated impressive performance in medical segmentation tasks. However, it is essential to note that these methods overlook the crucial Euclidean spatial information of voxels in three-dimensional space. The characteristics and distribution of lesions within an organ exhibit a strong correlation with the Euclidean spatial information. For example, the different types of tumors may spatially distribute in different regions, reflecting significant spatial disparities (see Fig. 1). Spatial information plays a pivotal role in distinguishing between various brain tumors. Consequently, it becomes imperative to capture this spatial information in order to unveil the spatial distribution patterns of lesions. Nevertheless, the aforementioned methods that rely heavily on real-valued 3D convolutions may fail to capture the spatial regularity of the lesion. This limitation arises from the intrinsic nature of real-valued 3D convolutions, which struggles to preserve the dependence on the Euclidean spatial aspects of the data. This convolution intends to mix voxel values in respective field of kernel, as illustrated in Fig. 1. Moreover, the convolution kernel is one-to-one with the data, which increases the parameter and computational complexity. Recently, a burgeoning area of research [5] has extended data from

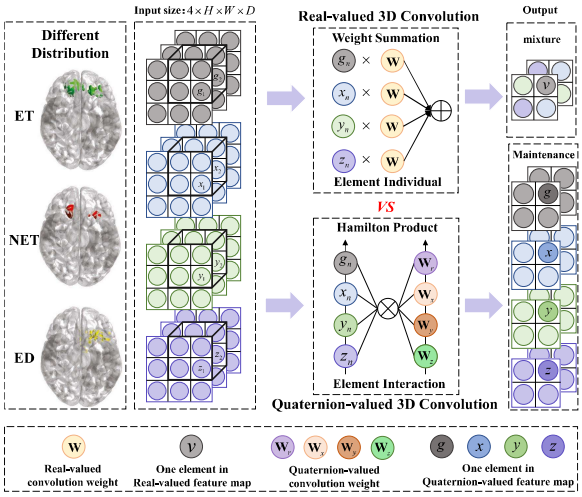


Fig. 1. Illustration of different tumor distributions and convolution. In the first, we show three different tumor types in brain. In gliomas, the lesion regions consist of Necrotic Tumor (NET), Enhancing Tumor (ET), as well as Peritumoral Edema (ED), where the tumors have spatially distinct distributions that reflect spatial information. Furthermore, we compare different convolution effectiveness within the same input that contains grey-valued and 3D coordinates. We can observe that the real-valued 3D convolution handle the input via weight summation and the output is a mixed feature maps, which loss the spatial dependence. In contrast, the quaternion-valued 3D convolution utilizes Hamilton product to interactively learn the inner relations of elements with respective weight, which maintain the spatial dependence well in the final output. In summary, the quaternion-valued 3D convolution maintain the spatial dependence better than real-valued 3D convolution.

real-valued field to hypercomplex field. The quaternion in hypercomplex field can represent four-dimensional data perfectly [5]. Hence, the medical image sequences can be represented by the quaternion. While the real part of the quaternion represents the grey values, the imaginary part of quaternion corresponds exactly to the Euclidean coordinates. Therefore, by avoiding mixing of regional information caused by weighted summation, the quaternion-valued convolution powerfully maintain the Euclidean information (see Fig. 1). Despite the quaternion-valued convolution learn the spatial information in one modal, it becomes imperative to devise fusion methods to also learn the spatial information from other modalities.

As we all know, multi-modal medical images contain many different features owing to the distinct imaging principles employed. Thus, an efficient cross-modal fusion method can facilitate extracting the latent feature embeddings [6], [7], [8], [9]. To use multiple modalities to synthesize target-modality images, the Hi-Net [8] fuses the cross-modal feature and integrates multi-modal multi-level representations. Then, the MSTN [7] hopes to handle the modality-discrepancy in both feature and classifier level by a collaborative ensemble learning scheme, which performances well in the cross-modality Re-ID. Moreover, to approximate the positive concentration and negative separation in category-wise supervised learning, Ye et al. [6] proposed an instance-wise unsupervised softmax embedding and train with a Siamese network which project to another feature modal by proposed data augmentation. In summary, these works have demonstrated the effectiveness of multi-modal fusion.

From the preceding discussions, we propose a novel Quaternion Cross-modality Spatial Learning (Q-CSL) in quaternion field. Our method begins by representing voxels in quaternion, effectively establishing a connection between spatial coordinates and gray-scale values. We subsequently employ quaternion convolution to extract lesion features, offering a unique Euclidean spatial perspective. To be more precise, the Quaternion Spatial-association Convolution (QSConv) which employs Hamilton product to facilitate interaction between Euclidean coordinates and gray-scale values. Notably, the quaternion convolution brings about a significant reduction in model parameters, courtesy of its inherent weight-sharing strategy. Moreover, the Quaternion Encoder Block (QEB), comprising QSConv, serves as a lightweight building block. Subsequently, after extracting spatial features of each modality, we propose the De-level Quaternion Cross-modality Fusion (De-QCF) module, which compresses multi-modal features in the decode stage (De-level). Only features contributing to the structural identity are forwarded to the decoder in the subsequent phase. In De-QCF, we introduce Intra-modal Hamilton-attention to establish local and global dependencies in quaternion. Here, we use the Hamilton product instead of the dot product in self-attention, allowing for interaction within the triad group (Q, K, V) and the sharing of multiple confidence scores in quaternion field. Following the intra-modal Hamilton-attention, we propose Cross-modality Complementary Fusion (CCF) mechanism to complement different spatial features from multi-modal.

The contributions in this work are concluded as follows:

- We propose a novel Quaternion Cross-modality Spatial Learning (Q-CSL), driven by quaternion inputs and featuring the De-QCF module. Q-CSL explores spatial distribution information and learns multi-modal spatial structure features.
- We introduce quaternion to represent voxels and propose Quaternion Spatial-association Convolution (QSConv) and transpose convolution (QSTConv) components. QSConv and QSTConv are integrated into lightweight encoder and decoder blocks that mine the intrinsic spatial information.
- We propose a De-level Quaternion Cross-modality Fusion (De-QCF) Module in decoder stage. After exploring intra-modal local-global spatial information by Hamilton-attention, the Cross-modality Complementary Fusion (CCF) Mechanism combines individual attention maps to complement spatial features.
- Our Q-CSL compared with recent methods performs well with mere 0.01061 M parameters and 9.95 G FLOPs on BraTS 2020 and 2021 datasets.

II. Related Works

1) Medical Image Segmentation Network: Recently, the deep learning technology is popular in different areas [10], [11], [12], e.g., semantic categorization and imaging segmentation. In particular, medical image segmentation is a popular and challenging task in recent years, which spawn many successful convolution networks (CNNs) [1], [2], [3], [4], [13], [14], [15].

They utilize non-linear extraction capability of real-valued 3D convolution to learn inner feature of medical image sequence. For instance, Milletari et al. [2] proposed a method that imitates the U-Net architecture and utilizes 3D convolution to learn three dimension data. Moreover, in order to regularise the shared encoder and impose additional constraints on the decoder layer, Myronenko et al. [3] built an encoder-decoder architecture, also adding Variable Autoencoder (VAE) branches to reconstruct the input image itself. Although the CNNs perform well in different medical image segmentation task, they ignore the local and global feature. Hence, many works [16], [17], [18], [19], [20] focus on the transformer-based method which learn local and global image features recently. The TransBTS [16] utilizes encoder-decoder architecture to sample the feature maps and extract 3D global spatial features by the transformer layer. Zhou et al. [20] proposed skip attention mechanism and interleave convolution and self-attention operations to learn local and global volume representations. The VTNet [18] proposes to fuse different attention feature maps in transformer encoder-decoder, and capture fine details for boundary refinement via the cross attention. However, the spatial information learned via real-valued convolution or easy attention method is lacking for segmentation. The real-valued 3D convolution tends to identify the types of tumor and ignore the spatial information in segmentation. Therefore, we propose to introduce quaternion to represent images and construct quaternion-valued convolution to capture spatial information.

2) Quaternion Neural Networks: In fact, many recent works extend the number field of neural networks from real-valued to quaternion-valued. Quaternion neural networks have shown great potential in a variety of areas, e.g., color images recognition classification [21], 3D acoustic signal processing [22], and speech recognition [23], even human pose estimation [24]. Quaternion instead of real-valued is utilized to represent data inputs in these works. With orthogonality of quaternion in hyper-complex domains, quaternion perfects to fit three and four dimensional inputs. The Hamilton product is essential for quaternion effectiveness, enabling unique interactions between two quaternion. However, we note that while quaternion has a wide range of applications in fields such as vision, audio and speech recognition, little attention has been paid to 3D medical image segmentation. 3D medical images, especially brain MRI, reflect the distribution of different tumors. Thus, we introduce quaternion to represent MRI images and learn the distribution features by Quaternion Spatial Convolution. Also, we further introduce quaternion into multi-modal fusion.

3) Multi-Modal Fusion: Academic researchers now have accessed to various types of organ images, such as MRI scans of the heart [25], MRIs of the brain [26], and CT scans of the lungs [27], which contain diverse features across different modalities. Many previous works in multi-modal segmentation focus on early-level fusion [28], [29], late-level fusion [30], [31] and layer-level fusion [32] etc. Initially, the early-level fusion utilizes low-order feature to fuse modality, lacking abstract information of high-order feature. Thus, late-level fusion [30] was proposed to learn multi-modal high-order features. However, late-level strategy overlook intra-modal interaction in early stage and lack superficial features. Consequently, Dolz et al. proposed [32] a layer fusion strategy. Each modality own one pathway, and semantic information is interconnected through a dense network from low to high order, which helps with inner learning. However, the complex connection of feature maps leads to redundancy in modality fusion. Additionally, some researches [33], [34], [35] point out that the method should alleviate the multiple level feature gaps and fuse modalities via convenient method. Ye et al. [35] indicated that there are many noises between different modal and they propose HAT to enhance inter-modal homogeneous feature. To segment different levels of labels, Liang et al. [33] proposed a tree energy loss to utilize low-level and high-level structural relation among pixels which make a great improvement. TriSeNet [34] extracts high-resolution spatial features, high-level semantic features, and detailed boundary features, then fuses them to segment the foreground more accurately.

4) Self-Attention Mechanism: Recently, attention and self-attention mechanisms are prevalent in visual tasks. These works [36], [37], [38] consider channels and spatial attention, which can consider relating local to global and excavate global highlight. These mechanisms address limitation of convolution receptive field. Thus, some works [39], [40], [41], [42], [43] start trying to apply attention mechanism in medical image segmentation. For example, the Inf-Net [43] considers to mine the relationship between COVID-19 infection areas and boundary cues by recurrent reverse attention modules and explicit edge-attention guidance. Ashish et al. [41] propose to utilized guided self-attention mechanisms to capture richer contextual dependencies, which overcomes the multiple extraction of the same low-level features. In 3D medical images, the fusion of contextual knowledge is important to improve segmentation accuracy. Hence, we propose corresponding fusion mechanism to learn features.

III. Method

A. Overall

We propose a novel framework called Quaternion Cross-modality Spatial Learning (Q-CSL) (Show in Fig. 2). To begin with, we introduce a new form of representation for 3D medical images, called quaternion voxel representation. Then, we introduce Quaternion Spatial-association Convolution to learn spatial information. In decoder stage, we propose the De-level Quaternion Cross-modality Fusion (De-QCF) module for multi-modal fusion. The De-QCF fuses multi-modal information from complementary features of different modalities via the intra-modal Hamilton-attention and cross-modality complementary fusion (CCF) mechanism.

B. Quaternion Voxel Spatial-Association Learning

1) Quaternion Algebra: We have been inspired by the works [44], [22] which demonstrate that quaternion is a superior way to represent 4D information. For the four-dimension, the quaternion inherits the advantage of representing orthogonal

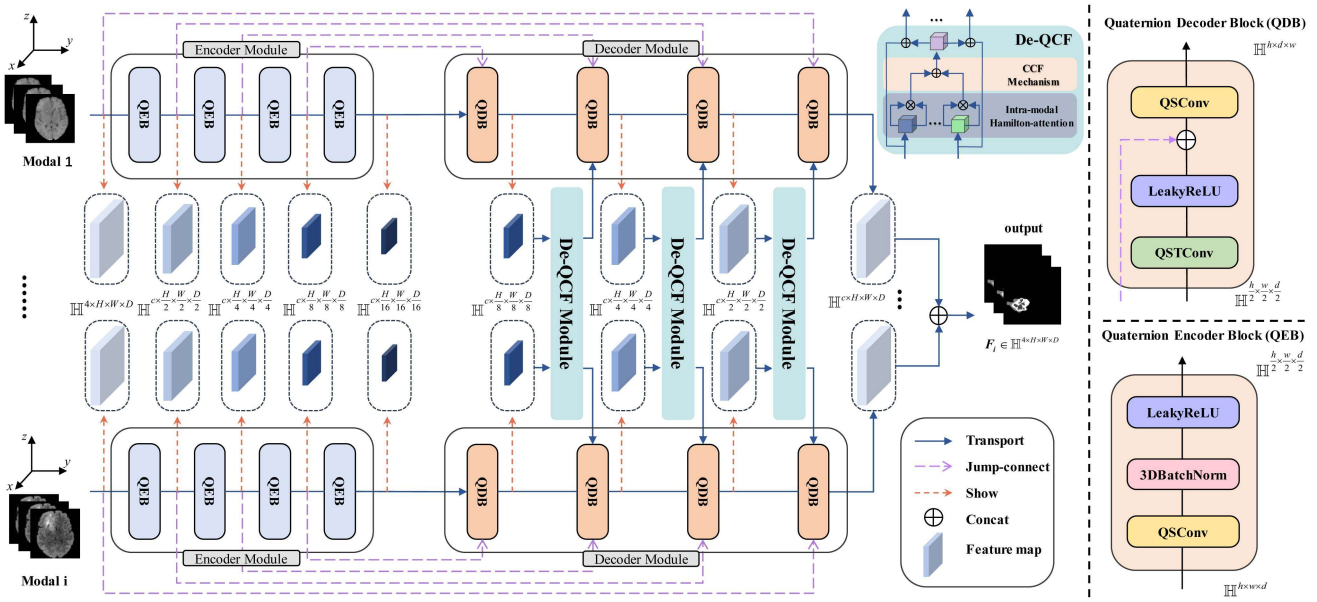


Fig. 2. Overview of the proposed Quaternion Cross-modality Spatial Learning (Q-CSL). The Q-CSL contains De-level Quaternion Cross-modality Fusion (De-QCF) Module. The output feature maps from the Quaternion Encoder Block (QEB) and Quaternion Decoder Block (QDB) are passed into the next layers. In decode stage, each decoded feature maps are transferred to De-QCF module that utilizes Intra-modal Hamilton-attention to excavate internal spatial-association relationship of each modal and fuses different modalities via Cross-modality Complementary Fusion (CCF) Mechanism. Finally, the model concatenates each feature maps and outputs the final labels.

world which adds the connection of different part. We introduce the common quaternion algebra.

Quaternion: The Quaternion in $\mathbb{H}$ is a part of the hyper-complex number. A quaternion $Q \in \mathbb{H}$ is formulated as

$$Q = r + x\mathbf{i} + y\mathbf{j} + z\mathbf{k}, \tag{1}$$

where r is the real part, x, y, z are the imaginary parts. In the imaginary part, there exists a relation that is $\mathbf{i^2} = \mathbf{j^2} = \mathbf{k^2} = \mathbf{ijk} = -1$, where i, j, k are orthogonal in quaternion field.

Quaternion Scalar Multiplication quaternion multiply a scalar is defined as

$$\alpha Q = \alpha r + \alpha x\mathbf{i} + \alpha y\mathbf{j} + \alpha z\mathbf{k}. \tag{2}$$

Hamilton product: In quaternion field, Hamilton Product is an irreducible product. The Hamilton product of two quaternions Q_1, Q_2 can be represented as

$$\begin{aligned} Q_1 \otimes Q_2 = {} & (r_1 r_2 - x_1 x_2 - y_1 y_2 - z_1 z_2) \\ & + (r_1 x_2 + x_1 r_2 + y_1 z_2 - z_1 y_2)\mathbf{i} \\ & + (r_1 y_2 - x_1 z_2 + y_1 r_2 + z_1 x_2)\mathbf{j} \\ & + (r_1 z_2 + x_1 y_2 - y_1 x_2 + z_1 r_2)\mathbf{k}. \end{aligned} \tag{3}$$

The Hamilton product can interact two quaternions. In the deep leaning, compared with eight real-valued neurons which just dot product with each other, quaternion-valued neuron crisscrossly learn inner information. In this work, we utilize Hamilton product to propose model components.

2) Quaternion Voxel Representation: Recently, a number of quaternion works focus on color image reconstruction [45], 3D audio [46] and multi-modal audio [47]. Inspired by these successful works on representation, we introduce quaternion to represent voxel-level medical image data.

Original multi-modal images have three dimensions. We amuse the original input of a modality is $\boldsymbol{U}_i$, where i denotes different modals, i.e., T1, T2, T1ce, and Flair. Generally, the input $\boldsymbol{U}_i \in \mathbb{R}^{H \times W \times D}$, where H, W, D denote high, wide, and depth of the data. Since three dimensions contain richer spatial information, medical images that combine coordinate are more conducive to excavate distributions of lesion in the space. Thus, we combine coordinate and gray-scale value as quaternion.

Suppose one position coordinate is $p = [x, y, z]^{\mathrm{T}}$, and each voxel value of image input $\boldsymbol{U}_i$ is the gray-scale value g_i. Therefore, the corresponding input matrices are $\boldsymbol{P} = [\boldsymbol{X}_i, \boldsymbol{Y}_i, \boldsymbol{Z}_i]$ and $\boldsymbol{G}_i$ where $\boldsymbol{G}_i, \boldsymbol{X}_i, \boldsymbol{Y}_i, \boldsymbol{Z}_i \in \mathbb{R}^{H \times W \times D}$. In order to fit the quaternion input dimension, we concatenate inputs $\boldsymbol{U}_i$ and coordinates $\boldsymbol{P}$ as a quaternion input for each modality.

$$\boldsymbol{F}_i = \mathcal{F}_{cat}(\boldsymbol{U}_i, \boldsymbol{P}) = \boldsymbol{G}_i + \boldsymbol{X}_i\mathbf{i} + \boldsymbol{Y}_i\mathbf{j} + \boldsymbol{Z}_i\mathbf{k}, \tag{4}$$

where $F_{cat}(\cdot, \cdot)$ is the concatenate operation and $\boldsymbol{F}_i \in \mathbb{H}^{4 \times H \times W \times D}$. For a spatial explanation, the gray-scale value g_i of all voxels connect a coordinate position p in the space (see in Fig. 3). Then the quaternion inputs are learned by quaternion-valued 3D convolution which can preserve the spatial dependency.

3) Quaternion Spatial-Association Convolution: Due to the voxel-level inputs are represented as quaternion inputs, we introduce quaternion convolution kernel $\mathbf{W}$ to redefine the 3D convolution which calculate the output feature map $\boldsymbol{F}_i^l$ where l is the number of layer. Interacting the kernel and quaternion input features via Hamilton product, the method can learn potential dependence information between coordinates and data inputs. Then, we define the quaternion convolution kernel as

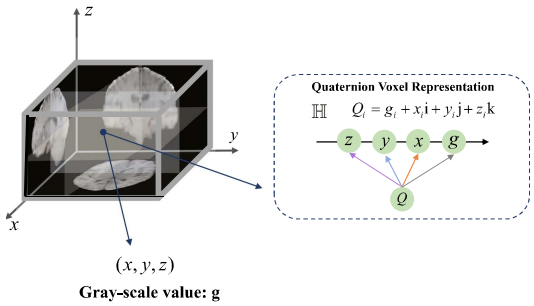


Fig. 3. Quaternion representation of 3D MRI. In the brain MRI, the grey-valued is g which corresponds to the unique coordinates (x, y, z) in the space. Because the quaternion easily matches the 4D inputs, we can represent the MRI in the quaternion field $\mathbb{H}$. Thus, the real-valued inputs are transformed to the quaternion field where one voxel in MRI contains coordinates and grey-valued information.

$\mathbf{W} = \mathbf{W}_r + \mathbf{W}_x\mathbf{i} + \mathbf{W}_y\mathbf{j} + \mathbf{W}_z\mathbf{k}$ and the quaternion bias matrix as $\mathbf{b} = \mathbf{b}_r + \mathbf{b}_x\mathbf{i} + \mathbf{b}_y\mathbf{j} + \mathbf{b}_z\mathbf{k}$. The quaternion convolution kernel $\mathbf{W}$ can be expressed in matrix form as:

$$\mathbf{W} = \begin{bmatrix} \mathbf{W}_r & -\mathbf{W}_x & -\mathbf{W}_y & -\mathbf{W}_z \\ \mathbf{W}_x & \mathbf{W}_r & -\mathbf{W}_z & \mathbf{W}_y \\ \mathbf{W}_y & \mathbf{W}_z & \mathbf{W}_r & -\mathbf{W}_x \\ \mathbf{W}_z & -\mathbf{W}_y & \mathbf{W}_x & \mathbf{W}_r \end{bmatrix}, \tag{5}$$

where $\mathbf{W} \in \mathbb{H}^{c\times h\times w\times d}$. And the c is output channel, the h, w, d are dimensions of kernel size.

Thus, a quaternion spatial-association convolution operator is formulated as:

$$\mathrm{Conv}(\boldsymbol{F}_i) = \mathbf{W} \otimes \boldsymbol{F}_i. \tag{6}$$

For a more visual understanding of internal interactions, individual elements can be written out as:

$$\begin{aligned} \mathbf{W} \otimes \boldsymbol{F}_i = {} & \mathbf{W}_r\boldsymbol{G}_i - \mathbf{W}_x\boldsymbol{X}_i - \mathbf{W}_y\boldsymbol{Y}_i - \mathbf{W}_z\boldsymbol{Z}_i \\ & + \mathbf{W}_x\boldsymbol{G}_i + \mathbf{W}_r\boldsymbol{X}_i - \mathbf{W}_z\boldsymbol{Y}_i + \mathbf{W}_y\boldsymbol{Z}_i \\ & + \mathbf{W}_y\boldsymbol{G}_i + \mathbf{W}_z\boldsymbol{X}_i + \mathbf{W}_r\boldsymbol{Y}_i - \mathbf{W}_x\boldsymbol{Z}_i \\ & + \mathbf{W}_z\boldsymbol{G}_i - \mathbf{W}_y\boldsymbol{X}_i + \mathbf{W}_x\boldsymbol{Y}_i + \mathbf{W}_r\boldsymbol{Z}_i. \end{aligned} \tag{7}$$

And it can be written in a matrix form:

$$\mathbf{W} \otimes \boldsymbol{F}_i = \begin{bmatrix} \mathbf{W}_r & -\mathbf{W}_x & -\mathbf{W}_y & -\mathbf{W}_z \\ \mathbf{W}_x & \mathbf{W}_r & -\mathbf{W}_z & \mathbf{W}_y \\ \mathbf{W}_y & \mathbf{W}_z & \mathbf{W}_r & -\mathbf{W}_x \\ \mathbf{W}_z & -\mathbf{W}_y & \mathbf{W}_x & \mathbf{W}_r \end{bmatrix} \begin{bmatrix} \boldsymbol{G}_i \\ \boldsymbol{X}_i \\ \boldsymbol{Y}_i \\ \boldsymbol{Z}_i \end{bmatrix}. \tag{8}$$

Via the Hamilton product, the quaternion convolution operation interact the elements in $\mathbf{W}$ and $\boldsymbol{F}_i$. this interaction facilitates learning intrinsic association in elements of $\boldsymbol{F}_i$, which real-valued convolution with simple summation fail. Overall, the process of Quaternion Spatial-association Convolution (QSConv) is represented as:

$$\boldsymbol{S}_i^l = \sum_{n=0}^{c} \mathbf{W}^l \otimes \boldsymbol{F}_i^l + \mathbf{b}^l, \tag{9}$$

where $\boldsymbol{S}_i^l$ denotes output feature map after convolution, l is layer number. In this work, we define after downsampling the feature size will reduce by half, so $\boldsymbol{S}_i^l \in \mathbb{H}^{c\times\frac{H}{2}\times\frac{W}{2}\times\frac{D}{2}}$. The interior of Hamilton product is shown in Fig. 4.

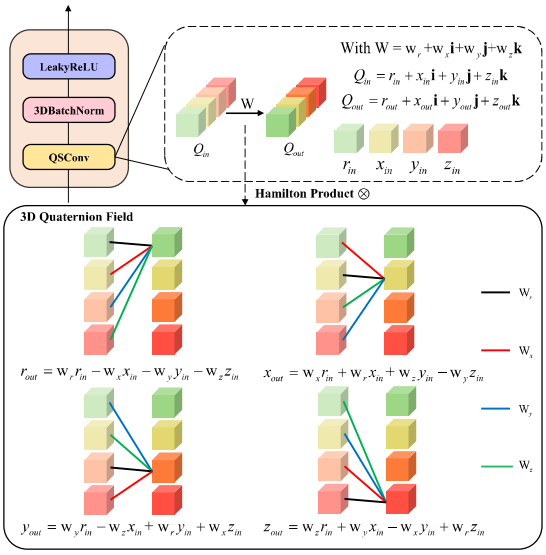


Fig. 4. Overview of Quaternion Spatial-association convolution (QSConv). The Kernel $\boldsymbol{W}$ and feature Q_{in} interact via Hamilton product. In the Hamilton product, the different parts of Q_{in} (i.e., r_{in}, x_{in}, y_{in} and z_{in}) are related with different weights that make the deeper interaction rather than a simple weighted sum.

It is worth discussing the difference of learnable parameters between quaternion-valued and real-valued. When the inputs contain 4 elements, the quaternion-valued method just require 4 learnable parameters to generate the 4 elements outputs. In contrast, the real-valued method needs 16 learnable parameters. The Quaternion-valued method can reduce learnable parameters to one-fourth of the real-valued method. It benefits that not only reducing the freedom degrees of the model which avoids over-fitting [44], but also making it possible to build deeper and wider networks in the same parameters. Especially in 3D medical image, it relieves the burden of building networks in multi-modal learning in a sub-modal way.

4) Quaternion Encoder-Decoder: In previous successful works [1], [2], [3], they show that U-Net-like encoder-decoder architecture is suitable for image segmentation work. It connects low-order and high-order semantic information which help to supplement richer information for segmentation. Therefore, we adopt the skip-connect Encoder-Decoder architecture for each modality. And we utilize the proposed QSConv in stead of the conventional real-valued convolution in the algorithm architecture. In downsampling stage, the encoders are defined as:

$$\boldsymbol{F}_i^{l+1} = \mathrm{EN}\left(\boldsymbol{F}_i^l\right), \tag{10}$$

where $\mathrm{EN}(\cdot)$ denotes a downsampling block of model, and output feature $\boldsymbol{F}_i^{l+1} \in \mathbb{H}^{c\times\frac{h}{2}\times\frac{w}{2}\times\frac{d}{2}}$.

This block that we called Quaternion Encoder Block (QEB) is expressed as:

$$\mathrm{EN}(\boldsymbol{F}_i^l) = \phi(\mathcal{F}_{norm}(\mathrm{Conv}(\boldsymbol{F}_i^l))) = \phi(\mathcal{F}_{norm}(\boldsymbol{S}_i^l)), \tag{11}$$

where $\phi(\cdot)$ is LeakyReLU activation function and $\mathcal{F}_{norm}(\cdot)$ is 3D batch normalization.

In decoder stage, upsampling blocks need to connect with output feature from encoder. Hence, we define Quaternion Spatial-association Transpose Convolution (QSTConv) as:

$$\mathrm{TransConv}(F_i) = \mathbf{W}_T \otimes \boldsymbol{F}_i, \tag{12}$$

where $\mathbf{W}_T \in \mathbb{H}^{c\times h\times w\times d}$ is kernel of quaternion transpose convolution. And the upsampling feature is formulated as:

$$\boldsymbol{H}_i^l = \sum_{n=0}^{c} \mathbf{W}_T^l \otimes \boldsymbol{F}_i^l + \mathbf{b}^l, \tag{13}$$

where c is half of channel of input feature $\boldsymbol{F}_i^l$ and upsampling output feature $\boldsymbol{H}_i^l \in \mathbb{H}^{c\times 2H\times 2W\times 2D}$.

Assuming that Encoder-Decoder layer number is $N(0 < l \leq N)$, and there is a relationship $Del = N - l + 1$ in Decoder stage. According to the above, the blocks that we called Quaternion Decoder block (QDB) can be defined as:

$$\begin{aligned}\boldsymbol{O}_i^{Del+1} &= \mathrm{DE}\left(\boldsymbol{F}_i^{Del}\right) \\ &= \phi\left(\mathrm{Conv}_{1\times1\times1}\left(\boldsymbol{F}_{cat}^{Del}\right)\right),\end{aligned} \tag{14}$$

where $\mathrm{DE}(\cdot)$ donates decoder blocks which utilize $\mathcal{F}_{cat}(\cdot)$ to concatenate upsampling feature $\boldsymbol{H}_i^{Del}$ and downsampling feature $\boldsymbol{S}_i^l$ and transport into $1\times1\times1$ quaternion convolution:

$$\boldsymbol{F}_{cat}^{Del} = \mathcal{F}_{cat}\left(\boldsymbol{H}_i^{Del}, \boldsymbol{S}_i^l\right). \tag{15}$$

This process fuse low-order and high-order features. Additionally, each modal has its own Encoder-Decoder network, which facilitates learning intra-modal potential feature information. In the same time, it is worth noting that multi-modal medical images contain complementary lesion information. Hence, multi-modal fusion strategy driven by spatial association information is important for segmentation accuracy.

C. De-Level Quaternion Cross-Modality Fusion Module

Inspired by the work in [9], we propose to introduce a limited interaction module of modal fusion to condense the feature of different modalities (see in Fig. 6). This module is plugged in the decoder phase that we call De-level, which can acquire different-level semantic information (e.g., the encoded spatial information and the decoded high-level information). Via this method, the De-level mechanism easily fuse most of the important feature information instead of redundant information. Additionally, in order to capture important global-local information within the respective modality and filter out redundant feature information during fusion, we propose De-level Quaternion Cross-modality Fusion (De-QCF) Module in decoder stage.

1) Intra-Modal Hamilton-Attention: According to Fig. 6, upsampling output $\boldsymbol{O}_i^{Del+1}$ is delivered into individual Hamilton-attention. Firstly, $\mathrm{Conv}_{1\times1\times1}(\cdot)$ is used to generate internal attention map $\boldsymbol{Q}_i, \boldsymbol{K}_i, \boldsymbol{V}_i$ where $\boldsymbol{Q}_i, \boldsymbol{K}_i, \boldsymbol{V}_i \in \mathbb{H}^{C\times H\times W\times D}$. The dimension of attention map remain the same as input feature map, which is benefit to maintain contextual relationships ans avoid to break the spatial dependency in the quaternion field. Then, Hamilton-attention is formulated as:

$$\mathcal{A}_i = \mathrm{HaAtt}(\mathbf{W}_Q \otimes \boldsymbol{F}_i, \mathbf{W}_K \otimes \boldsymbol{F}_i, \mathbf{W}_V \otimes \boldsymbol{F}_i), \tag{16}$$

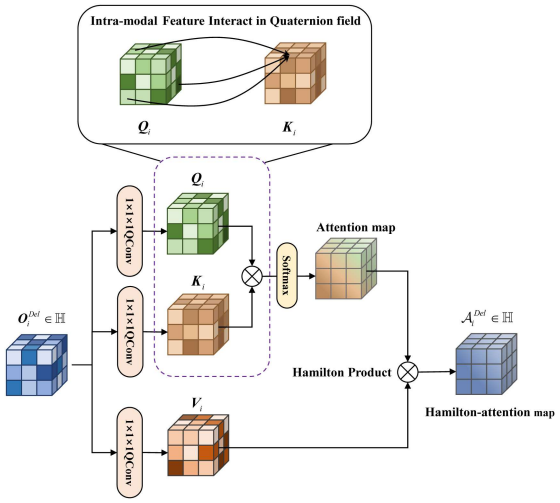


Fig. 5. Illustration of the intra-modal Hamilton-attention of each modal. Instead of dot product, we use Hamilton product to operate queries, keys, and values in quaternion field. It offer more complete information to figure out the local-global relationship.

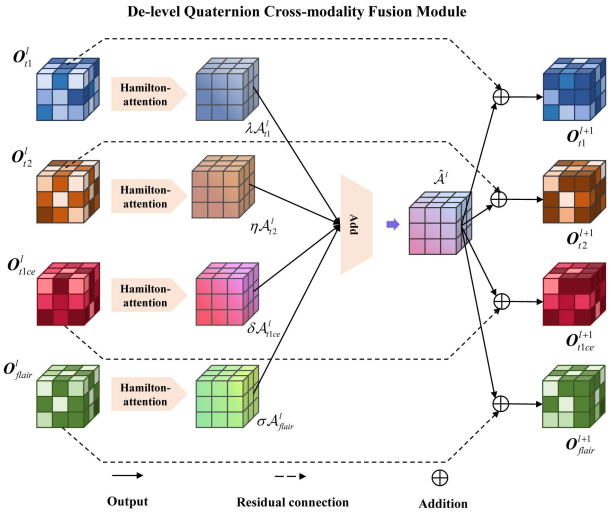


Fig. 6. Overview of De-level Quaternion Cross-modality Fusion (De-QCF) Module. We utilize $\lambda, \eta, \delta, \sigma$ as learnable parameters of each modality. Then, the fused Hamilton-attention maps are residual connected with original QDB outputs. The fusion mechanism can select more contributing modality to supplement the information.

where $\mathrm{HaAtt}(\cdot)$ donates intra-modal Hamilton-attention operation, $\otimes$ is Hamilton product, $\mathcal{A}_i$ is the output attention map, and the learnable parameters $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V \in \mathbb{H}$. The process of intra-modal Hamilton-attention can be expressed as:

$$\mathrm{HaAtt}(\boldsymbol{Q}_i, \boldsymbol{K}_i, \boldsymbol{V}_i) = \mathrm{softmax}\left(\frac{\boldsymbol{Q} \otimes \boldsymbol{K}^T}{\sqrt{d_k}}\right) \otimes \boldsymbol{V}, \tag{17}$$

where $1/\sqrt{d_k}$ denotes the scale and $\mathrm{softmax}(\cdot)$ is softmax function.

According to Equation (17) and Fig. 5, we observe that the attention queries, keys, and values are interacted via Hamilton

product which capture the local-global information in the quaternion field. Intuitively, this non-linear product indeed increases the interaction between local and global and let the model sense more spatial information.

2) Cross-Modal Complementary Fusion Mechanism: Optimizing learnable parameters has always been a central objective of deep learning training. In order to select more contributing modality information, we introduce learnable parameters λ, η, δ, and σ to scale each modality feature. Summing up the Hamilton-attention features of respective modality, cross-modal complementary fusion (CCF) mechanism compress the information to flow through decoder phase. The operation is formulated as:

$$\hat{\mathcal{A}} = \lambda \mathcal{A}_{t1} + \eta \mathcal{A}_{t1ce} + \delta \mathcal{A}_{t2} + \sigma \mathcal{A}_{flair}, \tag{18}$$

where $\hat{\mathcal{A}}$ denotes fusion attention map. Then, $\boldsymbol{O}_i$ and $\hat{\mathcal{A}}$ are added up by a residual connect way. The final output features of each modality are:

$$\mathcal{A}_i = \hat{\mathcal{A}} + \boldsymbol{O}_i. \tag{19}$$

According to equation (13), next quaternion decoder block inputs are expressed as:

$$\boldsymbol{O}_i^{Del+1} = \mathcal{F}_{cat}\left(\mathcal{A}_i, \boldsymbol{S}_i^{l-1}\right). \tag{20}$$

The De-QCF enables not only exploring the global dependence of each modality without learning overly redundant information but also compresses the more important features flow into the next decoder block.

Algorithm 1: Q-CSL.

Input: original input $\boldsymbol{U}_i \in \mathbf{R}^{H\times W\times D}$, 3D coordinates $\boldsymbol{X}_i, \boldsymbol{Y}_i, \boldsymbol{Z}_i \in \mathbf{R}^{H\times W\times D}$, number of modal $mnum$, number of layer N

Output: four kinds of label $\boldsymbol{F}_i \in \mathbf{R}^{4\times H\times W\times D}$

update different modality input $\boldsymbol{F}_i$ by Equation (4)

for $i \leftarrow 1$ **to** $mnum$ **do**
  for $k \leftarrow 1$ **to** $N/2$ **do**
    // Encoder phase
    compute feature map $\boldsymbol{S}_i$ by Equation (9)
    compute next layer inputs $\boldsymbol{F}_i^{l+1}$ by Equation (11)
  end
  for $k \leftarrow N/2+1$ **to** N **do**
    // Decoder phase
    // $Del = N - l + 1$ and l is the number of layer
    compute feature map $\boldsymbol{H}_i^l$ by Equation (13)
    compute output feature $\boldsymbol{O}_i^{Del+1}$ by Equation (14)
    for $j \leftarrow 1$ **to** $mnum$ **do**
      // Fusion phase
      compute each modality $\hat{\mathcal{A}}_i$ by Equation (18)
    end
    compute fusion modal $\hat{\mathcal{A}}$ by Equation (19)
    for $j \leftarrow 1$ **to** $mnum$ **do**
      // update next deconvolution inputs
      update $\boldsymbol{O}_i^{Del+1}$ by Equation (20)
    end
  end
end

IV. Experiment

A. Dataset and Evaluation Metric

1) Brats: In this work, we employ BraTS 2021 and BraTS 2020 [48], [49], [50] to evaluate our model, which contains multi-modal images. The dataset task is aim to segment three lesion regions, i.e., enhancing tumor (ET), tumor core (TC), and whole tumor (WT). The organizers provided four MRI modalities containing T1, T1ce, T2, and FLAIR. In brain gliomas, the lesion regions consist of Necrotic Tumor (NET), Enhancing Tumor (ET), as well as Peritumoral Edema (ED). The size of images is $155 \times 240 \times 240$, and images include a lot of background parts. We test all the method in 1000 training cases and 251 testing cases on the BraTS 2021. And all the methods are verified in 269 training cases and 100 testing cases on the BraTS 2020.

2) ACDC: The ACDC (Automated Cardiac Diagnosis Challenge) dataset [25] consists of cardiac magnetic resonance imaging (MRI) data of 100 patients. Each patient's data includes images of the left ventricle (LV), right ventricle (RV), and myocardium (MYO). The goal of the dataset is to segment these structures accurately. For the experiments, we divide ACDC into three subsets: a training set with 70 samples, a validation set with 10 samples, and a test set with 20 samples.

3) Evaluation Metric: We use Dice coefficient [51] and 95% Hausdorff Distance (HD95). Dice coefficient is defined as

$$\mathrm{Dice}_k(\hat{y}, y) = \frac{2 \cdot ||\hat{y} \cap y||}{(||\hat{y}|| + ||y||)} \times 100\%, \tag{21}$$

where k denotes different tumor types (WT, ET and TC). The dice of lesion regions consist of tumor dice coefficient. A larger Dice coefficient indicates that the predicted labels are closer to ground truth labels, which represents better segmentation accuracy. And 95% Hausdorff Distance demonstrates the extent of the gap between ground truth labels and predicted labels.

For ACDC dataset, we utilize Jaccard index to compute the similarity between segmentation predicted labels and ground truth labels. The Jaccard index is defined as

$$J(A, B) = \frac{|A \cup B|}{|A \cap B|}. \tag{22}$$

B. Implementation Details

1) Experimental Setting: Our Q-CSL is implemented in PyTorch1.6.0 on NVIDIA A5000 GPUs. The loss function is Generalised Dice Loss. We employ Adam optimizer [52] with weight decay of 10^{-5} and amsgrad of TRUE for training. While training, the dataset batch size is 4, learning rate is initialized to 1×10^{-3}, which descent follows $lr = 0.9 \cdot (1 - \frac{epoch}{maxepoch})^8$. And we random crop data from (155,240,240) to (128,128,128), and mirror flipping across the axial, coronal and sagittal planes by a probability of 0.5 randomly, and intensity shifting the value of images for data augmentation.

2) Network Setting: The framework Q-CSL is shown in Fig. 2. For resolve computational burden, we utilize random cropped image inputs. Then, the feature maps are downsampling

TABLE I
COMPARISON WITH EXISTING MODELS ON BRATS 2021 DATASET IN TERMS OF DICE SCORE AND HD95

Model	Dice(%)↑			HD95(mm)↓			Ave.		Params.
	ET	WT	TC	ET	WT	TC	Dice(%)	HD95(mm)	
VNet [2]	82.14	87.89	84.69	5.756	6.608	6.413	84.90	6.259	45.6M
3D U-Net [13]	79.30	85.72	83.49	5.658	12.041	9.000	82.73	8.899	1.78M
Med3D [53]	75.71	83.67	83.46	6.582	10.401	8.110	80.94	8.364	17.41M
HyperDense-Net [32]	73.84	74.86	80.51	6.690	15.26	6.403	76.40	9.451	6.62M
3D-DenseSegNet [54]	81.72	86.06	82.82	5.745	11.445	9.111	83.53	8.767	1.55M
UNet++ [55]	78.13	84.93	82.25	23.88	8.64	17.04	81.77	16.52	26.2M
Attention U-Net [14]	78.97	85.66	83.73	18.47	15.97	11.84	82.78	15.43	103.89M
CoTr [56]	55.70	74.60	74.80	9.45	9.20	10.45	68.36	9.70	41.9M
TransBTS [16]	81.81	87.95	**85.00**	16.79	12.78	11.14	84.94	13.57	33M
UNETR [17]	79.78	**90.10**	83.66	9.72	15.99	10.01	84.51	11.90	102.5M
Ours	**82.21**	89.27	84.69	**5.744**	**5.477**	**6.303**	**85.39**	**5.841**	**0.01061M**

VNet, 3D U-Net, Med3D, HyperDense-Net and 3D-DenseSegNe etc. Are used to compare with our method. The best Dice and HD95 are marked in bold.

TABLE II
COMPARISON WITH EXISTING MODELS ON BRATS 2020 DATASET IN TERMS OF DICE SCORE AND HD95

Model	Dice(%)↑			HD95(mm)↓			Ave.		Params.
	ET	WT	TC	ET	WT	TC	Dice(%)	HD95(mm)	
VNet [2]	73.55	85.56	77.26	2.449	9.486	2.828	78.79	4.921	45.6M
3D U-Net [13]	71.64	84.93	79.35	3.013	10.198	3.068	78.64	5.426	1.78M
Med3D [53]	71.34	81.35	76.32	3.464	13.34	3.00	76.33	6.601	17.41M
HyperDense-Net [32]	65.11	74.24	73.02	5.099	14.17	5.099	70.79	8.122	6.62M
3D-DenseSegNet [54]	75.41	84.38	73.57	3.0	10.44	**2.236**	77.78	5.225	1.55M
Attention U-Net [14]	71.83	85.57	75.96	32.94	11.91	19.43	77.79	21.42	103.89M
TransBTS [16]	76.31	88.78	80.36	29.83	**5.60**	9.77	81.86	15.06	33M
UNETR [17]	71.18	88.30	75.85	34.46	8.18	10.63	78.44	17.75	102.5M
VAE-UNet [57]	67.06	74.69	79.23	3.16	12.52	3.00	73.66	6.23	18.79M
VTU-Net [18]	76.45	**88.73**	80.39	28.99	9.54	14.76	81.86	15.06	10.2M
Ours	**77.05**	88.49	**82.47**	**2.00**	9.403	3.00	**82.67**	**4.801**	**0.01061M**

The best Dice and HD95 are marked in bold.

half by QEB with half size QSConv, the feature map size is $[128^3, 64^3, 32^3, 16^3, 8^3]$. We adopt 4 layers QEB. In decode stage, we recovery the output size as same as input size and transport the outputs into loss function via 4 layers QDB totally. After each QDB, the De-QCF module is used, which does not change the size of feature maps. In order to save computational resource, we only apply this block two times. Finally, the feature maps of each modality are concatenated and then reduced the number of channel to 4 by three layers QSConv.

C. Comparison With State-of-The-Art Methods

1) Brats: BraTS is a challenging dataset for brain tumor segmentation task, which requires segmenting the target tumors from a large background. In Table I, we illustrate the Dice score and 95% Hausdorff Distance to show our method performance in BraTS 2020 and 2021, which results reflect our Q-CSL achieve with competitive parameter size (0.01061 M). For example, our Q-CSL outperforms the HyperDense-Net 8.99% in average Dice score, which proves that quaternion encoder-decoder architecture is better than the high dense connect convolution network in extracting large size feature with many details. In the same construction individual path network show a stronger efficient in multi-modal than one path, Q-CSL is outperform to 3D U-Net 2.66 % in mean Dice score. Q-CSL also shows great performance in HD95 when compared with recent transformer architectures such as UNETR and TransBTS. The mean HD95 of Q-CSL is 5.841 mm, which better predicts the boundaries of the tumor. While the UNETR and CoTr average HD95 is only above 8.0 mm, i.e. 11.90 mm and 9.70 mm. The MAAB from the BraTS competition achieve 78.02, 89.07, and 80.73 in ET, WT, and TC Dice score, respectively. Benefiting from the novel quaternion voxel representation which combines coordinates and gray-scale values, the Q-CSL performs well in HD95 in those models. The HD95 reflects the groundtruth segmented by our method is better fit to the ground truth labels, which certificate learning the map between coordinates and gray-scale values can segment tumor structure better. On the BraTS 2020 dataset (Table II), our method achieves a Dice score of 77.05% for ET, which is a good performance and not much different from other models. In terms of WT and TC, our method achieves a Dice score of 88.49% and 82.47%, respectively. The multi-model method HyperDense-Net achieves a Dice score of 73.84, 74.86, and 80.51 for ET, WT, and TC. The UNet-like architecture UNet++ gets 82.78% for mean Dice score. The transformer-based method TransBTS and UNETR also achieve 81.86 and 78.44, respectively. In terms of HD95, our method has the lowest HD95 value in terms of ET, which is 2.00 mm.

Besides, the transformer-base models achieve a better Dice score, for example the TransBTS get 81.81, 87.95, and 85.00 for ET, WT, and TC Dice score in BraTS 2021. And the popular method nnU-Net is get a better term in Dice (79.89, 91.23, and 85.06) in BraTS 2020, due to the complex data preprocessing. Thus, our Q-CSL needs to focus on more efficient architecture and increase the preprocessing method which can help performance improvement.

Through experiment on BraTS dataset, we confirm the quaternion voxel representation facilitates excavation of tumor

TABLE III
RESULTS OF PARAMETERS, FLOPS, AND MEAN DICE SCORE OF COMPARATIVE MODELS

Model	Params.(M)	FLOPs(G)	Mean Dice(%)
VNet [2]	45.61	789.00	84.90
3D U-Net [13]	1.78	59.03	82.73
Med3D [53]	17.41	3219.53	80.94
HyperDense-Net [32]	6.62	401.74	76.40
3D-DenseSegNet [54]	1.55	1999.37	83.53
TransBTS [16]	33.00	333.00	84.94
UNETR [17]	102.50	193.50	84.51
Ours	**0.01061**	**9.95**	**85.39**

The data is tested in the input size (4; 128; 128; 128) on brats 2021.
The bold values are the best results in the metrics.

TABLE IV
DICE, HD95, AND JACCARD SCORES OF TESTING SAMPLES ON ACDC

	Dice(%)	HD95(mm)	Jaccard
RV	82.07	10.52	0.7204
Myo	83.65	4.426	0.7230
LV	94.19	3.186	0.8920
Mean	86.63	6.047	0.7784

There are three kind of label and the mean scores.

distribution. The De-QCF module is important for multi-modal fusion learning, which utilizes Hamilton product to interact high-level feature in global. After learning local to global inner spatial information, the De-QCF of our method learns about cross-modal Hamilton-attention feature maps to complement internal modal feature maps. Moreover, the parameter size of our method is only 0.01061 M and the FLOPs are 9.95 G, which reflects the powerful segmentation capability of our method in less computational resources. In summary, our method achieves good performance on the BraTS 2020 and 2021 datasets, has a low number of parameters, and is suitable for use in resource limited environments.

2) ACDC: We do an experiment on heart vascular segmentation, which is important for relevant diagnosis. From Table IV, we can observe that the LV has the highest Dice coefficient at 94.19%, while the RV and Myo have Dice coefficients of 82.07% and 83.65%, respectively. The average Dice coefficient is 86.63%. And LV has the smallest HD95 value at 3.186 mm, while the RV and Myo have HD95 values of 10.52 mm and 4.426 mm, respectively. Next, the average Jaccard index is 0.7784, which reflects the intersection-over-union between groundtrue and prediction. In summary, our Q-CSL also can expand to other medical images easily.

3) Relevant Analysis: To better show the segmentation capability of our model, we demonstrate box plot (see in Fig. 8) of the Dice score of every case on BraTS 2021 testing. Through the box plot, we can clearly observe that the three Dice scores of cases segmented by our Q-CSL are concentrated in nearly area, which reflect that Q-CSL have strong generalization ability of segmenting different brain tumor lesions and high stability of segmentation. From an explainable perspective, our method does fuse different modality features to learn the inner spatial properties of brain tumors in BraTS 2021.

To evaluate the computational performance and parameters of the model, we test the parameters and FLOPs of the compared models. The detailed results are shown in Table III and visualization of computational performance is shown in Fig. 9. Our method significantly outperforms other models in terms of parameters and FLOPs in a competitive Dice score on BraTS 2021. This excellent performance is due to the introduction of quaternion, which infiltrates our method and compose many of the operational components. The quaternion model parameters are one-quarter equivalent to real-valued networks. Based on this excellent property, we can construct a deep and sophisticated network for figuring out more information to improve segmentation accuracy.

4) Segmentation Visualization: To more intuitive to understand our proposed work, we show the segmentation testing results in Fig. 7. In the pictures, one row represents the same MRI sequences, which makes it easier to inspect the details in segmentation phrase. Compared to the groundtruth pictures, the results of the novel method Q-CSL are essentially close to it, and the detail structures are depicted nearly, which verify our method indeed learning the inner spatial information in quaternion field. Interesting in eighth column pictures of our method, we figure out that the results of sequences in this disease case are all close to the groundtruth, even in some tiny changes, which reflects that the Q-CSL can findi the relationship between sequences, in other words, it learns in global 3D space.

In Fig. 10, we can observe our Q-CSL can learn different vascular labels in heart. The yellow label is right ventricle (RV), and the green label is myocardium (Myo). The red label indicates the left ventricle. While the label is small, Q-CSL also segments the small ventricles of the heart. In short, this visualisation of experiment reflects our method is suitable to different organ medical images.

D. Ablation Experiments

We demonstrate ablation experiments to check the performance of our main contributions, explain the architecture in which we make weather preserve most of the structural information, and implement inner modal interactions and multi-modal fusion in the De-QCF module. To guarantee a fair comparison, we performed each ablation experiment on the BraTS 2021 dataset under the same experimental setup. We define a backbone consisting of four skip-connect networks for each modality and convolution, which is a normal 3D real-valued convolution. Then, we replace the 3D real-valued convolution with quaternion-valued convolution to inspect the new module. Finally, we insert the De-level Quaternion Cross-modality Fusion Module in the decode stage.

1) Efficientness of QSConv and QSTConv: Quaternion Spatial-association Convolution and Quaternion Spatial-association Transpose Convolution are brand new convolution designs, which is defined for excavating relationship of quaternion inputs in quaternion field. These components are always used in encoder-decoder block, which can maintain the dependency between coordinates and gray-scale values to learn spatial information. Therefore, as shown in Table V , we further improve the performance by 0.42% in mean Dice score. These results show that while the convolutional computational field transforms from real-valued to quaternion-valued with quaternion inputs, the module can interact with elements in the quaternion representation to improve performance.

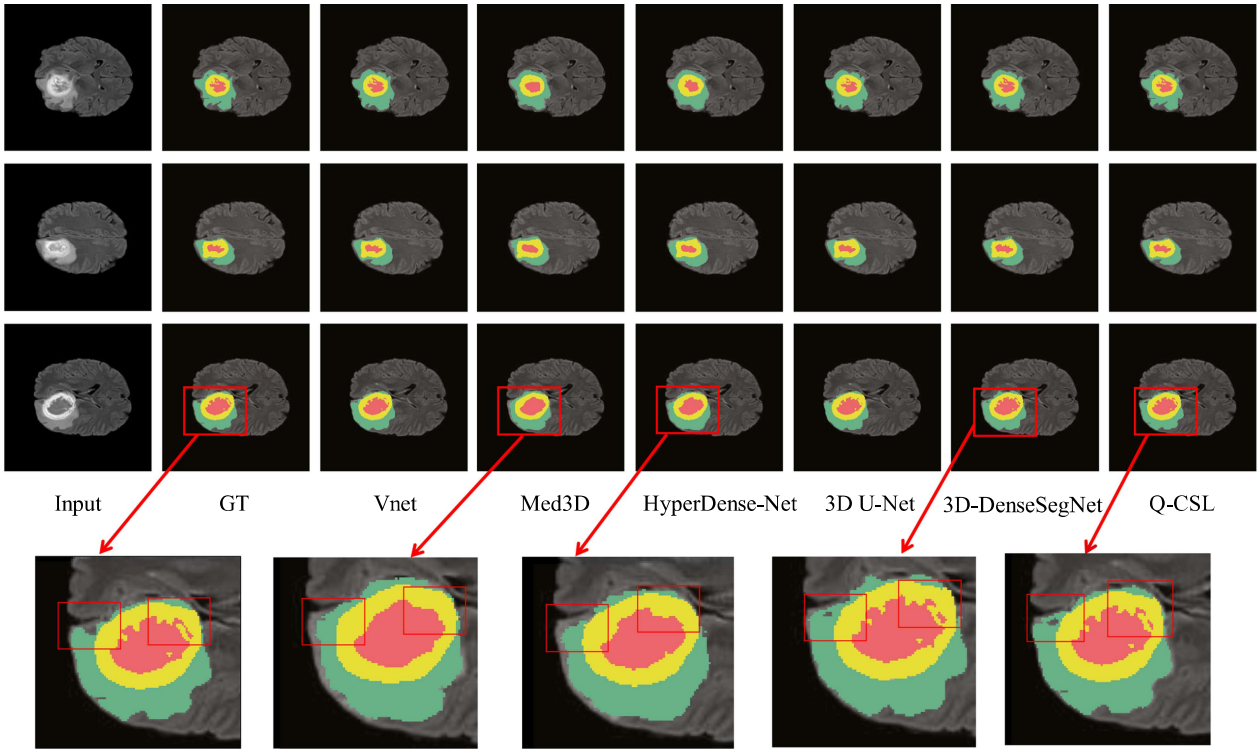


Fig. 7. Visualization of results on the BraTS 2021 Dataset. The tumor structures are depicted as Necrotic Tumor (NET)(red), Enhancing Tumor (ET)(yellow), as well as Peritumoral Edema (ED)(green) in these figures. For better visual the tumor details, we enlarge some results of methods.

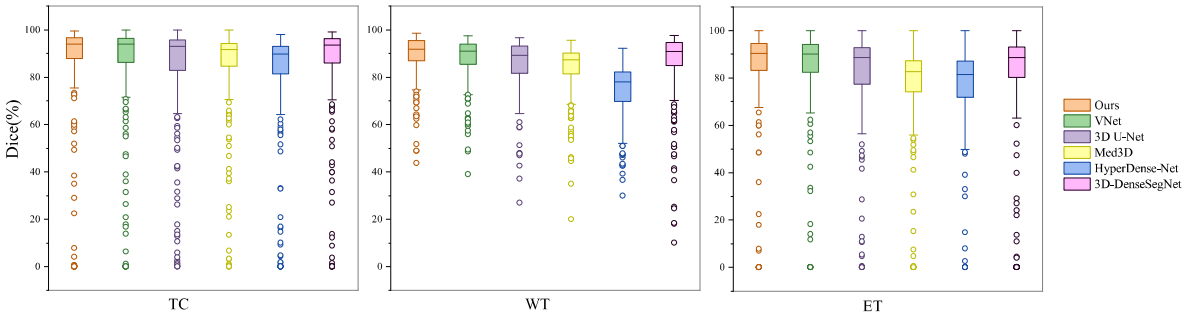


Fig. 8. Box plot of contrasting models in BraTS 2021 dataset. And the data in box plot are Dice score of every patient cases.

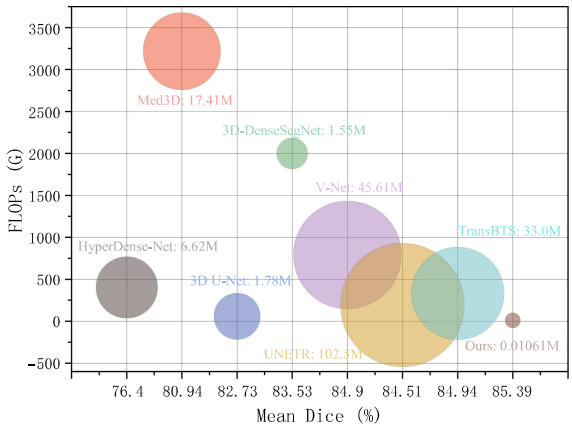


Fig. 9. Visualization of parameters, FLOPs, and mean Dice score of comparative models on bubble chart. The metric are based on BraTS 2021, and the image size is (128,128,128) with four modalities.

TABLE V
ABLATION STUDY OF Q-CSL ON THE BRATS 2021

Model	Dice(%)↑			HD95(mm)↓		
	ET	WT	TC	ET	WT	TC
Backbone	80.43	84.70	83.10	5.831	7.615	11.358
Replace to QSConv	81.57	89.12	84.24	5.745	8.124	6.324
+De-QCF	**82.21**	**89.27**	**84.69**	**5.744**	**5.477**	**6.303**

QSConv is quaternion spatial-association convolution and De-QCF is multi-modal fusion module.
The bold values are the best results in the metrics.

TABLE VI
DETAILED RESULTS OF PARAMETERS, FLOPS, AND MEAN DICE SCORE OF OUR METHOD IN ABLATION STUDY

Model	Params.(M)	FLOPs(G)	Mean Dice(%)
Backbone	20.53	725.85	82.74
Replace to QSConv	0.00972	9.33	84.97
+De-QCF	0.01061	9.95	85.39

The data is tested in the input size (4; 128; 128; 128) on brats 2021.

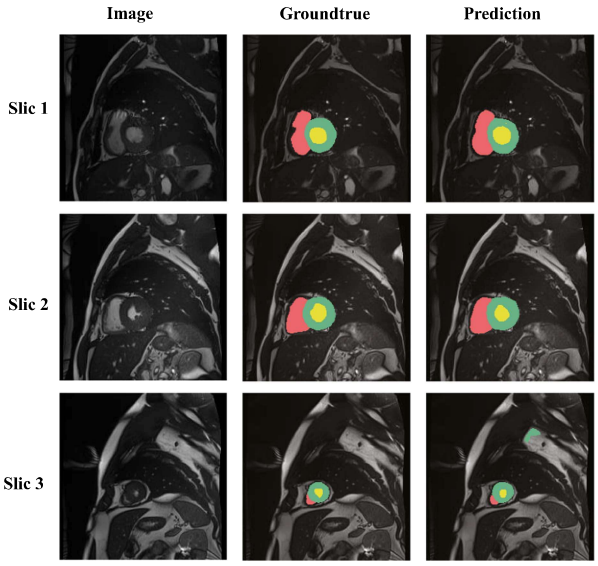


Fig. 10. Visualization of one case in ACDC dataset. There are three slic above, which contain original image, groundtrue, and prediction result.

2) Efficientness of De-QCF Module: The De-level Quaternion Cross-modality Fusion (De-QCF) Module is a plug-and-play module, aiming to explore inner modal spatial relationship and fuse different modalities attention features which relate local structure to global. In Table V, the results show that the module improves performance from 81.57% to 82.21% in ET, from 89.12% to 89.27% in WT and from 84.24% to 84.69% in TC, and the 95% Hausdorff Distance demonstrates that the model become stable after adding De-QCF. Generally, the De-QCF is evaluated to excavate the distributions of targets in local-global feature by intra-modal Hamilton-attention, and the De-level fusion is able to fuse different modal features in decode stage.

V. Discussion

From the above experimental results, we know that the Q-CSL is a superior spatial-aware segmentation method that inherits the lightweight advantage of quaternion. From the comparison results in BraTS, our Q-CSL achieves a great performance in Dice and HD95 scores with an awesome of model parameters (i.e., the 0.01061 M). In short, the superiority of our method has a great segmentation ability when the model parameter is tiny.

The superiority of segmentation reflects the spatial awareness of our method which is caused by the proposed quaternion-related representation and components. In contrast to proposed methods which neglect the modality fusion and relationship between lesions and coordinates [17], [58], our method combines the spatial information with grey values. Due to the correlation of different quaternion parts, the space and lesions are linked closer than that in the real-valued field. Furthermore, the Q-CSL utilizes a modality fusion mechanism that complements information of the more contributing modality to the feature outputs. It is worth to noting that the multi-modal data can be more efficient than a single modality.

Additionally, the most outstanding advantage of our method is the number of model parameters, which benefits from the weight-sharing strategy of quaternion convolution. From the discussion in the METHOD section, we know that the Quaternion-valued method can reduce learnable parameters to one-fourth of the real-valued method. In these works [59], [60], they point out that real-time efficiency is important to the segmentation application. Hence, we will focus on the lightweight model with the real-time applications.

Moreover, some problems limit the model performance improvement, i.e., the coordinates confusion and the large feature map memory. The consecutive coordinate values may cause the rare lesion distributions to be hard to identify due to the training experience of the model. Besides, although the model parameters are reduced, the memory sources are always occupied by huge feature maps during training, resulting in insufficient computational resources. Therefore, we look forward to representing the multi-modal data in quaternion and fusing the multi-modal via a computational cost-effective method.

VI. Conclusion

We propose a novel Quaternion Cross-modality Spatial Learning (Q-CSL) for multi-modal medical images segmentation. An entirely new representation of quaternion voxels is introduced. And quaternion spatial-association convolution (QSConv) excavates the inner spatial information in quaternion field. Then, De-level Quaternion Cross-modality Fusion (De-QCF) Module fuse each modal spatial-association feature and relate local to global. The key of these components is the Hamiltonian product. It shares each element with another quaternion, which fully exploits the information of both quaternions. On the BraTS 2021 and BraTS 2020 datasets, the results, FLOPs, and parameters demonstrate the segmentation capability of our framework. Q-CSL offers a new perspective on the multi-modal medical image segmentation task. We will look forward to better architecture and test the method in different organ medical images in the future.

References

[1] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Interv.*, 2015, pp. 234–241.
[2] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-Net: Fully convolutional neural networks for volumetric medical image segmentation," in *Proc. IEEE Int. Conf. 3D Vis.*, 2016, pp. 565–571.
[3] A. Myronenko, "3D MRI brain tumor segmentation using autoencoder regularization," in *Proc. Int. MICCAI Brainlesion Workshop*, 2018, pp. 311–320.
[4] H. M. Luu and S.-H. Park, "Extending nn-unet for brain tumor segmentation," in *Proc. Int. MICCAI Brainlesion Workshop*, 2021, pp. 173–186.
[5] T. Parcollet, M. Morchid, and G. Linarès, "A survey of quaternion neural networks," *Artif. Intell. Rev.*, vol. 53, pp. 2957–2982, 2020.
[6] M. Ye, J. Shen, X. Zhang, P. C. Yuen, and S.-F. Chang, "Augmentation invariant and instance spreading feature for softmax embedding," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 2, pp. 924–939, Feb. 2020.
[7] M. Ye, X. Lan, Q. Leng, and J. Shen, "Cross-modality person re-identification via modality-aware collaborative ensemble learning," *IEEE Trans. Imag. Process.*, vol. 29, pp. 9387–9399, Feb. 2020.
[8] T. Zhou, H. Fu, G. Chen, J. Shen, and L. Shao, "Hi-Net: Hybrid-fusion network for multi-modal MR image synthesis," *IEEE Trans. Med. Imag.*, vol. 39, no. 9, pp. 2772–2781, Sep. 2020.

[9] A. t. Nagrani, "Attention bottlenecks for multimodal fusion," in *Proc. Adv. Neural Inf. Process Syst.*, 2021, pp. 14200–14213.

[10] Y. Zhang et al., "Deep learning in food category recognition," *Inf. Fusion*, vol. 98, 2023, Art. no. 101859.

[11] H. Zhu, J. Wang, S.-H. Wang, R. Raman, J. M. Górriz, and Y.-D. Zhang, "An evolutionary attention-based network for medical image classification," *Int. J. Neural Syst.*, vol. 33, no. 3, 2023, Art. no. 2350010.

[12] W. Wang, Y. Pei, S.-H. Wang, J. Manuel Gorrz, and Y.-D. Zhang, "Pstcnn: Explainable COVID-19 diagnosis using PSO-guided self-tuning CNN," *Biocell: Official J. Sociedades Latinoamericanas de Microscopia Electronica*, vol. 47, no. 2, 2023, Art. no. 373.

[13] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, "3D U-Net: Learning dense volumetric segmentation from sparse annotation," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Interv.*, 2016, pp. 424–432.

[14] J. t. Schlemper, "Attention gated networks: Learning to leverage salient regions in medical images," *Med. Image Anal.*, vol. 53, pp. 197–207, 2019.

[15] Z. Zhou, Z. He, and Y. Jia, "Afpnet: A 3D fully convolutional neural network with atrous-convolution feature pyramid for brain tumor segmentation via MRI images," *Neurocomputing*, vol. 402, pp. 235–244, 2020.

[16] W. t. Wang, "Transbts: Multimodal brain tumor segmentation using transformer," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Interv.*, 2021, pp. 109–119.

[17] A. t. Hatamizadeh, "Unetr: Transformers for 3D medical image segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 574–584.

[18] H. Peiris, M. Hayat, Z. Chen, G. Egan, and M. Harandi, "A robust volumetric transformer for accurate 3D tumor segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Interv.*, 2022, pp. 162–172.

[19] A. Hatamizadeh, V. Nath, Y. Tang, D. Yang, H. R. Roth, and D. Xu, "Swin unetr: Swin transformers for semantic segmentation of brain tumors in MRI images," in *Proc. Int. MICCAI Brainlesion Workshop*, 2021, pp. 272–284.

[20] H.-Y. Zhou et al., "nnformer: Interleaved transformer for volumetric segmentation," *IEEE Trans. Image Process.*, vol. 32, pp. 4036–4045, 2023.

[21] C. J. Gaudet and A. S. Maida, "Deep quaternion networks," in *Proc. Int. Joint Conf. Neur. Netw.*, 2018, pp. 1–8.

[22] D. Comminiello, M. Lella, S. Scardapane, and A. Uncini, "Quaternion convolutional neural networks for detection and localization of 3D sound events," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2019, pp. 8533–8537.

[23] T. Parcollet et al., "Quaternion convolutional neural networks for end-to-end automatic speech recognition," in *Proc. 19th Conf. Int. Speech Commun. Assoc.*, Sep. 2018, pp. 22–26.

[24] D. Pavllo, D. Grangier, and M. Auli, "Quaternet: A quaternion-based recurrent model for human motion," in *Proc. Brit. Mach. Vis. Conf.*, Sep. 2018, p. 299.

[25] O. t. Bernard, "Deep learning techniques for automatic MRI cardiac multi-structures segmentation and diagnosis: Is the problem solved?," *IEEE Trans. Med. Imag.*, vol. 37, no. 11, pp. 2514–2525, Nov. 2018.

[26] Y. t. Ding, "MVFusFra: A multi-view dynamic fusion framework for multimodal brain tumor segmentation," *IEEE J. Biomed. Health Inform.*, vol. 26, no. 4, pp. 1570–1581, Apr. 2022.

[27] D. Xiang, B. Zhang, Y. Lu, and S. Deng, "Modality-specific segmentation network for lung tumor segmentation in PET-CT images," *IEEE J. Biomed. Health Inform.*, vol. 27, no. 3, pp. 1237–1248, Mar. 2023.

[28] S. Cui et al., Automatic semantic segmentation of brain gliomas from MRI images using a deep cascaded neural network," *J. Healthcare Eng.*, vol. 2018, Mar. 19, 2018, Art. no. 4940593, , doi: 10.1155/2018/4940593.

[29] G. Wang, W. Li, S. Ourselin, and T. Vercauteren, "Automatic brain tumor segmentation using cascaded anisotropic convolutional neural networks," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Interv.*, 2017, pp. 178–190.

[30] K. t. Kamnitsas, "Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation," *Med. Image Anal.*, vol. 36, pp. 61–78, 2017.

[31] D. Nie, L. Wang, Y. Gao, and D. Shen, "Fully convolutional networks for multi-modality isointense infant brain image segmentation," in *Proc. Int. Symp. Biomed. Imag.*, 2016, pp. 1342–1345.

[32] J. t. Dolz, K. Gopinath, J. Yuan, H. Lombaert, C. Desrosiers, and I. Ben Ayed, "Hyperdense-Net: A hyper-densely connected CNN for multi-modal image segmentation," *IEEE Trans. Med. Imag.*, vol. 38, no. 5, pp. 1116–1126, May 2019.

[33] Z. Liang, T. Wang, X. Zhang, J. Sun, and J. Shen, "Tree energy loss: Towards sparsely annotated semantic segmentation," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 16907–16916.

[34] Z. Liang, K. Guo, X. Li, X. Jin, and J. Shen, "Person foreground segmentation by learning multi-domain networks," *IEEE Trans. Imag. Process.*, vol. 31, pp. 585–597, 2022.

[35] M. Ye, J. Shen, and L. Shao, "Visible-infrared person re-identification via homogeneous augmented TRI-modal learning," *IEEE Trans Inf. Forensics Secur.*, vol. 16, pp. 728–739, 2021.

[36] J. t. Fu et al., "Dual attention network for scene segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3146–3154.

[37] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, "CCNet: Criss-cross attention for semantic segmentation," in *Proc. Int. Conf. Comput. Vis.*, 2019, pp. 6896–6908, 2019.

[38] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 3–19.

[39] G. T. Zhang, "Cross-modal prostate cancer segmentation via self-attention distillation," *IEEE J. Biomed. Health Inform.*, vol. 26, no. 11, pp. 5298–5309, Nov. 2022.

[40] Y. t. Jiang, "Apaunet: Axis projection attention unet for small target in 3D medical segmentation," in *Proc. Asian Conf. Comput. Vis.*, 2022, pp. 283–298.

[41] A. Sinha and J. Dolz, "Multi-scale self-guided attention for medical image segmentation," *IEEE J. Biomed. Health Inform.*, vol. 25, no. 1, pp. 121–130, Jan. 2021.

[42] Z. Xing, L. Yu, L. Wan, T. Han, and L. Zhu, "Nestedformer: Nested modality-aware transformer for brain tumor segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Interv.*, 2022, pp. 140–150.

[43] D.-P. t. Fan, "Inf-Net: Automatic COVID-19 lung infection segmentation from CT images," *IEEE Trans. Med. Imag.*, vol. 39, no. 8, pp. 2626–2637, Aug. 2020.

[44] X. Zhu, Y. Xu, H. Xu, and C. Chen, "Quaternion convolutional neural networks," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 631–647.

[45] T. Parcollet, M. Morchid, and G. Linarès, "Quaternion convolutional neural networks for heterogeneous image processing," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2019, pp. 8514–8518.

[46] M. R. Celsi, S. Scardapane, and D. Comminiello, "Quaternion neural networks for 3D sound source localization in reverberant environments," in *Proc. IEEE Int. Workshop Mach. Learn. Signal Process.*, 2020, pp. 1–6.

[47] X. Qiu, T. Parcollet, M. Ravanelli, N. D. Lane, and M. Morchid, "Quaternion neural networks for multi-channel distant speech recognition," in *Proc. 21st Annu. Conf. Int. Speech Commun. Assoc. Interspeech*, Shanghai, China, Oct. 25–29, 2020, pp. 329–333.

[48] S. Bakas et al., "Advancing the cancer genome atlas glioma MRI collections with expert segmentation labels and radiomic features," *Sci. Data*, vol. 4, no. 1, pp. 1–13, 2017.

[49] U. Baid et al., "The rsna-asnr-miccai brats 2021 benchmark on brain tumor segmentation and radiogenomic classification," 2021, *arXiv:2107.02314*.

[50] B. H. t. Menze, "The multimodal brain tumor image segmentation benchmark (BRATS)," *IEEE Trans. Med. Imag.*, vol. 34, no. 10, pp. 1993–2024, Oct. 2015.

[51] L. R. Dice, "Measures of the amount of ecologic association between species," *Ecology*, vol. 26, no. 3, pp. 297–302, 1945.

[52] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Representation*, Y. Bengio and Y. LeCun, Eds. May 2015.

[53] S. Chen, K. Ma, and Y. Zheng, "Med3D: Transfer learning for 3D medical image analysis," 2019, *arXiv:1904.00625*.

[54] T. D. Bui, J. Shin, and T. Moon, "3D densely convolutional networks for volumetric segmentation," 2017, *arXiv:1709.03199*.

[55] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "Unet++ : Redesigning skip connections to exploit multiscale features in image segmentation," *IEEE Trans. Med. Imag.*, vol. 39, no. 6, pp. 1856–1867, Jun. 2020.

[56] Y. Xie, J. Zhang, C. Shen, and Y. Xia, "Cotr: Efficiently bridging CNN and transformer for 3D medical image segmentation," in *Proc. 24th Int. Conf. Med. Imag. Comput. Comput. Assist. Interv.*, 2021, pp. 171–180.

[57] C. Lyu and H. Shu, "A two-stage cascade model with variational autoencoders and attention gates for MRI brain tumor segmentation," in *Proc. Brainlesion: Glioma, Mult. Sclerosis, Stroke Traumatic Brain Injuries: 6th Int. Workshop*, 2020, pp. 435–447.

[58] A. S. Akbar, C. Fatichah, and N. Suciati, "Unet3D with multiple atrous convolutions attention block for brain tumor segmentation," in *Proc. Int. MICCAI Brainlesion Workshop*, 2021, pp. 182–193.

[59] J. Shen, Y. Liu, X. Dong, X. Lu, F. S. Khan, and S. Hoi, "Distilled siamese networks for visual tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 12, pp. 8896–8909, Dec. 2022.

[60] Z. Zhao, S. Zhao, and J. Shen, "Real-time and light-weighted unsupervised video object segmentation network," *Pattern Recognit.*, vol. 120, 2021, Art. no. 108120.

Complex & Intelligent Systems (2025) 11:84

ORIGINAL ARTICLE

Adaptive micro partition and hierarchical merging for accurate mixed data clustering

Yunfan Zhang[1]

Received: 5 November 2023 / Accepted: 23 November 2024 / Published online: 19 December 2024

Abstract
Heterogeneous attribute data (also called mixed data), characterized by attributes with numerical and categorical values, occur frequently across various scenarios. Since the annotation cost is high, clustering has emerged as a favorable technique for analyzing unlabeled mixed data. To address the complex real-world clustering task, this paper proposes a new clustering method called Adaptive Micro Partition and Hierarchical Merging (AMPHM) based on neighborhood rough set theory and a novel hierarchical merging mechanism. Specifically, we present a distance metric unified on numerical and categorical attributes to leverage neighborhood rough sets in partitioning data objects into fine-grained compact clusters. Then, we gradually merge the current most similar clusters to avoid incorporating dissimilar objects into a similar cluster. It turns out that the proposed approach breaks through the clustering performance bottleneck brought by the pre-set number of sought clusters k and cluster distribution bias, and is thus capable of clustering datasets comprising various combinations of numerical and categorical attributes. Extensive experimental evaluations comparing the proposed AMPHM with state-of-the-art counterparts on various datasets demonstrate its superiority.

Keywords Cluster analysis · Heterogeneous attributes · Neighborhood rough set · Unsupervised learning

Yunfan Zhang are co-first authors.

Yunfan Zhang

Introduction

Cluster analysis is a frequently employed real-world data analytics technique [1]. The primary aim of cluster analysis is to discover inherent patterns or structures within the data. In real scenarios, datasets often include a variety of attributes that may have diverse value types, such as numerical attributes and categorical attributes [2]. Unlike numerical attributes, which have well-defined distances in Euclidean space, categorical attributes cannot undergo arithmetic computations and lack a well-defined similarity space due to the absence of explicit numerical meanings and the presence of divergent concepts among different possible attribute values.

Table 1 demonstrates two fragments of such real datasets with all two types of attributes. The upper fragment is from a medical dataset, containing the categorical attributes "Scalp",[1] "Family",[2] numerical attribute "Age", and class attribute "Diagnosis". The lower part is a fragment of the Contraceptive Prevalence Survey, which includes categori-

[1] "Scalp" refers to the scalp involvement in the dataset.

[2] "Family" refers to the family history in the dataset.

Table 1 Fragments of mixed datasets "dermatology" (upper part) and "contraceptive method choice" (lower part)

ID	Scalp[1] (categorical)	Family[2] (categorical)	Age (numerical)	...	Diagnosis
$Patient_1$	None	No	55	...	seboreic dermatitis
$Patient_2$	Mild	Yes	8	...	psoriasis
$Patient_3$	None	No	26	...	lichen planus
$Patient_4$	Moderate	No	40	...	psoriasis
ID	Education (categorical)	Exposure[3] (categorical)	Children (numerical)	...	Contraceptive used
$Family_1$	Basic	Good	3	...	No-use
$Family_2$	Low	Not good	10	...	No-use
$Family_3$	High	Not good	2	...	Long-term
$Family_4$	Intermediate	Good	1	...	Short-term

cal attributes "Education", "Exposure",[3] numerical attribute "Children", and class attribute "Contraceptive used". Such mixed datasets are prevalent in real-world unsupervised learning tasks, while the heterogeneity of attributes introduces complexities into cluster analysis. Specifically, when conventional Hamming distance is employed and only yields "0" and "1" distances, the original dissimilarity degrees among categorical attribute values are thus wasted.

To cope with the mixed data cluster problem, most existing clustering methods can be roughly divided into two types: (i) approaches that directly weight and fuse different dissimilarity measures, e.g., Euclidean [3] and Hamming distances [4], and (ii) approaches that first define a metric for heterogeneous attributes and then perform based on it.

For the former type [5, 6], numerical and categorical attributes are typically measured using the Euclidean and Hamming distances, respectively, for combination. To enhance the effectiveness of combining metrics, metric-based approaches have been designed in recent years, which measure the similarity between categorical attribute values [7] based on the inter-object dissimilarities present in the dataset as a solution for capturing the relationships between values and attributes in categorical data. Recently, some metrics [8, 9] are introduced to measure the similarity of possible values as the Conditional Probability Distributions (CPDs) of their respective values across disparate attributes. However, all the aforementioned approaches are proposed for nominal data only, which is a subtype of categorical data.

For the latter type that defines metrics for heterogeneous attributes, some measures and metrics are defined in a unified way in the literature. The similarity metric [20] quantifies the similarity of data objects within a unified probability framework, addressing both numerical and categorical attributes, but ignores to further measuring the dissimilarity of divergent concepts among different possible attribute values. Lin's distance metric [10] and the entropy-based similarity metric [16] consider the value orders of categorical attributes and measure the dissimilarity between potential values from an information theory standpoint. However, these dissimilarity-based clustering methods often fail to account for the varying degrees of dissimilarity among heterogeneous attribute values, which may result in grouping less similar data objects or splitting similar ones. For defining similarities more finely, the state-of-the-art methods may still encounter challenges when it comes to clustering various heterogeneous attribute data. For example, context-based distance metrics [13–15] quantify interdependence among categorical attributes, select relevant attributes, and use CPDs to indicate similarity between potential attribute values. Recently, there has been an exploration of the interdependence between categorical attributes through the development of a unified distance metric [18]. Most recently, more advanced solutions, including Mix2Vec [21], Het2Hom [22] and homogeneous distance metric [19] have been proposed. The first two approaches effectively preserve the intrinsic structural information of heterogeneous attribute values. However, their effectiveness heavily depends on the chosen encoding strategies for categorical attributes, e.g., Het2Hom projects all attribute values into spaces created by different pairs of concepts related to that attribute. In contrast, the third approach [19] uniformly establishes distance metric to measure nominal [23] and ordinal categorical attributes but is not suitable for datasets containing typical numerical attributes.

In general, all the approaches mentioned for clustering mixed data also face a fundamental challenge, i.e., the limited application caused by the reliance on prior knowledge or assumptions. Even the most appropriately defined metrics for handling heterogeneous attributes rely on certain assumptions or models, such as probability, entropy, and so on. These assumptions and models are based on prior knowledge, which may not always be well-suited for the complexity of real datasets and clustering tasks. Table 2 provides a sum-

[3] "Exposure" refers to the media exposure in the dataset.

Table 2 The significant issues considered by existing clustering methods

Methods	Heterogeneity	Intra-attribute	Intter-attribute	Importance	Unification
Euclidean + hamming distance [5]	√	√			
Lin's similarity metric [10]		√			
Association-based dissimilarity measure [11]			√		
Attribute-weighting k-means [12]	√	√		√	
Context-based measure [13], [14]			√	√	
Jia's distance metric [15]		√	√	√	
Entropy-based distance metric [16]		√		√	
Distance learning-based clustering [17]		√		√	
Heterogeneous coupling learning [9]		√	√	√	
Unified distance metric [18]		√	√	√	
Homogeneous distance measure [19]		√		√	
AMPHM (proposed)	√	√	√	√	√

mary of various existing methods about their consideration of significant mixed data clustering issues, where "Heterogeneity", "Intra-attribute", "Inter-attribute", "Importance", "Unification" indicate the consideration of: (1) data heterogeneity (i.e., the existence of data made of both numerical and categorical attributes); (2) intra-attribute dependence; (3) inter-attribute dependence; (4) importance of different attributes; and (5) unification of distance metrics (i.e., unification of metrics of numerical and categorical attributes). It can be seen that none of the existing methods considers all the significant issues, which may result in an unsatisfactory clustering performance.

Therefore, we propose a novel clustering algorithm that is based on original data information. Specifically, only the data objects with a relatively sparse boundary, allowing them to be clearly distinguished from their surrounding objects, are included in the micro partition, thereby reducing the grouping of objects that are not very similar based on existing distance measures. We utilize the neighborhood rough set theories, which are designed to group local similar data objects into one neighborhood set, to form the spatially local compact clusters. Our proposed method can appropriately partition data objects based on distance and density while ensuring that the granularity of each small cluster is fine-tuned. These clusters can be called micro clusters. To merge these micro clusters appropriately, we further propose a merging mechanism that initiates the entire clustering process from the most granular level, i.e., micro clusters, and gradually merges to the more rough-grained state of larger clusters, ultimately forming rational clusters. Comprehensive experiments including clustering performance and ablation studies have been conducted on various real-world datasets to verify the effectiveness of our proposed method.

The main contributions of this article are three-fold:

(1) To the best of our knowledge, this makes the first attempt to specifically formulate a clustering approach based on neighborhood rough set theory and hierarchical merging for addressing the problem of reliance on prior knowledge and assumptions in existing methods. Thus, this innovative paradigm has the potential to stimulate further exploration of unsupervised learning tasks.
(2) A novel neighborhood relation is proposed to simultaneously take into account both distance and density, which partition heterogeneous attribute data into fine-grained clusters to ensure local compact partition while the similar objects may cooperate to reject to be merged into dissimilar objects. It turns out that it applies to analyzing a wide range of data with complex object distributions.
(3) With the proposed novel merging mechanism, the algorithm hierarchically merges micro clusters, so that the micro clusters are processed to a more rough-grained state of larger clusters. This process appropriately extracts and exploits information. Therefore, it achieves a more intuitively rational inference process in forming the clusters and boosts the performance.

The rest of this article is organized as follows. Section "Related work" provides an overview of existing related techniques. In Sect. "Proposed method", the heterogeneous attribute metric, the micro partition, the merging mechanism, and the proposed algorithm are presented. Then, experimental results are shown in Sect. "Experiments". Finally, we conclude in Sect. "Conclusion".

Related work

As our approaches mainly consist of two parts, i.e., dissimilarity measurement in the existing heterogeneous attribute

data clustering method, and neighborhood set, this section mainly makes an overview of the existing above-mentioned related works, focusing on mixed data clustering methods and neighborhood rough sets.

Existing mixed data clustering methods

Early attempts like the k-prototypes method and its variants [5, 6], used one-hot encoding to represent categorical attribute values [24] as binary vectors, which allowed these attributes to be treated as numerical data. Then, numerical attributes were typically measured using the Euclidean distance, while categorical attributes employed the Hamming distance [4], which has limitations in distinguishing the dissimilarities between different value pairs. Hence, the conventional k-prototypes method overlooks the fact that the categorical attributes do not contain a well-defined space.

Thus, many advanced similarity-based methods were designed to preserve the information and cluster the mixed data well. Using statistical information from related attributes to reflect similarities in a target categorical attribute is a viable method for measuring similarity more reasonably. Various measures in this field, such as association-based [11], Ahmad's [25], and context-based [13, 14] measures, share a common approach to calculating the distance between two conditional probability distributions derived from related attributes, which indicates their dissimilarity. Recent metrics [15, 26] can effectively utilize both intra-attribute and inter-attribute information. This prevents issues with measurement when attributes are independent of each other. Most recently, a method [27] employed PCA and density to merge data. However, the above-mentioned approaches still have not well solved the heterogeneity of various categorical attributes.

Lin's distance metric [10] was one of the initial efforts to define suitable measures for categorical attributes. It calculated entropy values for attribute values to express their similarity. The entropy-based distance metric [16, 28] unifies the similarity of categorical attributes and provides an attribute weighting scheme. Moreover, the distance learning-based approaches [17, 29] innovatively learned and adjusted the order structure of the ordinal attributes and then clustering them. After this, the unified distance metric [15] was introduced to unify the measurement of distances between nominal and ordinal attributes, consistently capturing valuable relationship information. Most recently, the homogeneous distance metric [19] was introduced, focusing on the graph structure of attribute values, while the Mic2Mac [30] has been proposed to cluster heterogeneous attribute data based on the neighborhood set.

For data representation methods, common methods include numerical coding (NC), which uses one-hot encoding for nominal attributes and ranking for ordinal ones. Nonetheless, NC has its drawbacks, such as the absence of robust theoretical foundations, resulting in the curse of dimensionality for nominal attributes. Therefore, to be interpretable and facilitate a more comprehensive exploration of information, an advanced representation method introduced in [9] uses multiple kernel spaces to learn different types of value couplings, which has demonstrated superior clustering performance on categorical data. Recently, an entropy-based hierarchical clustering approach [31] has been proposed to analyze the nominal attribute, while an order forest method [32] has been proposed to tackle qualitative-attribute clustering tasks. It's essential to emphasize that the representation methods mentioned above are tailored exclusively for categorical data. The above-mentioned researches have promising prospects in switched reaction–diffusion systems [33], switched nonlinear systems [34, 35], and nanodental internet of things systems [36]. Meanwhile, competitive learning also has been introduced into categorical data clustering [37] and federated clustering [38].

Most recently, for large-scale datasets, the deep learning model-based clustering approach Mix2Vec [21] and QGRL [39] converts heterogeneous attribute values into vectors while preserving their structural distribution information, which has many promising applications, e.g., agricultural data [40], financial decision aid [41]. However, it still treats numerical attributes and categorical attributes differently and does not fully account for the heterogeneity of categorical attributes.

Overall, most existing methods for clustering heterogeneous attributes typically face one or both of the following limitations: (1) they are designed for datasets with one specific type of attributes only, and (2) they rely heavily on prior knowledge or assumptions.

Neighborhood rough sets

Neighborhood rough set (also referred to interchangeably as the neighborhood set for simplicity) is commonly used for partitioning categorical or heterogeneous attribute biomedical data [42]. Specifically, it lets each data object $\mathbf{x}_i$ to find a neighborhood set $N(\mathbf{x}_i)$, consisting of closer data objects (i.e. with smaller distance) to $\mathbf{x}_i$. The distance between $\mathbf{x}_i$ and $\mathbf{x}_j$, as reflected by attributes A, is denoted as $D(\mathbf{x}_i, \mathbf{x}_j)$. Conventionally, there are two types of neighborhood relations, i.e., the k-nearest

$$N^k(\mathbf{x}_i) = \{\mathbf{x}_j | D_k(\mathbf{x}_i, \mathbf{x}_j) < D(\mathbf{x}_i, \mathbf{x}_g)\}, \tag{1}$$

and the δ-radius

$$N^\delta(\mathbf{x}_i) = \{\mathbf{x}_j | D(\mathbf{x}_i, \mathbf{x}_j) \leq \delta\}, \tag{2}$$

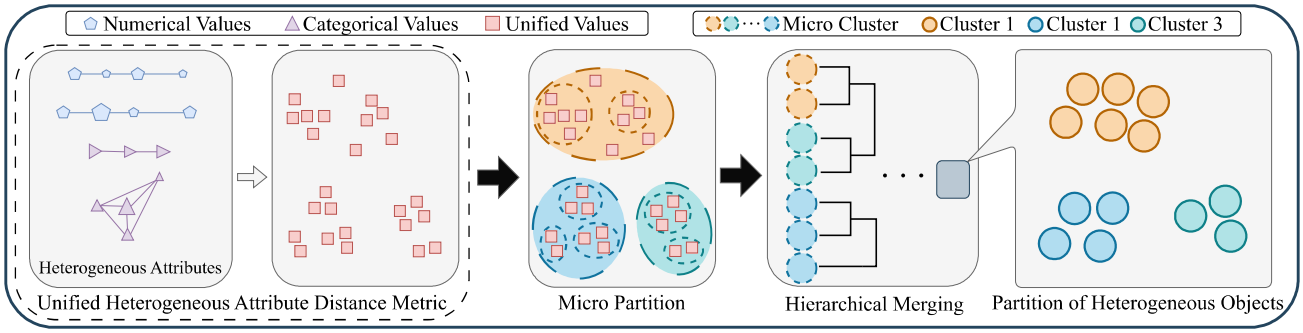


Fig. 1 Pipline of the proposed approach. Initially, we use the heterogeneous distance metric to measure different types of attributes in a unified way. Then, our proposed micro partition and hierarchical merging are used to form and merge the micro clusters, respectively. Finally, we can partition mixed attribute data

where $j, g \in \{1,2,...,n\}$, $g \neq j$, and the k in Eq. (1) represent the first k data objects with the nearest distance to $\mathbf{x}_i$. To simplify without causing any ambiguity, we use $N(\mathbf{x}_i)$ to refer to the neighborhood relation in a general sense. The classical rough set is to select categorical attribute subsets, which is applied in two Online Feature Selection (OFS) approaches [43, 44], which aim to select attribute subsets that effectively distinguish more data objects. However, these methods operate under the assumption that all attributes are categorical, which does not align with the characteristics of most real datasets.

Thus, many heterogeneous attribute selection methods [45] have been proposed. Recently, a more advanced [46] has been proposed, which incorporates entropy to improve the distinction of neighborhoods. Nevertheless, the challenging information gap between heterogeneous attributes cannot be effectively bridged through metric combination through these neighborhood sets approaches, potentially leading to significant information loss in subsequent analyses.

Proposed method

Our proposed clustering approach AMPHM primarily consists of three parts, i.e., heterogeneous attribute distance metric, neighborhood set-based micro partition, and hierarchical merging mechanism. Specifically, the second part is to partition objects appropriately based on the first proposed distance metric. Then the hierarchical merging mechanism merges neighborhood sets until the number of neighborhood sets is equal to the number of clusters. The whole technological process is shown in Fig. 1. Therefore, in this section, we start by defining the problem, then we introduce the heterogeneous attribute distance metric, neighborhood rough set-based micro partition, hierarchical merging mechanism, and the whole clustering algorithm in Section "Problem formulation", respectively.

Problem formulation

Table 3 provides a list of frequently used symbols in this paper, and their specific definitions will be provided upon their first appearance in the following text. A heterogeneous attribute dataset, denoted as $X = \{\mathbf{x}_1, \mathbf{x}_2,..., \mathbf{x}_n\}$, is a set of n data objects. $A = \{A_1, A_2,..., A_d\}$ is the attribute set consisted of d attributes. The value set of attributes, denoted as $V = \{V_1, V_2,..., V_d\}$, stores the value domains corresponding to each attribute, with V_r representing the possible values of the r-th attribute. Clustering involves assigning n data objects in X to k appropriate clusters, denoted as $C = \{C_1, C_2,..., C_l,..., C_k\}$, where C_l is a cluster containing a set of data objects, and $X = \bigcup_{l=1}^{k} C_l$. The set of representative data objects in each cluster is denoted as $R = \{\mathbf{r}_1, \mathbf{r}_1,..., \mathbf{r}_g,..., \mathbf{r}_k\}$. A commonly used representation for clustering involves the maintenance of an $n \times k$ partition matrix $\mathbf{U}$, which indicates the assignment of n objects to k clusters. The value of the (i, l)-th entry in $\mathbf{U}$ as $u_{i,l}$. and its value is calculated by

$$u_{i,l} = \begin{cases} 1, & l = arg\min_{g} D(\mathbf{x}_i, \mathbf{r}_g) \\ 0, & \text{otherwise,} \end{cases} \tag{3}$$

according to (3) we have

$$\sum_{l=1}^{m} u_{i,l} = 1, 1 \leq i \leq n, \tag{4}$$

and $u_{i,l} \in \{0, 1\}$. Since the neighborhood sets for our hierarchical merging is based on $N(\mathbf{x}_i)$, while obtaining $N(\mathbf{x}_i)$ needs the inter-object distances, we adopt a common form of inter-object distance based on attributes A as

Table 3 Explanation of symbols

Symbol	Explanation
X	Dataset
$\mathbf{x}_i$	i-th data object
x_i^r	r-th value of $\mathbf{x}_i$
A	Attribute set
A_r	r-th attribute
V_r	Possible value of A_r
v_h^r	h-th value of V_r
$\mathrm{v_r}$	Number of possible values of A_r
R	Representative objects set
$\mathbf{r}_g$	g-th representative object R
d	Number of attributes
n	Number of objects
C_l	l-th cluster
$u_{i,l}$	A value indicating the affiliation between $\mathbf{x}_i$ and c_l
$N^{\xi}(\mathbf{x}_i)$	The micro partition of $\mathbf{x}_i$
ρ_i	i-th density
ξ_i	i-th density gap
$D(\cdot,\cdot)$	Data dissimilarity
w^{rt}	Weight indicating the importance of A_t to A_r

$$D(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{\sum_{A_r \in A} D^r(x_i^r, x_j^r)^2}, \tag{5}$$

where $x_i^r \in V_r$ is the value of $\mathbf{x}_i$ on attribute A_r, while $D^r(x_i^r, x_j^r)$ measures the distance between $\mathbf{x}_i$ and $\mathbf{x}_j$ in terms of A_r. In the following subsections, we present how to define $D^r(x_i^r, x_j^r)$, $N(\mathbf{x}_i)$, and the hierarchical merging mechanism, respectively.

HADM: heterogeneous attribute distance metric

To construct the unified distance metric $D^r(x_i^r, x_j^r)$ concerning different heterogeneous attributes A_r, we employ a transformation cost as a metric. This cost quantifies the effort needed to convert one Conditional Probability Distribution (CPD) into another. To provide a more intuitive explanation of the principles, we will begin by specifying the CPD and formulating the distance between possible values of a categorical attribute. Afterward, we demonstrate how this distance unifies both categorical and numerical cases. Lastly, we will derive the object-level distance. The CPD of an attribute A_t, which has v^t possible values $V_t = \{v_1^t, v_2^t, ..., v_{\mathrm{v_t}}^t\}$, given a possible value v_g^r of another attribute A_r, can be expressed as

$$\mathbf{P}_g^{r \leftarrow t} = [p(v_1^t | v_g^r), p(v_2^t | v_g^r), ..., p(v_{\mathrm{v_t}}^t | v_g^r)], \tag{6}$$

where $p(v_h^t | v_g^r)$ is the conditional probability of v_h^t as given v_g^r. We represent the CPD as $\mathbf{P}_g^{r \leftarrow t}$. The superscript $r \leftarrow t$ indicates that this CPD describes the gth possible value of A_r according to the values of A_t. For simplicity, we denote $r \leftarrow t$ as rt from here on. The difference between two CPDs, such as $\mathbf{P}_g^{rt}$ and $\mathbf{P}_q^{rt}$, effectively reflects the dissimilarity between two possible values, i.e., v_g^r and v_q^r. Hence, we employ Earth Mover's Distance (EMD) [47, 48], originally developed for calculating the transformation cost between two histogram descriptors, to quantitatively measure the difference between the CPDs describing two categorical possible values. Consequently, the distance between v_g^r and v_q^r as represented by A_t can be denoted using EMD as

$$D^{rt}(v_i^r, v_j^r) = \sigma((\mathbf{P}_i^{rt} - \mathbf{P}_j^{rt}, \mathbf{0})) \cdot \mathbf{1}, \tag{7}$$

where $\sigma(\cdot,\cdot)$ matches each pair of corresponding bits in two vectors and selects the maximum value. The $\mathbf{0}$ and $\mathbf{1}$ are v^t-dimensional vectors with all elements equal to 0 and 1, respectively. Due to varying degrees of inter-attribute dependence, different attributes A_ts may have different contributions based on their respective weights w^{rt} when forming the overall distance $D^{rt}(v_g^r, v_h^r)$ between v_g^r and v_h^r by

$$D^r(v_i^r, v_j^r) = \sum_{A_t \in A} D^{rt}(v_i^r, v_j^r) \cdot w^{rt}. \tag{8}$$

The contribution w^{rt} indicated in Eq. (8) by using this dependence, which measures the inter-attribute dependence between A_r and A_t, can be written as

$$w^{rt} = \frac{\sum_{g=1}^{\mathrm{v}^r-1} \sum_{h=g+1}^{\mathrm{v}^r} D^r(v_g^r, v_h^r)}{\mathrm{v}^r(\mathrm{v}^r - 1)/2}. \tag{9}$$

Considering the possible values of a categorical attribute as concepts proposed in [49], Eq. (9) essentially measures the average inter-concept distances of A_r, as indicated by A_t. To clarify Eq. (9), we consider an extreme example. When A_r and A_t are identical, they are perfectly interdependent, and the corresponding $D^{rt}(v_g^r, v_h^r)$ always reaches the maximum value of "1" for any combinations of g and h with $g \neq h$, as stated in Eq. (7). Therefore, the corresponding w^{rt} also reaches the maximum value of "1", signifying a complete 100% inter-attribute dependence. Based on Eqs. (7)–(9), the distance between $\mathbf{x}_i$ and $\mathbf{x}_j$ w.r.t. attribute A can be calculated using Eq. (5). Now we demonstrate the applicability of the defined distance metric to both categorical and numerical attributes while satisfying the properties of a metric.

Theorem 1 *Equation (8) treats both categorical and numerical attributes in a homogeneous way, under the assumption that the distance space for each numerical attribute is an independent one-dimensional continuous Euclidean distance space with a domain of [0, 1].*

Proof Each numerical attribute space can be considered a linear space with an infinite number of evenly spaced possible values. Because they are independent, w^{rt} is always equal to 0 for numerical attributes where $r \neq s$. Therefore, we should only consider the special case, i.e., r equals s, where $w^{rt} = \lim_{h\to\infty}(\sum_{g=1}^{h} 1/h)/1 = 1$ according to Eq. (9). Here, $h \to \infty$ represents the number of intervals between adjacent possible values. The denominator "1" reflects the presence of only one inter-concept distance because there are only two concepts, "0" and "1". Therefore, when $r = s$, the EMD with the use of Euclidean distance simplifies to $D^{rt}(v_g^r, v_h^r) = |v_g^r - v_h^r|$, which is essentially equivalent to the Euclidean distance in $D(\mathbf{x}_i, \mathbf{x}_j)$ as described in Eq. (5).□

Theorem 2 *$D(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is a distance metric.*

Proof As Eq. (7) is a metric, it follows that Eq. (8) based on Eq. (7) is also a metric. Additionally, Eq. (5), which is computed using finite arithmetic operations based on Eq. (8), ensures that $D(\mathbf{x}_i, \mathbf{x}_j)$ satisfies all the properties of a metric:

(1) $D(\mathbf{x}_i, \mathbf{x}_j) \geq 0$; $D(\mathbf{x}_i, \mathbf{x}_j) = 0$ iff $\mathbf{x}_i = \mathbf{x}_j$;
(2) $D(\mathbf{x}_i, \mathbf{x}_j) = D(\mathbf{x}_j, \mathbf{x}_i)$;
(3) $D(\mathbf{x}_i, \mathbf{x}_j) \leq D(\mathbf{x}_i, \mathbf{x}_k) + D(\mathbf{x}_k, \mathbf{x}_j)$.

□

NRSMP: neighborhood rough set-based micro partition

The existing conventional neighborhood rough set $N^k(\mathbf{x}_i)$ and $N^\delta(\mathbf{x}_i)$ (i.e., Eqs. (1) and (2)) create n neighborhood sets that may partially overlap with each other. This overlapping can be problematic for partitions because it introduces high computational costs when considering each current attribute. Furthermore, in the case of unevenly distributed data objects, $N^k(\mathbf{x}_i)$ might inappropriately group dissimilar objects into the same neighborhood set, while $N^\delta(\mathbf{x}_i)$ could create neighborhood sets that overlap with boundaries. These neighborhood sets located in the central regions, may not effectively support our partitioning but would require a significant computational cost.

To effectively distinguish class boundaries and partition objects more efficiently while reducing computational costs, we introduce a novel approach called the neighborhood rough set-based micro partition. This approach aims to create non-overlapping neighborhood sets. To achieve this, we begin by selecting representative data objects, which gather neighbors based on the concept of a density gap.

Definition 1 Density gap: Density gap ξ_i, for an object $\mathbf{x}_i$ with a density of ρ_i, represents the minimum distance between $\mathbf{x}_i$ and $\mathbf{x}_j$ with a higher density ρ_j. It can be expressed as

$$\xi_i = minD(\mathbf{x}_i, \mathbf{x}_j), \quad s.t.\ \rho_i < \rho_j \ and\ \mathbf{x}_i \in X \backslash \mathbf{x}_j, \tag{10}$$

where ρ_i and ρ_j are the density of $\mathbf{x}_i$ and $\mathbf{x}_i$, respectively. And the $X\backslash\mathbf{x}_j$ is the dataset excluding $\mathbf{x}_j$. Moreover, for the object with the highest density, its density gap is defined as the maximum $D(\mathbf{x}_i, \mathbf{x}_j)$ with $\mathbf{x}_j \in X$.

The density ρ_i in Definition 1 can be computed by

$$\rho_i = \frac{D(\mathbf{x}_i, \mathbf{x}_{\langle i,g_i\rangle})}{g_i}, \tag{11}$$

where g_i represents the rank of a neighbor object $\mathbf{x}_{\langle i,g_i\rangle} \in D$ w.r.t. $\mathbf{x}_i$. This rank signifies that $\mathbf{x}_{\langle i,g_i\rangle}$ is the g_i-th closest object to $\mathbf{x}_i$ among all n data objects. More specifically, we initially construct the *ascending* micro partition set MP_i concerning $\mathbf{x}_i$ as $MP_i = \{\mathbf{x}_{\langle i,1\rangle}, \mathbf{x}_{\langle i,2\rangle}, ..., \mathbf{x}_{\langle i,n\rangle}\}$, where $\mathbf{x}_{\langle i,1\rangle} \equiv \mathbf{x}_i$ and $MP_i(j) = \mathbf{x}_{\langle i,j\rangle}$. Next, while iterating through MP_i from left to right, we select the object $\mathbf{x}_{\langle i,h\rangle}$ that initially fulfills the condition $D(\mathbf{x}_i, \mathbf{x}_{\langle i,h\rangle})/\ D(\mathbf{x}_{\langle i,h-1\rangle})/(h\text{-}1)$ determines the value of g_i as $g_i = h - 2$. The density computation intuitively selects neighbor objects, ensuring objects beyond a significant class boundary are not incorporated into the $\mathbf{x}_i$ neighborhood set.

Clearly, a representative object in a neighborhood set should be encircled by neighbors of lower density and also be situated at a considerable distance from other representative objects. As per Definition 1, objects with higher density gaps are better suited to serve as representative objects. Consequently, we initiate the micro partitioning process by ranking data objects in *descending* order based on density gaps and then creating micro partition accordingly

$$N^\xi(\mathbf{x}_i) = \left\{ \bigcup_{j=1}^{g_i} MP_i(j) \right\} \setminus \left\{ \bigcup_{\xi_q \leq \xi_i} N^\xi(\mathbf{x}_q) \right\}. \tag{12}$$

This process continues until the existing neighborhood sets incorporate all objects. The representative objects in all neighborhood sets are denoted as $NR = \{\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_s\}$, where $\mathbf{z}_g$ is a representative object in g-th neighborhood set. The proposed neighborhood rough set-based micro partition is summarized in Algorithm 1. Another critical issue that will be addressed below is how to merge the obtained neighborhood set $N^\xi(\mathbf{x}_i)$ by the merging mechanism.

Hierarchical merging mechanism

Based on the neighborhood rough set-based micro partition proposed in Section "namerefsc:NRSMP", we introduce the hierarchical mechanism to merge micro partitions appropriately. We iteratively update neighborhood sets $N^\xi(\mathbf{x}_i)$ and dataset X at each layer, following these two steps: 1) fix X,

Algorithm 1 NRSMP: Neighborhood Rough Set-based Micro Partition

Input: Dataset X , distance metric D.
Output: Neighborhood sets $N^{\xi}(\mathbf{x}_i)$, neighborhood representative object set NR.

1: **for** $i = 1$ to n **do**
2: Compute density ρ_i for $\mathbf{x}_i$ according to Eq. (11);
3: **end for**
4: **for** $i = 1$ to n **do**
5: Compute density gap ξ_i for $\mathbf{x}_i$ according to Eq. (10);
6: **end for**
7: **for** $i = 1$ to n **do**
8: **if** $\rho_i > 0$ **then**
9: Add $\mathbf{x}_i$ as representative object to NR;
10: **end if**
11: Compute $N^{\xi}(\mathbf{x}_i)$ according to Eq. (12);
12: **end for**

Algorithm 2 AMPHM: Adaptive Micro Partition and Hierarchical Merging Clustering Algorithm

Input: Dataset X, number of clusters k.
Output: Partition $\mathbf{U}$.

1: Set neighborhood sets for each object;
2: **while** $|NR| > k$ **do**
3: Compute D according to Eq. (5);
4: Compute $N^{\xi}(\mathbf{x}_i)$ and NR by Algorithm 1;
5: Update $X = NR$;
6: **end while**
7: Update representative object set $R = NR$;
8: Compute $\mathbf{U}$ according to Eq. (3).

compute $N^{\xi}(\mathbf{x}_i)$ and representative object set NR by Algorithm 1, and 2) fix NR, update X by the representative object set NR, i.e., $X = NR$. These two steps repeat until the number of neighborhood representative objects m is equal to the number of clusters k. A more specific explanation will be shown in Section "Clustering algorithm" below.

Clustering algorithm

With the neighborhood rough set-based micro partition in Section "namerefsc:NRSMP" proposed based on the heterogeneous attribute distance metric in Section "HADM: heterogeneous attribute distance metric" and the merging mechanism introduced in Section "Hierarchical merging mechanism", we propose the overall clustering algorithm. The whole clustering algorithm named Adaptive Micro Partition and Hierarchical Merging (AMPHM) is summarized in Algorithm 2.

AMPHM first partitions data into micro clusters, obtaining the neighborhood representative object set NR, then uses NR as the next layer dataset X to further partition and merge the NR of the previous layer while maintaining the overall cluster result in each layer. This algorithm continues until the number of neighborhood representative objects m is equal to the number of clusters k. Since our proposed approach

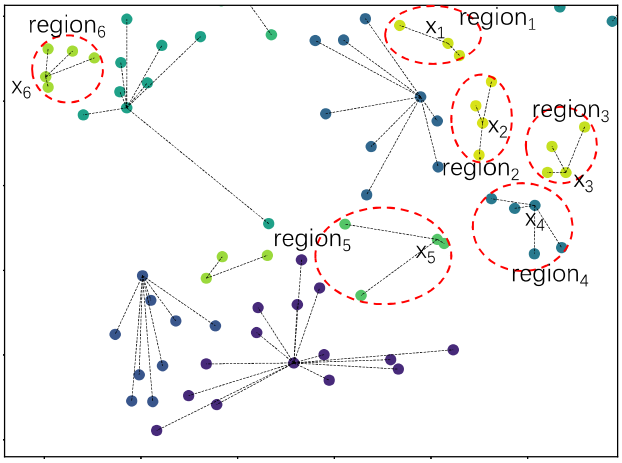


Fig. 2 Hierarchical merging with the micro sets in the form of the region. The regions are marked by red dashed circles, and different regions are partitioned by black solid lines. The points where black lines meet in each region are representative objects, with merging limited by density gap

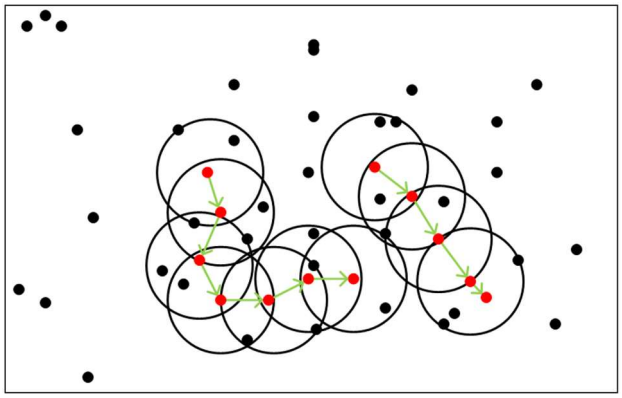


Fig. 3 In hierarchical clustering, data points or clusters are merged in each layer. The most common approach is one-by-one pairwise merging. In contrast to AMPHM, which enables the combination of multiple clusters simultaneously, the process of merging in this case involves the combination of individual clusters one at a time

shares similarities with hierarchical clustering, Figs. 2 and 3 illustrate their partial clustering processes to highlight distinctions between the two methods.

Theorem 3 *The time complexity of the AMPHM clustering algorithm is* $O(d_c^2 n + n \log n)$ *at any round.*

Proof For the worst-case analysis, we assume that all the attributes are categorical, i.e., $d_c = |A|$, and $v = max(v^1, v^2, ..., v^{d_c})$. The distances D correspond to attributes in A and are updated previously before each time we partition the neighborhood set N^{ξ}. Additionally, the time complexity at each round is the same. Therefore, we analyze the complexity of updating D and N^{ξ} once, respectively.

To update D requires obtaining $d_c \times d_c$ pairs of CPDs by scanning the n objects in X once, resulting in a time complexity of $O(nd_c^2)$. To compute the distances between a pair of

intra-attribute possible values using EMD in Eq. (7), it would take $O(v)$ time for each attribute. To calculate the weights for each pair of attributes, based on Eq. (8), it would require $v(v-1)/2$ calculations, resulting in $O(v^2)$ complexity for each attribute. For all the d_c attributes, we need to prepare weights between d_c pairs of attributes, and each attribute has $v(v-1)/2$ distances to compute. Therefore, the complexity for acquiring D is $O(nd^2 + v^3 d_c^2 + v^2 d_c^2)$, which can be simplified to $O(nd^2 + v^3 d_c^2)$.

To compute N^{ξ}, we need to parallelly compute an $n \times n$ distance matrix and sort all its rows in ascending order, which takes $O(n + n \log n)$ time. Then, for all the n objects, each one considers the remaining $n-1$ ones to identify its k_i value for obtaining the density ρ_i in Eq. (11), with a $O(n)$ complexity. To determine n density gaps for all objects, we need to examine the remaining $n-1$ objects for each one to identify the object with the highest density and the shortest distance. This process still has a complexity of $O(n)$. Next, the n density gaps need to be sorted in descending order, which requires $O(n \log n)$ time. Following that, we can create up to n neighborhood sets, where each set considers at most k_i objects based on the already sorted distances, resulting in a complexity of $O(nk_i)$. Therefore, the total complexity for calculating N^{ξ} is $O(n + n \log n + nlogn + nk_i)$, which simplifies down to $O(n \log n)$.

Thus, the overall time complexity of AMPHM at each round is $O(nd^2 + v^3 d_c^2 + n \log n)$. As v is a very small constant, typically with values in the interval $[1, 6]$, satisfying $v^3 \ll n$ for almost all real datasets, the final time complexity of AMPHM is $O(d_c^2 n + n \log n)$. □

Note that the computation cost of Algorithm 2 during the AMPHM process will drop significantly after updating the dataset X to NR in the first round.

Experiments

To evaluate the performance of AMPHM, three experiments have been conducted. (1) Clustering Performance Evaluation: AMPHM was compared with existing methods on 9 real datasets using two validity indices to evaluate the clustering performance; (2) Ablation Study: Various ablated versions of AMPHM are compared to prove its effectiveness from different aspects; and (3) Visualization: The cluster discrimination ability of the proposed method is compared with various existing counterparts.

Experimental settings

In the experiments, we compare the proposed AMPHM with ten other methods. These include eight clustering approaches from existing literature and two variants of AMPHM. Five counterparts proposed in recent years, including Jia's Distance Metric (JDM) [15], Coupled Metric Similarity (CMS) [26], Entropy-based Distance Metric (EDM) [16], Unified Distance Metric (UDM) [18] combined with k-modes (KMD) [50] and k-prototypes (KPT) [5], according to the attribute composition of datasets, while Object-Cluster Iterative Learning (OCIL) [20] designed for datasets that contain both numerical attributes and categorical attributes been selected, where JDM, CMS, EDM, and UDM are the state-of-the-arts. We also select three conventional clustering algorithms for comparison, i.e., w-k-means clustering algorithm (WKM) [12], which are representative algorithms in the attribute-weighting clustering stream, and original versions of KMD and KPT. We configured the parameters of the aforementioned methods to align with the values recommended in their respective papers. Additionally, we created two variations of AMPHM, denoted AMPHM$^{\mathrm{I}}$ and AMPHM$^{\mathrm{II}}$, to conduct ablation studies. More information about these two variants is available in Section "Ablation study".

Two validity indices are utilized to evaluate the clustering performance. CA [51] is a standard metric that calculates the match rate by determining the optimal permutation mapping between the identified clusters and their corresponding true labels. Adjusted Rand Index (ARI) [52–54] is an advanced validity index that is more discriminative in quantifying the performance of different counterparts. More specifically, ARI is an enhanced version of the Rand Index (RI) that mitigates issues related to random labeling. The RI measures the similarity between the clusters obtained by the algorithm and the actual data labels. It is calculated based on the proportion of object pairs that are assigned to the same or different clusters in both the predicted and actual clusters. CA falls within the range of [0, 1] and ARI has a range of values from $[-1, 1]$. All the selected validity indices show better clustering performance with higher values.

We use real-world datasets from various domains, including medicine, biology, and sociology, to conduct comprehensive experiments. All the datasets are publicly available and commonly used. The statistics for these datasets are summarized in Table 4. Datasets 1 to 7 are publicly available datasets obtained from the University of California, Irvine (UCI) machine learning repository.[4] Datasets 8, 9 collected from the Weka website.[5] All the datasets are preprocessed by removing objects with missing values. All the experiments are performed in the same computer with an Apple M1 CPU and 8GB RAM and are programmed using Python 3.9.6 and MATLAB 2021B.

Clustering performance evaluation

The clustering performance has been evaluated using the CA and ARI indexes, with the results presented in Tables 5 and 6,

[4] https://archive.ics.uci.edu/datasets.

[5] https://waikato.github.io/weka-wiki/datasets/.

Table 4 Statistics of the 9 Utilized Datasets. d_c, d_u, and n are the numbers of categorical, numerical attributes, and data objects, respectively. k^* is the true number of clusters and we set $k = k^*$ in the experiments

No.	Dataset	Abbrev	d_c	d_u	n	k^*
1	Dermatology	DT	33	1	366	6
2	Autism-adolescent	AA	2	7	104	2
3	Common Toad	CT	12	2	189	2
4	Hayes-roth	HR	4	0	132	3
5	Breast cancer	BC	9	0	286	2
6	Lymphography	LG	18	0	148	4
7	Congressional voting	VT	16	0	435	2
8	Employee selection	ES	4	0	488	9
9	Social workers	SW	10	0	1000	4

respectively. The best and second-best outcomes for each dataset are highlighted in **boldface** and underline, respectively. To maintain compactness, the results of KMD for categorical data and KPT for heterogeneous attribute data are combined in the same column. The "Ave. Rank" row in Tables 5 and 6 computes the average rank of the performance of the counterparts on all datasets.

The notable observations are: (1) AMPHM performs the best on most datasets in general, indicating its competitive clustering performance. (2) Although AMPHM does not perform the best on CT, VT, and ES datasets in the CA index, the best one differs on these datasets while AMPHM always outperforms them on other datasets, which also demonstrates that AMPHM is highly competitive. (3) On CT, LG, and VT datasets, AMPHM does not perform the best in terms of the ARI index. The reason could be that many attribute values of these datasets concentrate on two sides of the possible values. For such attribute values, AMPHM only has almost one intra-attribute to be weighted during clustering, which makes AMPHM downgrade into a conventional attribute weighting approach. As the ES dataset has few data objects in each class, AMPHM may not have sufficient information to form enough micro clusters and distinguish the compact clusters thus data objects have less ability to reject to be merged into dissimilar objects. The reason why AMPHM has competitive clustering performance compared to other counterparts is that our proposed heterogeneous attribute distance metric enables AMPHM to appropriately exploit distance structure information collaboratively represented by the heterogeneous attributes, and thus partition objects into prominent compact clusters, which are separable to achieve better clustering accuracy. Thus, it can be concluded that our AMPHM is very competitive in terms of clustering accuracy.

Table 5 Clustering performance evaluated by CA index. The average performance rank is reported in the Ave. Rank row

Dataset	KMD/KPT	WKM	OCIL	JDM	CMS	EDM	UDM	AMPHM
DT	0.5538±0.10	0.6234±0.09	0.6750±0.10	0.6654±0.10	0.6017±0.14	0.5872±0.10	0.6845±0.11	**0.7677±0.00**
AA	0.5301±0.03	0.5254±0.02	0.5186±0.03	0.5788±0.05	0.5244±0.03	0.5577±0.03	0.5579±0.02	**0.5962±0.00**
CT	0.5298±0.02	0.5234±0.03	0.5056±0.00	0.5224±0.02	0.5259±0.02	0.5365±0.03	**0.5781±0.02**	0.5450±0.00
HR	0.3638±0.01	0.4078±0.05	0.3761±0.04	0.3753±0.02	0.4048±0.04	0.4070±0.03	0.4040±0.03	**0.4167±0.00**
BC	0.5187±0.02	0.5835±0.09	0.5414±0.06	0.5823±0.10	0.5281±0.04	0.5304±0.02	0.5685±0.19	**0.7657±0.00**
LG	0.4531±0.04	0.4385±0.05	0.5003±0.04	0.4730±0.04	0.4189±0.05	0.4523±0.04	0.4704±0.04	**0.5608±0.00**
VT	0.8643±0.00	0.8570±0.07	**0.8810±0.00**	0.8681±0.00	0.8649±0.01	0.8317±0.10	0.8721±0.00	0.8736±0.00
ES	0.3674±0.03	0.3750±0.04	0.3839±0.04	0.3505±0.03	**0.4016±0.04**	0.3664±0.02	0.3681±0.03	0.3934±0.00
SW	0.3924±0.03	0.3750±0.03	0.3732±0.03	0.3337±0.03	0.3310±0.03	0.3321±0.01	0.3744±0.03	**0.4350±0.00**
Ave. Rank	5.7778	4.6667	4.5556	4.8889	5.6667	5.6667	3.4444	1.3333

Table 6 Clustering performance evaluated by ARI index. The average performance rank is reported in the Ave. Rank row

Dataset	KMD/KPT	WKM	OCIL	JDM	CMS	EDM	UDM	AMPHM
DT	0.4222±0.12	0.5091±0.09	0.6063±0.10	0.6140±0.13	0.5176±0.17	0.4393±0.12	0.6266±0.15	**0.6776±0.00**
AA	−0.0031±0.01	−0.0055±0.01	−0.0073±0.01	0.0182±0.03	−0.0086±0.01	0.0063±0.01	−0.0153±0.01	**0.0185±0.00**
CT	−0.0081±0.01	−0.0079±0.01	−0.0208±0.00	−0.0142±0.01	−0.0075±0.01	0.0017±0.01	**0.0133±0.02**	−0.0010±0.00
HR	−0.0121±0.00	0.0069±0.02	−0.0035±0.02	−0.0063±0.01	0.0075±0.02	0.0080±0.02	0.0070±0.02	**0.0094±0.00**
BC	−0.0041±0.00	0.0396±0.07	0.0112±0.04	0.0405±0.07	0.0029±0.02	0.0066±0.01	0.0620±0.02	**0.1092±0.00**
LG	0.1128±0.04	0.0845±0.04	**0.1817±0.05**	0.1231±0.04	0.0884±0.04	0.0893±0.03	0.1317±0.05	0.1293±0.00
VT	0.5297±0.00	0.5273±0.11	**0.5795±0.00**	0.5411±0.01	0.5319±0.03	0.4776±0.17	0.5527±0.01	0.5568±0.00
ES	0.1621±0.02	0.1717±0.03	0.1929±0.02	0.1666±0.02	**0.2115±0.03**	0.1627±0.04	0.2106±0.02	0.1725±0.00
SW	0.0571±0.02	0.0456±0.02	0.0516±0.02	0.0519±0.01	0.0505±0.02	0.0591±0.01	0.0760±0.01	**0.0847±0.00**
Ave. Rank	6.3333	5.8889	4.4444	4.5556	5.1111	4.8889	2.8889	1.8889

Ablation study

Ablation studies were carried out, and the clustering performance was assessed using the ARI index. To evaluate the effectiveness of the heterogeneous attribute dissimilarity metric, we modified AMPHM to employ only the conventional Euclidean distance and Hamming distance, creating a variant AMPHM$^{\mathrm{I}}$. To evaluate the effectiveness of the hierarchical merging mechanism, we conducted a comparison between AMPHM and a variation called AMPHM$^{\mathrm{II}}$, which incorporates the prototype selection strategy from the typically k-prototypes approach to cluster the representative objects after the initial formation of neighborhood sets. The clustering performance and the Ave. Rank of AMPHM and its two variations are shown in Fig. 4.

It can be observed from Fig. 4 that AMPHM outperforms its two variations, indicating the effectiveness of the heterogeneous attribute distance metric and hierarchical merging mechanism. More specifically, AMPHM outperforms the AMPHM$^{\mathrm{I}}$ on all the nine datasets. This illustrates that the heterogeneous attribute distance metric $D(\cdot,\cdot)$ proposed in Sect. 3.2 can sufficiently harness the original data information for different data distributions, which testifies to the effectiveness of our distance metric. Moreover, AMPHM outperforms AMPHM$^{\mathrm{II}}$ on eight datasets and AMPHM$^{\mathrm{I}}$ performs better than AMPHM$^{\mathrm{II}}$ on six datasets. This shows that the hierarchical merging mechanism effectively merges micro clusters into compact clusters, indicating the competitiveness of our proposed merging mechanism compared to the conventional k-means-type clustering process. The reason why AMPHM$^{\mathrm{I}}$ performs worse than AMPHM$^{\mathrm{II}}$ in some datasets (i.e. CT, LG, and VT) would be that the former adopts the simplest Euclidean and Hamming distances.

Visualization

In Fig. 5, t-SNE [55] is utilized to illustrate the cluster discrimination ability of AMPHM. We first cluster the dataset DT using KPT, WKM, JDM, CMS, UDM, and AMPHM. Then, data is processed by t-SNE into two-dimensional according to the distance matrix of data objects generated by the distance metrics adopted by the compared methods. It is visualized with data points marked in green for those that cluster correctly and red for those that cluster incorrectly, while different clusters are marked in different markers. Intuitively, the fewer red data points indicate a better clustering accuracy. We can observe from Fig. 5 that AMPHM has significantly fewer red points than other counterparts, which indicates the effectiveness of AMPHM. More specifically, by using NRSMP proposed in Sect. "NRSMP: neighborhood rough set-based micro partition" to partition micro clusters, AMPHM can collaboratively distinguish dissimilar objects. As a result, the merging mechanism appropriately merges

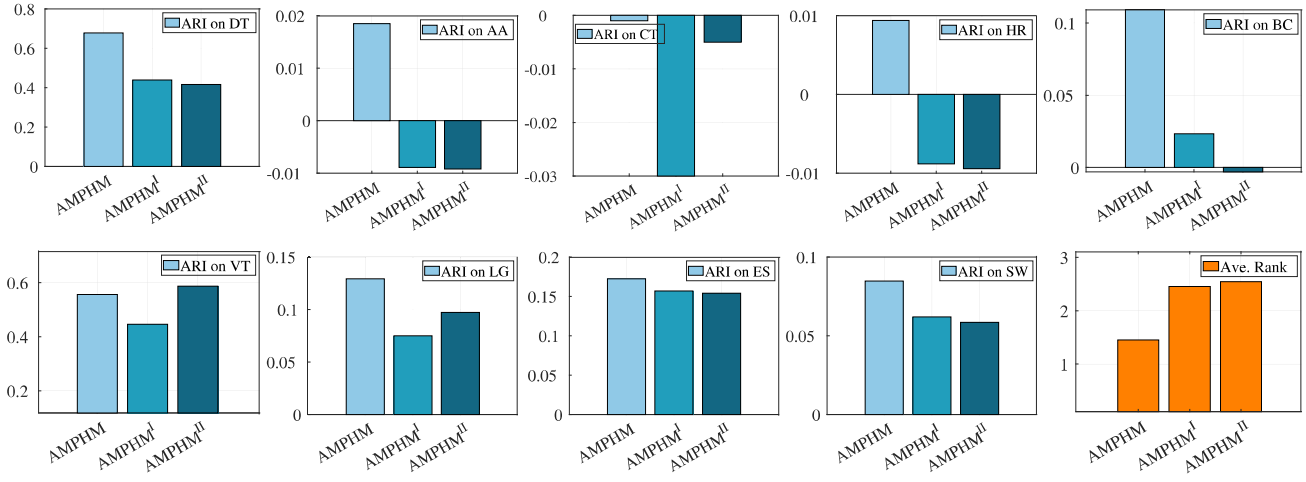


Fig. 4 Clustering performance and Ave. rank of AMPHM, AMPHM$^{\mathrm{I}}$ and AMPHM$^{\mathrm{II}}$ on all the 9 datasets, where a better measure yields a higher value

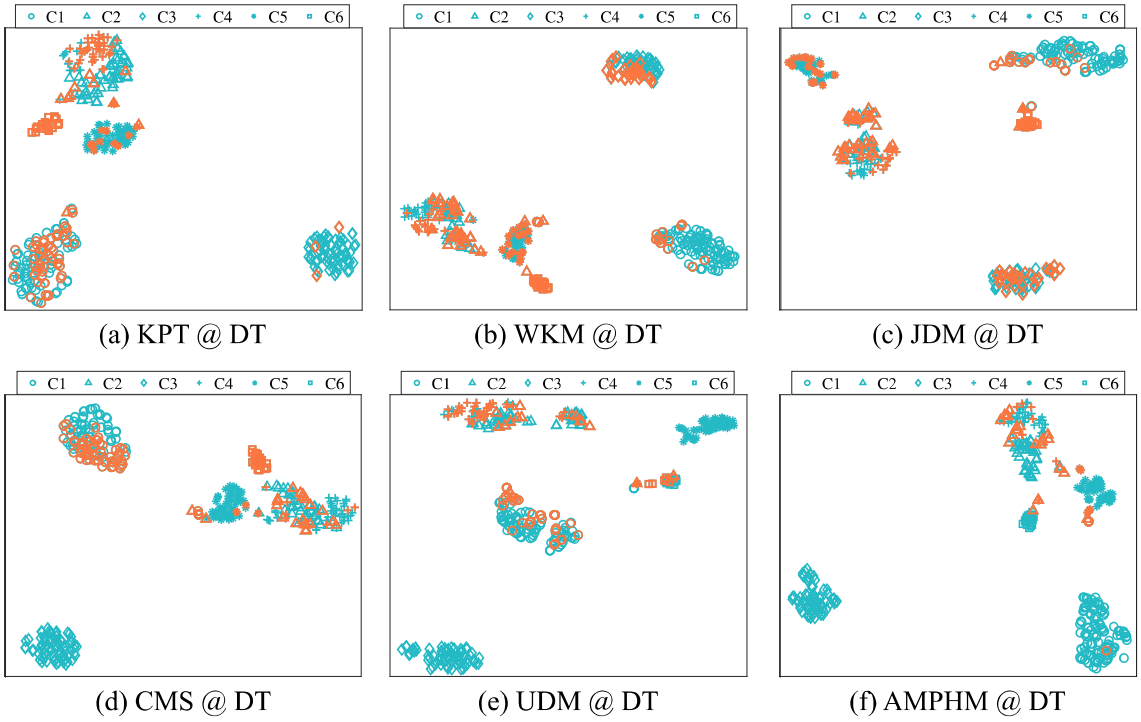


Fig. 5 t-SNE visualization of the DT datasets after KPT (**a**), WKM (**b**), JDM (**c**), CMS (**d**), UDM (**e**), and AMPHM (**f**) clustering, where objects are marked in different markers indicate their ground truth labels, while data points marked in red for those that cluster incorrectly

similar micro clusters and finally forms the final "true" clusters, indicating stronger cluster discrimination compared to other methods.

Conclusion

In this paper, we have proposed a novel clustering algorithm called Adaptive Micro Partition and Hierarchical Merging (AMPHM) for heterogeneous attribute data clustering. AMPHM mainly solved the two challenges in clustering real datasets, i.e., most existing clustering approaches relying on certain assumptions or models, and the clustering limitation on heterogeneous attributes, by adopting (1) a unified distance metric to quantify distances represented by heterogeneous attributes, (2) a micro partition mechanism based on neighborhood set to appropriately partition data objects into many compact micro clusters, and (3) a merging mechanism to hierarchically merge micro clusters. We simultaneously address the bias brought by prior clustering knowledge and attribute heterogeneity, which is far more challenging than coping with only one of them. The effectiveness of AMPHM

is illustrated by the comprehensive experiments including comparison with state-of-the-art counterparts and ablation study on various public datasets. Moreover, AMPHM is accurate and interpretable. Our future research will focus on the analysis of heterogeneous attribute data with more challenging issues, e.g., clustering mixed data under dynamic or federated environments.

Acknowledgements This work was supported in part by the National [redacted]

Data availability The datasets used in all the experiments of this paper are publicly available. The links of the source repositories are provided in Sect. "Experimental settings".

References

1. Hongshi X, Ma C, Lian J, Kui X, Chaima E (2018) Urban flooding risk assessment based on an integrated k-means cluster algorithm and improved entropy weight method in the region of Haikou. Chin J Hydrol 563:975–986. https://doi.org/10.1016/j.jhydrol.2018.06.060
2. Alan A (2002) Categorical data analysis. Wiley Series in Probability and Statistics. Wiley. https://doi.org/10.1002/0471249688
3. Ikotun AM, Ezugwu AE, Abualigah L, Abuhaija B, Heming J (2023) K-means clustering algorithms: a comprehensive review, variants analysis, and advances in the era of big data. Inf Sci 622:178–210. https://doi.org/10.1016/j.ins.2022.11.139
4. Arabie P, Baier Newark D, Critchley Cottbus F, Keynes M (2006) Studies in classification, data analysis, and knowledge organization. Springer, New York
5. Huang Z (1997) Clustering large data sets with mixed numeric and categorical values. In: Proceedings of the 1st Pacific-Asia conference on knowledge discovery and data mining, pp 21–34
6. Mohamed ABHK, Chiheb-Eddine BN, Nadia E (2015) MapReduce-based k-prototypes clustering method for big data. In: Proceedings of the IEEE international conference on data science and advanced analytics, pages 1–7. IEEE. https://doi.org/10.1109/DSAA.2015.7344894
7. Qian Y, Li F, Liang J, Liu B, Dang C (2016) Space structure and clustering of categorical data. IEEE Trans Neural Netw Learn Syst 27(10):2047–2059. https://doi.org/10.1109/TNNLS.2015.2451151
8. Jian S, Pang G, Cao L, Kai L, Gao H (2019) CURE: flexible categorical data representation by hierarchical coupling learning. IEEE Trans Knowl Data Eng 31(5):853–866. https://doi.org/10.1109/TKDE.2018.2848902
9. Zhu C, Cao L, Yin J (2022) Unsupervised heterogeneous coupling learning for categorical representation. IEEE Trans Pattern Anal Mach Intell 44(1):533–549. https://doi.org/10.1109/TPAMI.2020.3010953
10. Lin D (1998) An Information-Theoretic Definition of Similarity. In: Proceedings of the 15th International Conference on Machine Learning, pages 296–304
11. Le SQ, Ho TB (2005) An association-based dissimilarity measure for categorical data. Pattern Recogn Lett 26(16):2549–2557. https://doi.org/10.1016/j.patrec.2005.06.002
12. Huang JZ, Ng MK, Rong H, Li Z (2005) Automated variable weighting in k-means type clustering. IEEE Trans Pattern Anal Mach Intell 27(5):657–668. https://doi.org/10.1109/TPAMI.2005.95
13. Ienco D, Pensa Ruggero G, Meo R (2009) Context-Based Distance Learning for Categorical Data Clustering. In: Advances in Intelligent Data Analysis VIII 83–94. https://doi.org/10.1007/978-3-642-03915-7_8
14. Ienco D, Pensa RG, Meo R (2012) From context to distance. ACM Trans Knowl Discov Data 6(1):1–25. https://doi.org/10.1145/2133360.2133361
15. Jia H, Cheung Y, Liu J (2016) A new distance metric for unsupervised learning of categorical data. IEEE Trans Neural Netw Learn Syst 27(5):1065–1079. https://doi.org/10.1109/TNNLS.2015.2436432
16. Zhang Y, Cheung Y-M, Tan KC (2020) A unified entropy-based distance metric for ordinal-and-nominal-attribute data clustering. IEEE Trans Neural Netw Learn Syst 31(1):39–52. https://doi.org/10.1109/TNNLS.2019.2899381
17. Zhang Y, Cheung Y (2020) An ordinal data clustering algorithm with automated distance learning. In: Proceedings of the AAAI Conference on Artificial Intelligence 34:6869–6876. https://doi.org/10.1609/AAAI.V34I04.6168
18. Zhang Y, Cheung Y-M (2022) A new distance metric exploiting heterogeneous interattribute relationship for ordinal-and-nominal-attribute data clustering. IEEE Trans Cybern 52(2):758–771. https://doi.org/10.1109/TCYB.2020.2983073
19. Zhang Y, Cheung Y (2022) Learnable weighting of intra-attribute distances for categorical data clustering with nominal and ordinal attributes. IEEE Trans Pattern Anal Mach Intell 44(7):3560–3576. https://doi.org/10.1109/TPAMI.2021.3056510
20. Cheung Y, Jia H (2013) Categorical-and-numerical-attribute data clustering based on a unified similarity metric without knowing cluster number. Pattern Recogn 46(8):2228–2238. https://doi.org/10.1016/j.patcog.2013.01.027
21. Zhu C, Zhang Q, Cao L, Abrahamyan A (2020) Mix2Vec: unsupervised mixed data representation. In: Proceedings of the IEEE 7th International Conference on Data Science and Advanced Analytics, pp 118–127. IEEE. https://doi.org/10.1109/DSAA49011.2020.00024
22. Zhang Y, Cheung Y, Zeng A (2022) Het2Hom: representation of heterogeneous attributes into homogeneous concept spaces for categorical-and-numerical-attribute data clustering. In: Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, pages 3758–3765, California. https://doi.org/10.24963/ijcai.2022/522
23. Jain AK, Murty MN, Flynn PJ (1999) Data clustering: a review. ACM Comput Surv 31(3):264–323. https://doi.org/10.1145/331499.331504

24. Alamuri M, Surampudi BR, Negi A (2014) A survey of distance/similarity measures for categorical data. In: Proceedings of the International Joint Conference on Neural Networks, pages 1907–1914. IEEE. https://doi.org/10.1109/IJCNN.2014.6889941
25. Ahmad A, Dey L (2007) A method to compute distance between two categorical values of same attribute in unsupervised learning for categorical data set. Pattern Recogn Lett 28(1):110–118. https://doi.org/10.1016/j.patrec.2006.06.006
26. Jian S, Cao L, Kai L, Gao H (2018) Unsupervised Coupled Metric Similarity for Non-IID Categorical Data. IEEE Trans Knowl Data Eng 30(9):1810–1823. https://doi.org/10.1109/TKDE.2018.2808532
27. Chen L, Guo Q, Liu Z, Zhang S, Zhang H (2021) Enhanced synchronization-inspired clustering for high-dimensional data. Complex Intell Syst 7(1):203–223. https://doi.org/10.1007/s40747-020-00191-y
28. Zhang Y, Cheung Y (2018) Exploiting order information embedded in ordered categories for ordinal data clustering. In: Proceedings of the 24th International Symposium on Methodologies for Intelligent Systems, pp. 247–257. https://doi.org/10.1007/978-3-030-01851-1_24
29. Wang P, Zhang Y, Zhang Y, Lu Y, Li M, Cheung Y (2024) Clustering by learning the ordinal relationships of qualitative attribute values. In: Proceedings of the IEEE 34th International Joint Conference on Neural Networks, pp 1–8
30. Zhang Y, Luo X, Chen Q, Zou R, Zhang Y, Cheung Y (2024) Towards unbiased minimal cluster analysis of categorical-and-numerical attribute data. In: Proceedings of the 28th International Conference on Pattern Recognition, pages 1–16
31. Sangma JW, Sarkar M, Pal V, Agrawal A, Yogita (2022) Hierarchical clustering for multiple nominal data streams with evolving behaviour. Complex Intell Syst 8(2):1737–1761. https://doi.org/10.1007/s40747-021-00634-0
32. Zhao M, Feng S, Zhang Y, Mengkei L, Lu Y, Cheung Y (2024) Learning order forest for qualitative-attribute data clustering. In: Proceedings of the 27th European Conference on Artificial Intelligence, pages 1–8
33. Peng Z, Song X, Song S, Stojanovic V (2023) Hysteresis quantified control for switched reaction–diffusion systems and its application. Complex Intell Syst 9(6):7451–7460. https://doi.org/10.1007/s40747-023-01135-y
34. Sun P, Song X, Song S, Stojanovic V (2023) Composite adaptive finite-time fuzzy control for switched nonlinear systems with preassigned performance. Int J Adapt Control Signal Process 37(3):771–789. https://doi.org/10.1002/acs.3546
35. Zhang Z, Song X, Sun X, Stojanovic V (2023) Hybrid-driven-based fuzzy secure filtering for nonlinear parabolic partial differential equation systems with cyber attacks. Int J Adapt Control Signal Process 37(2):380–398. https://doi.org/10.1002/acs.3529
36. Abdel-Basset M, Abouhawwash M, Askar S, Hawash H, Nayyar A (2023) A system of the internet of nano-dental things
37. Shenghong C, Yiqun Z, Xiaopeng L, Yiu-ming C, Hong J, Peng L (2024) Robust categorical data clustering guided by multi-granular competitive learning. In: Proceedings of the IEEE 44th International Conference on Distributed Computing Systems, pp. 288–299
38. Zou R, Zhang Y, Zhang Y, Lu Y, Li M, Cheung Y (2024) Federated clustering with unknown number of clusters. In: Proceedings of the IEEE 6th International Conference on Data-driven Optimization of Complex Systems, pp. 671–677
39. Chen J, Ji Y, Zou R, Zhang Y, Cheung Y (2024) QGRL: quaternion graph representation learning for heterogeneous feature data clustering. In: Proceedings of the 30th SIGKDD Conference on Knowledge Discovery and Data Mining, pages 1–10
40. Mohamed M (2023) Agricultural Sustainability in the Age of Deep Learning: Current Trends, Challenges, and Future Trajectories. Sustain Mach Intell J 4. https://doi.org/10.61185/SMIJ.2023.44102
41. Mohamed M (2023) Empowering deep learning based organizational decision making: a survey. Sustain Mach Intell J 3. https://doi.org/10.61185/SMIJ.2023.33105
42. AlNuaimi N, Masud MM, Serhani MA, Zaki N (2022) Streaming feature selection algorithms for big data: a survey. Appl Comput Inf 18(1/2):113–135. https://doi.org/10.1016/j.aci.2019.01.001
43. Eskandari S, Javidi MM (2016) Online streaming feature selection using rough sets. Int J Approx Reason 69:35–57. https://doi.org/10.1016/j.ijar.2015.11.006
44. Mohammad Masoud Javidi and Sadegh Eskandari (2019) Online streaming feature selection: a minimum redundancy, maximum significance approach. Pattern Anal Appl 22(3):949–963. https://doi.org/10.1007/s10044-018-0690-7
45. Saúl Solorio-Fernández J, Carrasco-Ochoa A, Martínez-Trinidad JF (2022) A survey on feature selection methods for mixed data. Artif Intell Rev 55(4):2821–2846. https://doi.org/10.1007/s10462-021-10072-6
46. Zhang P, Li T, Yuan Z, Luo C, Liu K, Yang X (2022) Heterogeneous Feature Selection Based on Neighborhood Combination Entropy. IEEE Trans Neural Netw Learn Syst, pp. 1–14. https://doi.org/10.1109/TNNLS.2022.3193929
47. Rubner Y, Tomasi C, Guibas LJ (2000) The Earth Mover's Distance as a Metric for Image Retrieval. Int J Comput Vision 40(2):99. https://doi.org/10.1023/A:1026543900054
48. Jia X, Bin Lei YG, Winslett M, Ge Yu, Zhang Z (2015) Efficient Similarity Join Based on Earth Mover's Distance Using MapReduce. IEEE Trans Knowl Data Eng 27(8):2148–2162. https://doi.org/10.1109/TKDE.2015.2411281
49. Zhang Y, Cheung Y-M (2023) Graph-based dissimilarity measurement for cluster analysis of any-type-attributed data. IEEE Trans Neural Netw Learn Syst 34(9):6530–6544. https://doi.org/10.1109/TNNLS.2022.3202700
50. Huang Z (1998) Extensions to the k-means algorithm for clustering large data sets with categorical values. Data Min Knowl Disc 2(3):283–304. https://doi.org/10.1023/A:1009769707641
51. He Xiaofei, Cai Deng, Niyogi Partha (2005) Laplacian Score for Feature Selection. In *Advances in Neural Information Processing Systems*, pages 507–514
52. Santos Jorge M, Mark Embrechts (2009) On the use of the adjusted rand index as a metric for evaluating supervised classification. In: Proceedings of the 19th International conference on artificial neural networks, pages 175–184. https://doi.org/10.1007/978-3-642-04277-5_18
53. Rand WM (1971) Objective Criteria for the Evaluation of Clustering Methods. J Am Stat Assoc 66(336):846–850
54. Gates AJ, Ahn YY (2017) The impact of random models on clustering similarity. J Mach Learn Res 18:3049–3076. https://doi.org/10.5555/3122009.3176831
55. van der Maaten L, Hinton G (2008) Visualizing data using t-SNE. J Mach Learn Res 9(86):2579–2605

Received: Added at production | Revised: Added at production | Accepted: Added at production

DOI: xxx/xxxx

ORIGINAL ARTICLE

Towards Clustering of Incomplete Mixed-Attribute Data

[redacted] Xinxi Chen | Zexi Tan [redacted]

[redacted]

Correspondence

[redacted]

Abstract

Clustering analysis is one of the most important data mining and knowledge discovery tools in real applications. However, due to the widespread presence of missing values in real-world datasets, the effectiveness of most clustering algorithms is restricted as they typically require complete datasets as input. Missing values imputation is usually performed as data pre-processing under such circumstances. However, for the datasets composed of both numerical and categorical attributes (also known as mixed-attribute datasets), most existing imputation methods suffer from the following three limitations: 1) Only feasible for a certain type of attribute; 2) Encounter difficulties in considering the inter-dependence between different types of attributes; 3) Overlook the information provided by the incomplete objects. These limitations can lead to the loss of valuable information, thereby failing to restore the original data distribution, and thus mislead clustering algorithms. This paper therefore proposes a clustering-imputation co-learning method for incomplete mixed-attribute datasets to address these issues. This method integrates imputation and clustering into one learning process, emphasizing the interrelationships between mixed attributes during the imputation process and applying incomplete objects during the clustering process, ultimately producing a complete dataset and accurate clustering results. Experiments on various datasets validate that the proposed method has good imputation efficiency and can effectively improve clustering performance.

KEYWORDS

Clustering, missing values imputation, categorical attribute, mixed data, dissimilarity measure.

1 | INTRODUCTION

Clustering methods are widely applied in various research fields, such as biological research, clinical medicine, traffic forecasting, and environmental and meteorological studies, and most of them are designed for complete datasets. However, missing values are pervasive in real-world datasets[1 2 3]. There are many causes for the occurrence of missing values, including sample contamination, operational errors, machine malfunctions, and human mistakes[4 5 6 7]. Therefore, handling missing values becomes a crucial task before clustering analysis.

To handle missing values, data analysts generally have three options: 1) Directly delete[8] objects with missing values. This method is obviously only suitable for datasets with a few missing values, as deleting a large number of objects would result in significant information loss, making clustering analysis ineffective. 2) Use a constant (usually 0) to replace the missing values. This method can easily distort the original distribution of the data. 3) Estimate missing values, which is known as imputation[9]. The first two methods are likely to cause significant information loss and introduce errors, while the third method, which restores the dataset based on the original data, is considered a more reasonable and effective approach for handling missing values. However, the choice of an appropriate imputation method depends on the type of data[10 11 12]. Therefore, a universal method applicable to any type of attribute is still desired.

Missing values Imputation method (MI) is considered a reasonable and efficient way to handle missing values. Many successful methods have been developed for handling missing values in homogeneous datasets (where attributes are either numerical or categorical). Numerous studies focus on imputing numerical missing values, using methods such as Mean Substitution[13], Hot Deck-based MI[14], Logistic Regression-based MI[15], Expectation Maximization MI[16], *k*-Nearest-Neighbor-based MI[17], and

TABLE 1 An example of missing values imputation, with data from a subset of the Iris dataset. "?" is a missing value.

objects	speal_length	speal_width	petal_length	petal_width	species
x_1	5.1	3.5	1.4	0.2	setosa
x_2	4.9	3.0	1.4	0.2	setosax
x_3	7.0	3.2	4.7	?	versicolor
x_4	6.9	3.1	?	1.5	versicolor
x_5	6.9	3.2	5.7	2.3	virginica

Self-organizing Map-based MI[18]. In contrast, research on categorical data imputation is relatively sparse, with existing methods including Mode Substitution[13], C4.5-based MI[19], and Bayesian Principal Component Analysis-based MI[20]. Since various research fields continue to evolve, heterogeneous datasets are becoming increasingly common, rendering these methods designed for single-attribute type data unsuitable. Heterogeneous datasets, also known as mixed-attribute datasets, contain both numerical and categorical attributes. More specifically, categorical attributes can be divided into nominal and ordinal attributes, with ordinal attributes indicating a ranked order among values[21]. For example, in a medical dataset, Height (cm) descriptions like "150, 160, 170" are numerical; Gender descriptions like "male, female" are nominal; and Blood Pressure evaluations like "low, normal, high" are ordinal.

To utilize information from heterogeneous data, imputation methods should consider the intrinsic characteristics of different types of attributes and their interactions. Imputation methods for mixed attributes are rather scarce in the literature, the most well-known are Multivariate Imputation by Chained Equations (MICE)[22] and MissForest (MF)[23]. However, these methods treat ordinal data as nominal, losing valuable sequential information. Similarly, treating ordinal data as numerical is risky. For example, the ordinal attribute "Legs" from the ZOO dataset has five levels (0, 2, 4, 6, 8), using Mean Substitution would yield unreasonable results. Moreover, most imputation methods fill missing values based solely on complete objects, which is equivalent to deleting incomplete objects and ignoring the information contained in these incomplete objects. In other words, there is a significant difference in the utilization of effective information between ignoring an object with many missing values and one with few missing values. Furthermore, most methods utilize object class labels to impute missing values in a supervised scenario. However, since clustering is a commonly used exploratory tool in many research fields, these methods become inapplicable in an unsupervised scenario. The lack of labels in clustering data poses greater challenges for missing values imputation. Consequently, missing values are more likely to have a significant negative impact on downstream clustering tasks.

In many fields, humans remain the best problem solvers, making the application of human intuition to the issue of missing values particularly insightful. For instance, what should the missing values in Table 1 be estimated as? Here are three common thought processes humans use when predicting missing values: 1) We tend to believe that similar items group together. 2) We strive to utilize all available information, directly noticing the similarity between two objects (even if both are incomplete). 3) We infer the imputed values from the relationships between attributes. Therefore, we can observe that x_3 and x_4 are most similar, with each attribute value of x_3 being exactly 0.1 greater than x_4. Through this structural information, we can estimate the values of x_3 and x_4 as 1.6 and 4.6, respectively.

These human intuitions can precisely address the limitations mentioned above with imputation methods in mixed-attributed datasets. Inspired by this, we propose a method called Clustering Incomplete data with Missing value Imputation (CIMI). Firstly, for mixed-attribute data, we extend a dissimilarity measure based on Graph-Based Dissimilarity Measurement[24] to assess the distance between complete and incomplete objects. Secondly, we utilize this measure for clustering, integrating the imputation of missing values into the clustering process. Specifically, we cluster all the objects (including incomplete objects), identify the cluster to which the incomplete object belongs, and then use the cluster information to provide a reference for imputation. Indeed, we employ a progressive inference strategy, moving from certainty to uncertainty, to impute missing values. The proposed CIMI operates on an object-by-object basis, initially filling in the values we are most confident about. These completed values are then treated as confirmed data, which informs the imputation of subsequent values. Overall, the proposed CIMI leverages the human cognitive habit of grouping similar items together, utilizes all available information to notice similarities between incomplete objects, and infers imputed values from the relationships between attributes. By incorporating these intuitive approaches, CIMI effectively overcomes the limitations of existing imputation methods, solving the clustering and imputation problems simultaneously.

The main contributions of this paper are three-fold:

- A clustering method is introduced for incomplete heterogeneous datasets, establishing a complementary interaction between imputation and clustering to provide high imputation efficiency while achieving better cluster performance.
- A new imputation paradigm is proposed that allows incomplete objects to participate in the imputation process, overcoming the limitations of most imputation methods that ignore the valuable information provided by the incomplete objects, and thus provides richer hints for missing value imputation.
- We also analyze the relationship between imputation accuracy and clustering performance, offering an insightful hint for researchers to select suitable imputation methods in coping with certain clustering tasks.

The remainder of this paper is organized as follows: Section 2 introduces related works on handling missing values; Section 3 presents a detailed description of the proposed method; Section 4 and Section 5 report the experimental settings and results, respectively; And finally, a conclusion is drawn in Section 6.

2 | RELATED WORK

This section introduces common missing values imputation methods from the literature, which can be broadly categorized into two types: statistical-based methods and machine learning-based methods.

2.1 | Statistics-based missing values imputation

Among statistical methods, the most commonly used are Mean/median/mode Substitution (MS)[13] and Expectation-Maximization-based MI (EMMI)[16]. In the MS, missing values are replaced with the mean, median, or mode of the corresponding attribute in the complete dataset. EMMI estimates parameters iteratively between the E-step and M-step, maximizing the log-likelihood function of the complete objects. The primary advantages of these two methods are their ease of implementation and memory efficiency. However, they do not consider correlations between attributes, which may negatively impact imputation in certain applications, especially when the interrelationships between attributes are crucial. Additionally, various methods based on different theories are found in the literature, including the Local Least Square-based MI (LLSMI) and Bayesian Principal Component Analysis-based MI (BPCAMI). LLSMI estimates missing values through a linear combination of similar objects selected based on similarity measures[25]. Although simple and locally effective, its performance depends on the quality of the chosen similar objects. BPCAMI, on the other hand, estimates missing values through a combination of principal component regression, Bayesian estimation, and EM iteration, providing flexible and reliable imputation results despite its complexity and computational demands[20].

2.2 | Machine Learning-based missing values imputation

In recent years, machine learning-based methods for missing values imputation have gained significant popularity. Among these methods, k-Nearest-Neighbor MI (KNNMI) and k-Means Clustering imputation (KMCMI) are particularly notable. The core idea behind KNNMI is to use a distance metric to find the k nearest neighbors of an incomplete object, then replace the missing values with information from these neighbors[17]. This method is effective but its performance is highly dependent on data quality and sensitive to data scale, making it computationally expensive. Additionally, KNNMI has several variants, including iterative KNNMI[26], weighted KNNMI[27], and sequential KNNMI[28]. The KMCMI method involves two steps: first, forming clusters using k-means clustering, then handling missing values using cluster information[29]. When applying this type of method to actual datasets, it is necessary to consider that early imputation errors may propagate to further imputations. Fuzzy c-means MI provides a better solution for overlapping clusters, reflecting the complex nature of the data and reducing the risk of the algorithm getting trapped in local minima[30]. As another of standard imputation techniques, Random forest-based MI[31] preimpute the missing values, then grow the forest. Update the missing values using data proximity and iterate to improve results. The C4.5-based MI[32] handles missing values using a probability splitting method, which is efficient but may be unstable and memory-intensive as small changes in the data can significantly alter the decision tree structure. Rough set theory-based MI (RSTMI)[33] uses rough set rule induction to derive association rules of missing data patterns through approximation,

TABLE 2 Explanations of symbols.

Symbols	Explanations
x_i	Datasets objects
A^r	The r-th attribute of an object x_i
o_m^r	The m-th unique value of an attribute A^r
O^s	The unique value set of the attribute A^s
c_i	Cluster centroids
D	Attribute distance matrix
T and S	Incomplete objects set and the whole dataset
n and k	Number of objects and clusters

TABLE 3 An example of the description for the notation.

ID(x_i)	occupation (A^1)	credit(A^2)	Income(A^3)	...
Client1	doctor (o_1^1)	very-good(o_1^2)	16,000(o_1^3)	...
Client2	teacher(o_2^1)	good(o_2^2)	13,000(o_2^3)	...
Client3	driver(o_3^1)	neutral(o_3^2)	10,000(o_2^3)	...
Client4	driver(o_3^1)	?	10,000(o_2^3)	...
Client5	?	neutral(o_3^2)	?	...
Client6	?	?	16,000(o_1^3)	...
Client7	doctor (o_1^1)	?	?	...
Client8	?	?	?	...

dependency, and decision rules[34]. Association rule-based MI (ARMI)[35] describes dependency links between data entries in the dataset and fills in missing values according to these rules.

However, most imputation methods are designed for either numerical or categorical attributes alone. For instance, common methods developed for numerical attributes such as EMMI do not handle categorical attributes well. Other methods like C4.5MI, ARMI, and RSTMI are designed to only deal with categorical attributes, often discretizing numerical attributes before input, potentially losing valuable features of numerical data[36]. Thus, these single-attribute methods become unsuitable when dealing with the heterogeneous datasets often encountered in real-world applications. Methods for imputing missing values in heterogeneous data are relatively rare in the literature. Rubin first introduced multiple imputations[37]. Later, this study[38] proposes a maximum likelihood estimation method combining a multivariate normal model for continuous variables and a Poisson/multinomial model for categorical data. More sophisticated methods like MICE[22] create a complete dataset by repeatedly imputing missing values, relying on tuning parameters or specifying a parametric model. Choosing parameters without prior knowledge can be challenging. Another advanced method, MissForest[23], is based on random forest[39], reconstructing the missing values problem as a prediction task, imputing missing values by regressing all other attributes and using the fitted forest to predict the missing values. Tree-based[40] and random forest-based methods require integrating information from multiple trees, making them computationally complex and memory-intensive[41].

Previous studies have achieved successful imputation results, but they also have some limitations, as mentioned above. In summary, some methods are primarily designed for pure numerical data or pure categorical data, thus they cannot be directly applied to mixed data without any modifications. Moreover, most of these methods are designed to impute missing values using only information from complete objects, without considering incomplete objects. Consequently, their effectiveness is significantly reduced when facing a substantial amount of missing data in heterogeneous datasets. The following section defines the problem of missing values imputation on mixed-attribute dataset and then describes the proposed method.

3 | PROPOSED METHOD

3.1 | Preliminaries and problem formulation

The problem of clustering mixed-attribute dataset has been the subject of several prior studies[42,43]. Table. 2 lists the symbols used in the paper. A mixed-attribute dataset S can be represented as a tuple $S =< X, A, O >$, where $X = \{x_i | i = 1, 2, ..., n\}$ is the object set with n objects. The i-th object of X is a vector $x_i = [x_i^1, x_i^2, ..., x_i^d]$, which consists of d values from the d attributes. $A = \{A^r | r = 1, 2, ..., d\}$ is the attribute set composed of d attributes, including $d^{<n>}$ (nominal), $d^{<o>}$ (ordinal), and $d^{<u>}$ (numerical) attributes, where $d = d^{<n>} + d^{<o>} + d^{<u>}$. Corresponding to the attribute set A, $O = \{O^r | r = 1, 2, ..., d\}$ is the collection of unique value sets of each attribute, where $O^r = \{o_1^r, o_2^r, ..., o_{v^r}^v\}$ is the unique value set of attribute A^r, where v^r is the number of unique values. Note that if an attribute value in the dataset is missing value, it is represented as "?". For the sake of simplicity, objects with and without missing values are referred to as incomplete and complete objects, respectively, while datasets with and without missing values are referred to as incomplete and complete datasets, respectively. For example, Table.3 shows an incomplete mixed-attribute dataset with three attributes, and we can easily discover the characteristics of the three different attributes. For nominal and ordinal attributes, the possible values $o_{v^r}^r$ fall in a finite discrete space. In particular, there is an additional ordering relationship between ordinal attribute values (e.g., in the example table $o_1^2 \succ o_2^2 \succ o_3^2$, $\succ$ means that the left value is higher in rank than the right value). For numerical attributes, v^r values are relatively large, falling in an infinite $1 - D$ real space, and there is also an ordering relationship (e.g., in the example table $o_1^3 > o_2^3 > o_3^3$).

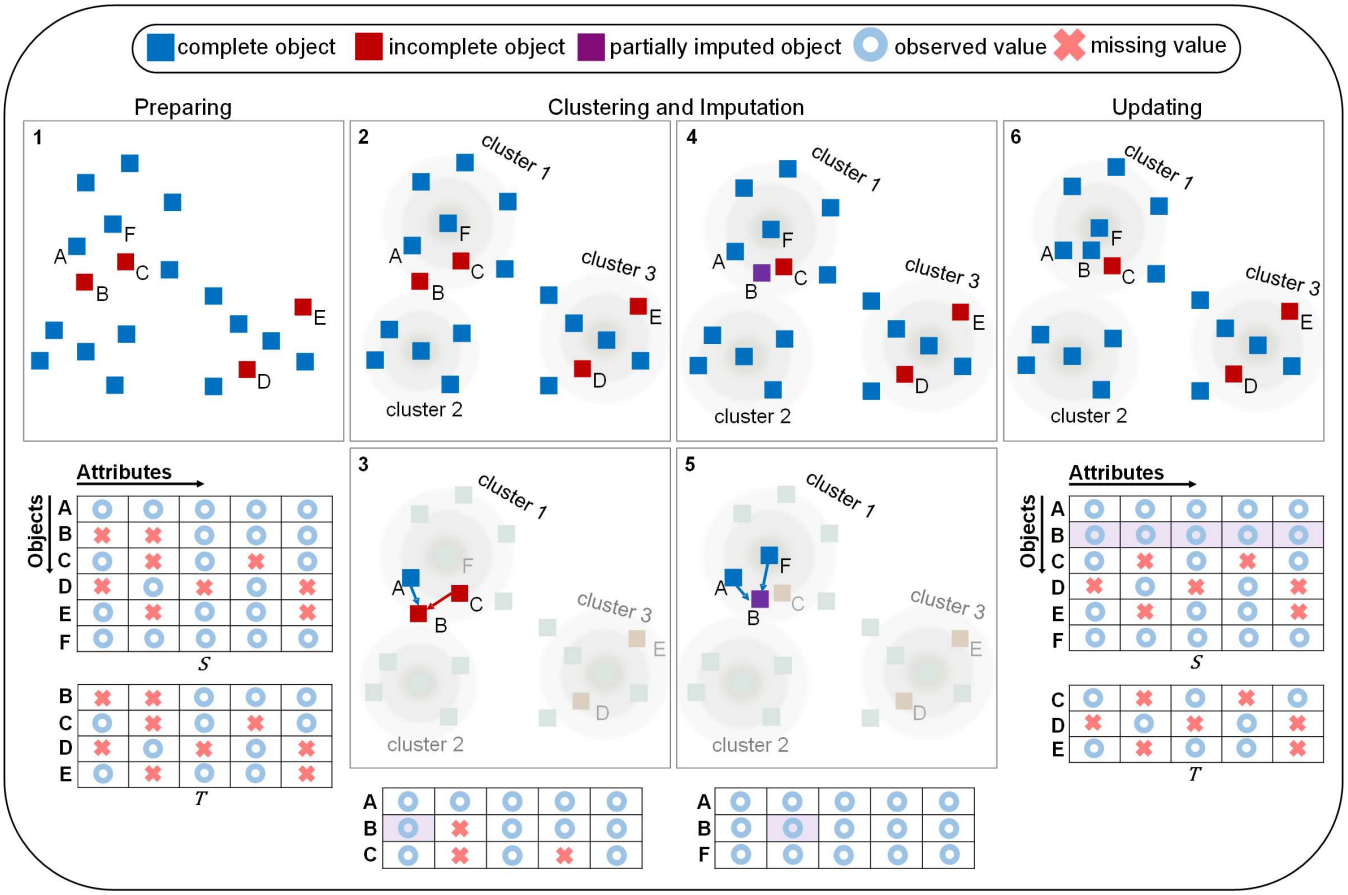


FIGURE 1 Overview of the proposed clustering-imputation co-learning paradigm. In this paradigm, we complete the imputation process for the incomplete object B in the order of blocks numbered 1−6. Blocks 2−3−4−5 represent the changes in objects during the clustering and imputation stages. Dataset S represents the incomplete dataset, while dataset T is a collection of incomplete objects extracted from S. Only a portion of the object from S and T are shown in the figure for reference. The changes in attribute values can be observed from the datasets S and T below the blocks. Blocks 3 and 5 respectively indicate the completion of the first and second missing value imputations for object B. After updating datasets S and T in the updating stage, the process iteratively returns to the preparing stage to select the next incomplete object for clustering and imputation until T is empty.

In the following, we provide a comprehensive overview of the proposed CIMI. Figure.1 shows the overview of the CIMI method. In this flowchart, this method is divided into three stages: the preparing stage, the clustering and imputation stage, and the updating stage.

In the preparing stage, incomplete objects with missing values are separated from the incomplete dataset S and noted as T, the first object in T is designated as the target object. The clustering and imputation process is divided into two steps: the clustering step and the imputation step. In the clustering step, we define a clustering algorithm named Iterative Centroid Replacement Clustering (ICRC). The algorithm randomly selects k objects from S as initial centroids to generate k clusters, subsequently updating the centroids until they no longer change. In each iteration, all non-centroid points are attempted as replacements for the current centroids, and the replacement loss is calculated. The non-centroid point with the smallest loss is chosen as the new centroid. The replacement loss is calculated as follows:

$$\text{Cost} = \sum_{i=1}^{n} (\Psi(x_i, c_j^*) - \Psi(x_i, c_j)), \tag{1}$$

where x_i is the remaining non-centroid objects, c_j^* is the new centroid, c_j is the old centroid, and the function $\Psi(\cdot, \cdot)$ denotes the dissimilarity between two objects.

Next, the CIMI performs the imputation step by identifying the cluster to which the target object belongs (denoted as the target cluster). In the target cluster, the natural neighbors[44] of the target object are calculated, and their values are used as reference values for imputation. According to attribute type, the mean/mode of the reference values for numerical/nominal

TABLE 4 An example of how to calculate the distance between incomplete objects.

	Attributes			
Objects	A^1	A^2	A^3	A^4
x_a	1	2	?	7
x_b	1	2	5	8

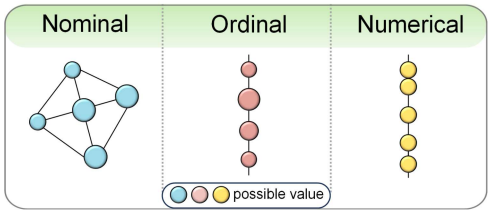


FIGURE 2 Attribute graph space structure.

(and ordinal) attributes is selected. For objects with multiple missing values, attributes with the highest inter-dependence on complete attributes are prioritized for imputation. After filling one missing attribute, the target object's cluster is recalculated, and the next missing attribute is imputed based on the new target cluster's reference objects. This process continues until all missing values in the target object are imputed. Next, in the updating stage, the target object is removed from T, the imputed target object is replaced in S, and the clustering and imputation stages are repeated until T is empty. Finally, the proposed CIMI returns a complete dataset and k clusters. In the following, we describe the three elements that affect the imputation results: the distance between incomplete objects, the inter-dependence between attributes, and the neighbors of the target object.

3.2 | Distance between incomplete objects

For the distance matrix calculation of incomplete objects, we first give the definition of the distance $\Psi(x_i, x_j)$ between two data objects x_i and x_j in the complete cases:

$$\Psi(x_i, x_j) = \sqrt{\sum_{r=1}^{d} \Psi^r(x_i^r, x_j^r)^2}, \tag{2}$$

where the $\Psi^r(x_i^r, x_j^r)$ is the dissimilarity of two possible values x_i^r and x_j^r of objects x_i and x_j in attribute A^r. When missing values occur in x_i, x_j, or both, we modify the distance definition. Specifically, we calculate the distance between the complete attribute values and then normalize it to account for the missing values[45]. Suppose md is the number of attributes that are missing, the distance $M\Psi(x_i, x_j)$ is computed as follows:

$$M\Psi(x_i, x_j) = \sqrt{\frac{d}{d - md} \sum_{r=1}^{d} M\Psi^r(x_i^r, x_j^r)^2}, \tag{3}$$

where $M\Psi^r(x_i^r, x_j^r)$ is defined as

$$M\Psi^r(x_i^r, x_j^r) = \begin{cases} 0, & \text{if } x_i^r \text{ or } x_i^r \text{ is missing} \\ \Psi^r(x_i^r, x_j^r), & \text{otherwise.} \end{cases}$$

In other words, we assume that the distances between the missing attributes are equal to the average of the distances between the complete attributes. This approach helps reduce the bias introduced by traditional methods, which only use the distance between complete attributes as the measure of distance.

Remark 1. As shown in Table 4, in this example, we use a simple Euclidean distance for illustration. The distance between x_a and x_b is calculated as: $4/(4-1)\sqrt{[(1-1)^2+(2-2)^2+(7-8)^2]} = \sqrt{1.33}$. The traditional method ignores the attributes with missing values and directly calculates the distance between the remaining complete attributes: $\sqrt{[(1-1)^2+(2-2)^2+(7-8)^2]} = 1$. Essentially, this approach assumes that the missing attribute value of x_a is equal to that of x_b, which means the missing value is assumed to be 5. If the true value is 4 or 6, then the distance between x_a and x_b is 1.41. Clearly, the distances calculated by our strategy are closer to the true distances. The traditional assumption introduces bias and negates the uncertainty of the missing values. In contrast, our strategy leverages the uncertainty of the missing values, which can significantly improve the stability of calculating the distance between incomplete objects.

3.3 | Inter-dependence between attributes

Through the innovative design of graph-based unified dissimilarity (GUD)[24], we recognize that there are connections between different attributes. Based on these connections, the generated unified measure enables a consistent representation of various types of attributes. We apply this unified measure to cluster incomplete mixed-type data, which avoids biases that arise from converting between different attribute types when calculating the distance between samples. Additionally, this unified measure allows us to calculate the inter-dependence between different attributes, thereby guiding the imputation of missing values.

Connections between different attributes:

- Ordinal and numerical: if we increase the value of the ordinal attribute infinitely, then the ordinal values can have characteristics of the numerical attribute values, infinity and ordering. The distance between ordinal values can be reflected by the numerical attribute.
- Ordinal and nominal: there is a rank relationship between ordinal attribute values (e.g., in the example Table4, the O^2 has three levels: $o_1^2 \succ o_2^2 \succ o_3^2$). All nominal attribute values can be regarded as ordinal attribute values with two-level relations. Each pair of nominal values can be treated as a pair of Concept-Contradiction Values (CCVs). In the nominal attribute space, there are $(v^r(v^r-1)/2)$ pairs of CCVs, while in the ordinal attribute space, there is only one pair of CCVs (e.g., o_1^2 and o_3^2).
- Nominal and numerical: now the nominal and numerical attributes can be connected based on the ordinal attribute. The distance between any two nominal values can be viewed as an ordinal attribute with only two possible values($v^r = 2$), then the distance can be reflected by the numerical attribute. Therefore, the three attributes are connected.

Then a graph space is constructed according to the intrinsic characteristics of the attributes and the connection relationship between them. This structure maps the heterogeneous attributes into homogeneous space, as shown in Fig. 2. In this graph-based structure, the possible values of each type of attribute are connected in different ways. Nodes represent possible values, and the connections between nodes represent the distances between these possible values. The size of the node represents the frequency of occurrence of the possible values. Based on this structure, distances between different types of attribute values can be uniformly measured. More details can be found in the previous study[24]. The $\psi^{rs}(o_m^r, o_h^r)$ represents the dissimilarity between values o_m^r and o_h^r as reflected by A^s, is defined as

$$\psi^{rs}(o_m^r, o_h^r) = \sum_{g=1}^{v_{mh}^r-1} |u_g^{rs} - u_t^{rs}| \cdot t_{gt}^{rs}, \tag{4}$$

where $t = g + 1$. u_g^{rs} and u_t^{rs} are the conditional probability distributions (CPD) of the values in O^s as given the g-th and t-th values in O_{mh}^r. More specifically, if A^r is a nominal attribute, every two nodes in the corresponding graph are directly linked, and thus,we have $O_{mh}^r \equiv \{o_m^r, o_h^r\}$ and $v_{mh}^r \equiv 2$. If A^r is an ordinal or numerical attribute, all the values in O^r ordered between o_m^r to o_h^r (including themselves) are the nodes on the shortest path from o_m^r to o_h^r in the corresponding graph, and thus, we have $v_{mh}^r = |m-h| + 1$, and $O_{mh}^r = \{o_m^r, o_{m+1}^r, ..., o_h^r\}$ if $m < h$, or $O_{mh}^r = \{o_h^r, o_{h+1}^r, ..., o_m^r\}$ if $m > h$. In Eq. (4), the $v_{mh}^r - 1$ subtransformation costs on the path through the v_{mh}^r nodes are successively accumulated. The CPD difference $|u_g^{rs} - u_t^{rs}|$ describes the differences between the corresponding values of A_g^{rs} and A_t^{rs} that should be transported for offsetting in the graph transformation. Vector $t_{gt}^{rs} = [t_{gt1}^{rs}, t_{gt2}^{rs}, ..., t_{gtv^s}^{rs}]^T$ stores the minimum total edge lengths that should be taken to transport each of the differences. The CPD of the values in O^s as given o_m^r is defined as

$$u_m^{rs} = [p(o_1^s|o_m^r), p(o_2^s|o_m^r), ..., p(o_{v^s}^s|o_m^r)]^T, \tag{5}$$

$$p(o_g^s|o^r m) = \frac{\sigma(X_g^s \cap X_m^r)}{\sigma(X_m^r)}, \tag{6}$$

where $X_m^r = \{x_i | x_i^r = o_m^r, i = 1, 2, ..., n\}$ is a subset of X with the r-th values of all its objects equal to o_m^r, and the function $\sigma(\cdot)$ counts the cardinality of a set.

Then we discuss the inter-dependence degree between attributes, which is used to guide the preferential selection of attributes for imputation. The distance $\Psi^r(o_m^r, o_h^r)$ is defined as

$$\Psi^r(o_m^r, o_h^r) = \sum_{s=1}^{d} \psi^{rs}(o_m^r, o_h^r) \cdot w^{rs}, \tag{7}$$

where the $\psi^{rs}(o_m^r, o_h^r)$ partially reflects the dependence of A^s on A^r and w^{rs} controls the contribution of A^s to $\Psi^r(o_m^r, o_h^r)$. If the dissimilarities between different values of A^r reflected by A^s are consistently higher than those reflected by other attributes, A^r and A^s are considered to have stronger inter-dependence[24]. The weight w^{rs} is defined as

$$w^{rs} = \frac{\sum_{q=1}^{v^{r<*>}-1} \sum_{c=q+1}^{v^{r<*>}} \psi^{rs}(o_q^r, o_c^r)}{\frac{v^{r<*>}(v^{r<*>}-1)}{2} \cdot (v_{qc}^r - 1)}, \tag{8}$$

where o_q^r and o_c^r are the q-th and c-th unique values in the set $O^{r<*>}$, and $O^{r<*>}$ is the set contains $v^{r<*>}$ concept-contradicted nodes of the corresponding graph. v_{qc}^r is the number of intermediate values on the shortest path between o_q^r and o_c^r. When multiple attributes in the target object have missing values, we designate the missing attribute as A^r and the complete attribute as A^s. We then impute the missing attribute with the highest inter-dependence with the complete attribute first, maximizing the likelihood of deriving the correct value based on the available information.

Remark 2. Most existing methods, when faced with multiple missing values, either impute them simultaneously, sequentially according to their order in the dataset, or based on the number of missing values, from fewest to most. These strategies ignore the inter-dependence between different attributes, potentially leading to a chain reaction. In other words, errors in earlier imputations can cause subsequent imputations to be incorrect, resulting in a significant loss of accuracy. Our method, however, prioritizes imputing the missing attribute that has the highest inter-dependence with the observed complete attributes. This way, each imputed value provides more valuable information for subsequent imputations. For example, in a patient dataset, if both "Blood Type" and "Weight" attributes are missing, while "Height" and "Gender" attributes are complete, we would prioritize imputing the weight attribute.

3.4 | Neighbors of target object

In the clustering and imputation stage of CIMI, we find neighbors in the target cluster that are similar to the target object to use as references for imputation. However, determining how many similar neighbors to use as references is a question worth exploring. Natural neighbor (NaN)[44] provides a solution to this problem. Inspired by human social friendships, the NaN is developed to find multiple neighbors for each data point by considering the characteristics of the dataset. Compared to traditional neighbor methods, NaN is parameter-free. Based on NaN searching, if x_i belongs to the neighbors of x_j and x_j belongs to the neighbors of x_i, then x_i and x_j are natural neighbors of each other. The natural neighbors of the object X_i is defined as follows:

$$x_j \in NaN(x_i) \Leftrightarrow (x_i \in NN_r(x_j)) \wedge (x_j \in NN_r(x_i)), \tag{9}$$

where r is the number of search cycles when dataset S reaches a natural steady state. $NaN(x_i)$ is the set that contains the natural neighbors of object x_i, whereas $NN_r(x_i)$ is the set that contains r nearest neighbors of x_i.

Remark 3. After clustering all the objects, within the cluster where the incomplete object is located, we first filter out the objects with complete values for the currently missing attribute. Then, we calculate the natural neighbors of the current incomplete object among these filtered objects. Finally, the corresponding attribute of these natural neighbors becomes the imputed value for the missing attribute (mean for numerical attributes, mode for categorical attributes). Since we have already obtained the distances between objects during the previous clustering process, finding the natural neighbor is straightforward and efficient. Accordingly, by natural neighbor, we can identify the most qualified objects to provide information for the target object.

3.5 | Overall algorithm and complexity analysis

The clustering algorithm ICRC and the entire imputation method CIMI are summarized in Algorithm 1 and Algorithm 2, respectively. The time complexity of the CIMI method can be divided into two parts: First, clustering all objects, which includes calculating the distance matrix $O(n^2)$ and initial clustering $O(Ink)$. Second, the imputation process, which includes sorting incomplete objects $O(m \log m)$, and iterative imputation $O(m(Ink + an_c \log n_c))$, where n_c is the number of objects in the target cluster, n is the number of objects, m is the number of incomplete objects, k is the number of clusters, I is the number of iterations, and a is the number of missing attributes. Combining these, the total time complexity is $O(n^2)+O(mInk)+O(man_c \log n_c)$. It is worth noting that CIMI is designed for heterogeneous datasets containing categorical attributes, and most such datasets are

Algorithm 1 ICRC: Iterative Centroid Replacement Clustering

Input: Dataset S, number of clusters k, objects distance matrix DT.
Output: k clusters of S, k cluster centroids.

$c_j \leftarrow k$ objects as initial centroids;
for object x_i in S **do**
  Assign x_i to the cluster represented by the nearest centroid based on DT;
end for
for each centroid c_j **do**
  for each unselected non-centroid object x_i **do**
    Calculate the $Cost$ of replacing x_i with c_j;
  end for
end for
if any $Cost < 0$ **then**
  $c_j^* \leftarrow$ non-centroid object in H with the minimum $Cost$;
  Swap the corresponding centroid c_j with c_j^*;
end if

Algorithm 2 CIMI: Clustering Incomplete data with Missing values Imputation

Input: Dataset S, number of clusters k, set T of incomplete objects divided from S.
Output: Complete dataset S, k clusters of S.

$x_i \leftarrow$ objects of sorted indices in T, w.r.t. increasing amount of missing values;
while T is not empty **do**
  $DT \leftarrow$ objects distance matrix of S using Eq.(7);
  k clusters, centroids = $ICRC(S, k, DT)$;
  $c_j \leftarrow$ cluster containing x_i;
  $NaN_{x_i} \leftarrow$ natural neighbor of x_i in c_j;
  for each missing attribute in x_i **do**
    $A^r \leftarrow$ the selected attribute using Eq.(8);
    $x_i \leftarrow$ impute x_i^r using A^r of NaN_{x_i};
  end for
  Remove x_i from T, update x_i in S;
end while

naturally small in scale, often consisting of manually recorded small-domain data, such as survey questionnaires. Therefore, the number of n will not be too large, avoiding excessive time complexity.

4 | EXPERIMENTAL SETTINGS

To evaluate the performance of CIMI, we conduct two types of experiments on three different types of datasets. 1) Imputation efficiency evaluation: we evaluate the imputation efficiency of CIMI with the existing methods on 9 real-world datasets using three indicators for different data types. 2) Clustering performance evaluation: we performed clustering on the datasets after imputation and compared the clustering performance of CIMI with existing methods using two clustering metrics. In the following, we will describe the experimental setting from three aspects: comparison methods, datasets and evaluation metrics.

4.1 | Comparison methods

The experiment compared the performance of the CIMI method with four imputation methods: Mean/Mode Substitution (MS, denoted as MMS for heterogeneous datasets)[13], k-Means Clustering-based Missing values Imputation (KMCMI)[46], k-Nearest

TABLE 5 Characteristics of datasets. $d^{<n>}$, $d^{<o>}$, $d^{<u>}$ indicate the numbers of nominal, ordinal and numerical attributes, respectively. n and k indicate the numbers of objects and clusters, respectively.

No.	Dataset	Abbrev.	$d^{<n>}$	$d^{<o>}$	$d^{<u>}$	n	k
1	Iris	IR	0	0	4	150	3
2	Wine	WI	0	0	13	178	3
3	Seed	SE	0	0	7	210	3
4	Zoo	ZO	15	1	0	101	7
5	Soybean	SB	35	0	0	47	4
6	Hayes Roth	HR	2	2	0	160	3
7	Diagnosis	DS	5	0	1	120	2
8	Teacher Assistant	TA	4	0	1	151	3
9	Hepatitis	HE	13	0	6	155	2

Neighbors-based Missing values Imputation (KNNMI)[17], and MissForest (MF)[23] across different types of datasets. For numerical datasets, Mean Substitution, KMCMI, and MF are used; for categorical datasets, Mode Substitution, KNNMI, and MF are used; and for mixed-attribute datasets, Mean/Mode Substitution, KNNMI, and MF are used. Additionally, in the KNNMI method, the k neighbors is set to the square root of the number of observed complete objects suggested by Lall and Sharma[47].

Many studies have compared different imputation methods, but have not evaluated them through clustering performance. Additionally, studies that have used clustering for comparison have only employed a single type of imputation method.[4]. In our experiments, after completing the imputation with different methods, we calculated the imputation accuracy and also compared the clustering metrics post-clustering. We aim to explore the deeper relationship between imputation and clustering through this approach. Therefore, we performed clustering on the imputed datasets using classical clustering methods based on the data type after imputation: k-Means clustering[48] for numerical datasets, k-Modes clustering[49] for categorical datasets, and k-prototypes clustering[50] for mixed-attribute datasets. Generally, two factors affect the performance of clustering algorithms and the CIMI method proposed in this paper: the initialization method used by the algorithm and the choice of the number of clusters k. For the first factor, the aforementioned algorithms use a random initialization method to generate initial clusters. This may result in low clustering quality in some instances. For this reason, each algorithm is run 10 times on each dataset. The overall performance is then calculated by averaging the results of all trials. For the second factor, an inaccurate estimate of k can affect the quality of clustering results. Therefore, in our experiments, k is set to the number of classes in the ground truth labels.

4.2 | Datasets

The performance of the CIMI method is evaluated using 9 datasets from the UCI Machine Learning Repository[‡]: 3 numerical datasets, 3 categorical datasets, and 3 mixed-attribute datasets. Table 5 lists the characteristics of these datasets. These datasets are used to assess the CIMI methods performance with various types of data encountered in real-world applications. The datasets in their entirety, before the introduction of any missing values, are referred to as the original data. Then we introduce missing values at rates of 10%, 20%, 30%, 40%, and 50%, removing data completely at random in each experiment.

4.3 | Evaluation metrics

To evaluate the efficiency of different imputation methods for various types of missing value data, and the effect of clustering post-imputation, we considered two different criteria. First, we assess the distance between the original and imputed data. For numerical data, we use root mean squared error (RMSE). For categorical data, we use the simple matching method for Accuracy. For heterogeneous data, we use modified RMSE (mRMSE)[51]. The specific formulas outlined below:

$$RMSE = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(\hat{x}_i^r - x_i^r)^2}, \quad Accuracy = \frac{\sum_{i=1}^{m} I(\hat{x}_i^r, x_i^r)}{m}, \tag{10}$$

[‡] http://archive.ics.uci.edu/

where m is the number of missing values, $\hat{x}_i^r$ and and x_i^r are the value of the original data and imputed data at the i-th missing value position. $I(\cdot,\cdot)$ is an indicator function that takes the value 1 when $\hat{x}_i^r = x_i^r$, and 0 otherwise.

$$mRMSE = \sqrt{RMSE_0^2 + RMSE_1^2}, \tag{11}$$

where $RMSE_0$ and $RMSE_1$ are $RMSE$s of categorical attributes M_0 and numerical attributes M_1, respectively and are defined in

$$RMSE_0 = \sqrt{\frac{1}{|M_0|}\sum_{A^r \in M_0} 1_{\hat{x}_i^r \neq x_i^r}}, \quad RMSE_1 = \sqrt{\frac{1}{|M_1|}\sum_{A^r \in M_1} (\hat{x}_i^r - x_i^r)^2}. \tag{12}$$

We used two metrics to evaluate the quality of the clustering results: an external metric, the Adjusted Rand Index (ARI)[52], and an internal metric, the Silhouette Index (CVI)[53]. These metrics use the information in the original dataset as the ground truth and compare it with the clustering results generated by the imputation methods to measure the match between the output clusters and the ground truth. Specifically, we omitted the class attributes in the dataset during the clustering process and used them only for evaluating the clustering results. The ARI is determined by comparing the true class labels of the original data with the cluster partitions generated by the clustering process. A higher ARI value indicates that the clustering results are more similar to the true class labels, signifying better clustering performance. The CVI considers both the compactness within clusters and the separation between clusters. A higher CVI value indicates more compact clusters and better separation between clusters. To compare with the clustering effect on the complete dataset, we also clustered the original complete dataset using the same clustering method, measured ARI and CVI, and recorded them as ORI in the clustering table.

5 | EXPERIMENTAL RESULTS

In this section, we will present the experimental results for three types of datasets: numerical, categorical, and mixed-attribute. The results for each dataset type are divided into two parts: imputation efficiency and clustering performance. Based on these results, we will analyze how missing values imputation impacts clustering performance and discuss the findings in detail.

5.1 | Results on numerical datasets

We first focus on numerical datasets, with results presented in Figure 3 and Table 6. As expected, the simplest MS method consistently performed poorly across different missing rates in the three datasets. We observed that the CIMI method outperforms the other methods, reducing imputation error by more than 10% in many cases, while the second-best method is MF. Moreover, by examining the ARI and CVI values for clustering under different missing rates, combined with the imputation results, we can first draw two conclusions: 1) Effective imputation can improve clustering, and 2) As the missing rate increases, imputation becomes more challenging due to the reduction of available information. Interestingly, while MS performed poorly overall, it sometimes resulted in higher CVI values compared to the KMCMI method. This anomaly is likely because erroneous imputations from MS create a large amount of identical erroneous data, which can be more easily clustered into the same group, inflating the CVI value. This hypothesis will be further tested on other types of datasets.

5.2 | Results on categorical datasets

Similarly, we conducted experiments on categorical datasets. As shown in Figure 4, CIMI consistently imputes missing values better than the compared methods, with accuracy improvements ranging from 10% to 30%. Surprisingly, the simple mode-substitution method did not perform poorly, especially when the missing rate is above 30%. In these categorical datasets, each attribute has a limited and small number of possible values. When the missing rate is high, the relationships between data attributes are greatly disrupted, making effective data values extremely sparse. In such cases, using more sophisticated methods becomes challenging, and simpler methods can be more effective. Furthermore, as shown in Table 7, CIMI integrates imputation into the clustering process, resulting in higher clustering efficiency in the final inference. Whether considering ARI or CVI, CIMI performs well. It is important to note that only when the imputation accuracy is high, meaning the original dataset is as accurately restored as possible, are the clustering results truly valuable. For cases with low imputation accuracy and high

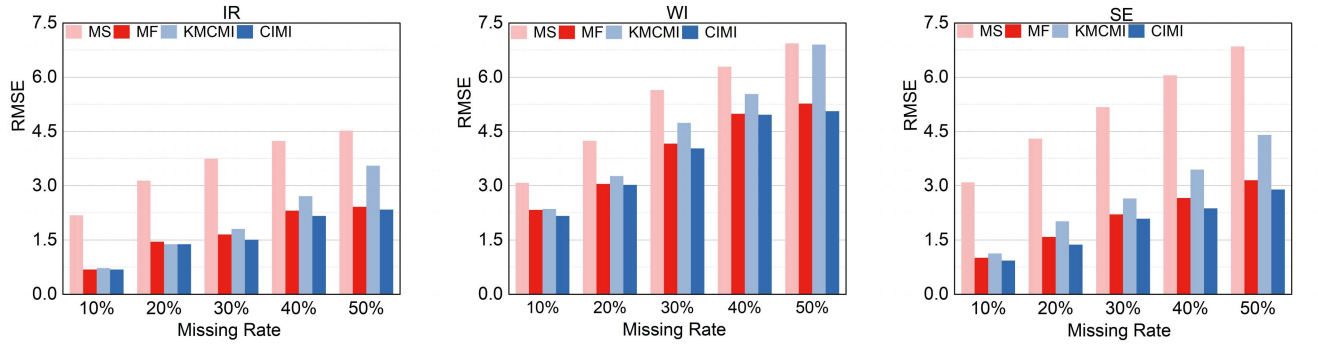


FIGURE 3 RMSE of the imputation methods on pure numerical datasets.

TABLE 6 Cluster performance ARI and CVI on numerical datasets.

Missing Rate	Methods	IR		WI		SE	
		ARI	CVI	ARI	CVI	ARI	CVI
0 %	ORI	0.7010	0.4820	0.8470	0.3010	0.6930	0.4220
10%	MS	0.6250	0.4270	0.8134	0.2770	0.7242	0.3454
	MF	0.6352	0.4974	0.8382	0.3204	0.6982	0.4298
	KMCMI	0.6840	0.0560	0.8182	0.3164	0.7024	0.0010
	CIMI	**0.7010**	**0.5060**	**0.8508**	**0.3214**	**0.7366**	**0.4374**
20 %	MS	0.5662	0.3492	0.8514	0.2312	0.6146	0.2994
	MF	0.7180	0.5016	0.8628	0.3190	0.6832	0.4390
	KMCMI	0.5958	0.0394	0.8316	0.0028	0.7260	0.0010
	CIMI	**0.7220**	**0.5032**	**0.8630**	**0.3196**	**0.6882**	**0.4416**
30 %	MS	0.4670	0.3172	0.6920	0.1910	0.5942	0.2672
	MF	0.7416	0.5270	0.7550	0.3132	0.6704	0.4608
	KMCMI	0.6796	0.0364	0.6560	0.0020	0.6168	0.0010
	CIMI	**0.7617**	**0.5416**	**0.7620**	**0.3402**	**0.6972**	**0.4770**
40 %	MS	0.3898	0.3178	0.7180	0.1656	0.3906	0.2588
	MF	0.5866	0.5438	0.7640	0.3506	0.6346	0.4586
	KMCMI	0.6094	0.0314	0.7112	0.0010	0.5916	0.0002
	CIMI	**0.6384**	**0.5810**	**0.7774**	**0.3835**	**0.6536**	**0.4594**
50 %	MS	0.4110	0.3396	0.4908	0.1240	0.2434	0.2470
	MF	0.7258	0.5482	0.6920	0.3332	0.6586	0.4804
	KMCMI	0.4604	0.0316	0.3740	0.0016	0.5668	0.0010
	CIMI	**0.7319**	**0.5572**	**0.7040**	**0.3474**	**0.6702**	**0.4912**

clustering efficiency, it indicates that when a large amount of data is wrongly imputed, erroneous data are more likely to be aggregated, thus demonstrating the aforementioned contradictory situation.

5.3 | Results on mixed-attributes datasets

In the following, we investigate the heterogeneous datasets, with the results shown in Figure 5 and Table 8. We can see that CIMI performs well, sometimes reducing the average mRMSE by up to 30% compared to KNNMI and slightly better than MF by an average of 10%. The three datasets each have unique characteristics and contain multiple possible values for each attribute, which encompass rich relationships between attribute values and are valuable for CIMI to utilize. These relationships provide better support for CIMI, resulting in strong performance on these datasets. Additionally, it can be observed that in scenarios with high missing rates in mixed datasets, the popular KNNMI method performs very poorly. This indicates that when calculating the distance between mixed-type data under high missing rates, the presence of missing values can easily cause significant bias, leading to imputation errors. Therefore, it is important to choose a more reasonable distance calculation

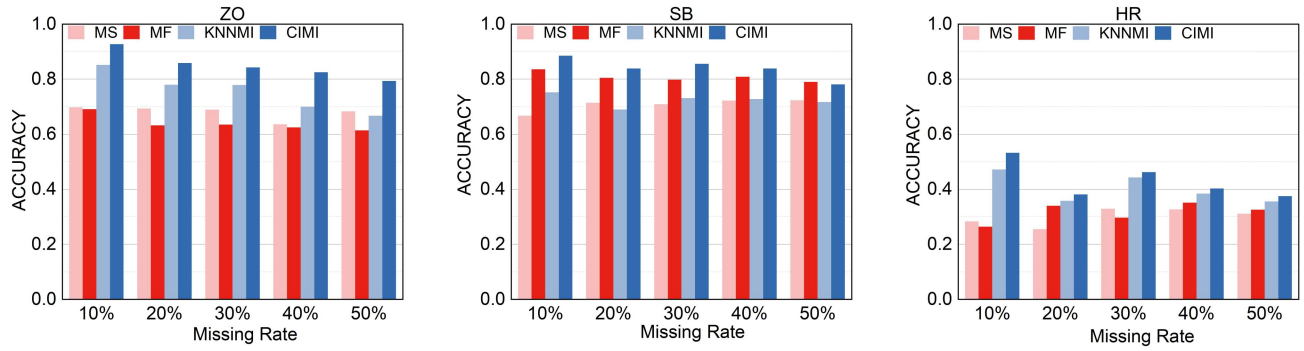


FIGURE 4 Accuracy of the imputation methods on categorical datasets.

TABLE 7 Cluster performance ARI and CVI on categorical datasets.

Missing Rate	Methods	ZO		SB		HR	
		ARI	CVI	ARI	CVI	ARI	CVI
0 %	ORI	0.8100	0.4940	1	0.4230	0.0024	0.2450
10 %	MS	0.6352	0.3464	0.4824	0.2406	-0.0034	0.1884
	MF	0.6774	0.3278	0.6094	0.2950	0.0004	0.1576
	KNNMI	0.7318	0.4442	0.7224	0.3324	-0.0016	0.1830
	CIMI	**0.8274**	**0.4662**	**1**	**0.4920**	**0.0708**	**0.2226**
20 %	MS	0.5924	0.2826	0.5670	0.2794	0.0024	0.1996
	MF	0.6410	0.2756	0.7524	0.3948	0.0004	0.1672
	KNNMI	0.5298	0.3386	0.5582	0.2550	0.0042	0.2780
	CIMI	**0.6722**	**0.4472**	**1**	**0.5044**	**0.0840**	**0.2862**
30 %	MS	0.3670	0.2070	0.3744	0.1812	0.0088	0.2202
	MF	0.4360	0.2178	0.7318	0.3420	-0.001	0.1584
	KNNMI	0.5604	0.3096	0.4300	0.2082	0.0038	0.2232
	CIMI	**0.6990**	**0.5350**	**1**	**0.5160**	**0.0140**	**0.2990**
40 %	MS	0.1814	0.1710	0.2574	0.1152	0.0210	0.2234
	MF	0.2110	0.1624	0.7848	0.4082	0.0114	0.1730
	KNNMI	0.2390	0.2248	0.6376	0.2984	-0.0098	0.2538
	CIMI	**0.4930**	**0.5580**	**0.8800**	**0.5630**	**0.0800**	**0.3310**
50 %	MS	0.1208	0.1978	0.1062	0.2278	0.0010	0.2874
	MF	0.2032	0.1682	0.6420	0.3322	0.0022	0.1844
	KNNMI	0.3008	0.2546	0.3004	0.2590	-0.0058	0.4422
	CIMI	**0.43240**	**0.6534**	**0.7390**	**0.5520**	**0.1150**	**0.4912**

strategy based on different missing scenarios before imputation. Moreover, the same anomaly mentioned earlier is observed here. On the DS and HE datasets with a 50% missing rate, KNNMI has the lowest imputation accuracy but the highest CVI value. This situation has also occurred in the previous two types of datasets. Therefore, we believe that when a large amount of erroneously imputed data is present, it tends to cluster together. As a result, this imputation behavior may leads to a higher CVI value.

5.4 | Discussions

After comparing the imputation efficiency and clustering performance of different methods across multiple datasets, we further explored the interesting phenomenon between imputation and clustering:

- When the imputed data is not completely accurate, the clustering performance (ARI and CVI) are sometimes higher than that of the original data. This can be seen from the provided imputation accuracy charts and clustering performance tables.

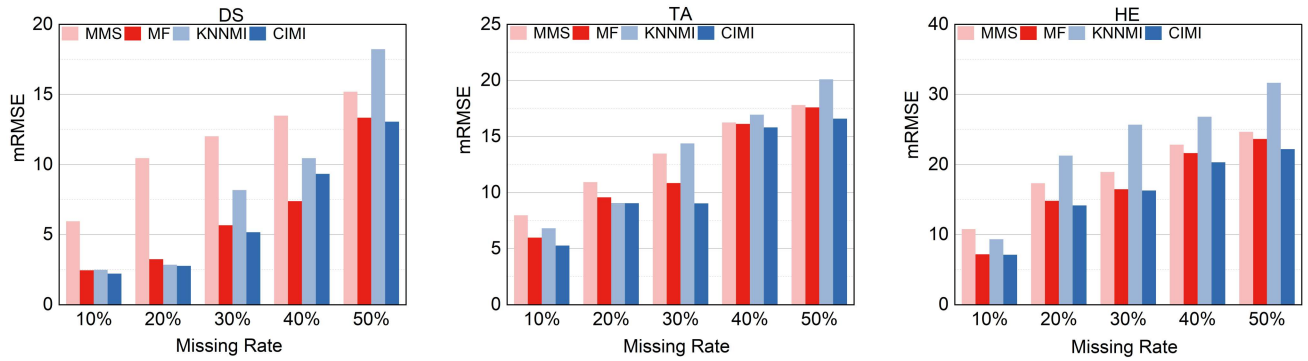


FIGURE 5 mRMSE of the imputation methods on mixed-attributes datasets.

TABLE 8 Cluster performance ARI and CVI on mixed-attributes datasets.

Missing Rate	Methods	DS		TA		HE	
		ARI	CVI	ARI	CVI	ARI	CVI
0 %	ORI	0.5590	0.3920	0.0190	0.2660	0.1750	0.2500
10 %	MMS	0.0680	0.3030	0.0200	0.0995	0.0826	0.2060
	MF	0.1425	0.3890	0.0122	0.0888	0.0894	0.2156
	KNNMI	0.2690	0.3390	0.0116	0.1567	0.0870	0.2276
	CIMI	**0.3150**	**0.3897**	**0.0233**	**0.1865**	**0.1028**	**0.2316**
20 %	MMS	0.1296	0.2944	0.0125	0.1555	0.1402	0.2234
	MF	0.2278	0.4206	0.0222	0.1447	0.0776	0.2024
	KNNMI	0.1094	0.1072	0.0213	0.1933	0.0626	0.1848
	CIMI	**0.3270**	**0.4558**	**0.0235**	**0.1963**	**0.1592**	**0.2366**
30 %	MMS	0.0952	0.3507	0.0090	0.1518	0.1190	0.2390
	MF	0.1847	0.3853	0.0111	0.1389	0.0722	0.2634
	KNNMI	0.3315	0.3981	0.0124	0.1723	0.0184	0.1648
	CIMI	**0.3476**	**0.4075**	**0.0151**	**0.1984**	**0.1040**	**0.2882**
40 %	MMS	0.0423	0.2103	0.0012	0.1626	0.0094	0.2008
	MF	0.2461	0.2242	0.0140	0.2138	0.0876	0.3022
	KNNMI	0.2511	0.3311	0.0054	0.1808	-0.0110	0.3126
	CIMI	**0.2558**	**0.4244**	**0.0180**	**0.2500**	**0.0916**	**0.3518**
50 %	MMS	0.0415	0.2548	0.0046	0.2145	0.0404	0.2400
	MF	0.0974	0.3896	0.0163	0.2113	0.0484	0.3024
	KNNMI	0.0297	**0.4382**	0.0035	0.2115	-0.0012	**0.3516**
	CIMI	**0.1905**	0.4059	**0.0186**	**0.2307**	**0.1396**	0.3288

The reason for this is that the imputed data is inferred from the existing data, making their relationships more explicitly correlated and easier to cluster.

- Correctly imputed datasets sometimes have worse clustering performance (in terms of CVI) than incorrectly imputed datasets. As the missing rate increases, the imputation accuracy may decrease due to significant information loss caused by missing values, making it difficult to infer the correct imputed values. At the same time, incorrect imputation can lead to large blocks of entirely erroneous data, which are more likely to produce high clustering performance.

In conclusion, the performance of different methods on various types of datasets indicates that the presence of missing values can significantly impact clustering results. Effective imputation can improve clustering accuracy. Moreover, there is a clear correlation between the accuracy of imputation and the quality of clustering; as imputation accuracy increases, clustering results also improve. Therefore, it is crucial to handle missing values appropriately before conducting clustering tasks. When evaluating methods capable of clustering incomplete objects, the choice of clustering metrics is essential. We recommend using external clustering metrics for evaluation. High imputation error rates can still produce favorable internal clustering metrics, which can easily mislead users and affect subsequent decisions.

The proposed CIMI method ensures imputation accuracy while achieving clustering results that are closer to those obtained using complete data, compared to other imputation methods. In the CIMI mechanism, clustering and missing values imputation form a mutually beneficial relationship. Specifically, clustering provides references for missing values, while missing values imputation supplies additional information for clustering. This synergy allows CIMI to achieve effective imputation and clustering performance within a single task, potentially eliminating the need for separate missing values handling as a preprocessing stage. Furthermore, CIMI consistently demonstrates stable performance across different types of datasets (any combination of nominal, ordinal, and numerical attributes), highlighting its robustness.

6 | CONCLUSION

To address the issue of clustering incomplete mixed-attribute data, this paper proposes a method called Clustering Incomplete data with Missing values Imputation (CIMI). CIMI integrates imputation and clustering tasks into a complementary module: clustering provides reliable reference information for imputing missing values, and imputation enriches object information for clustering, achieving a twofold benefit. CIMI emphasizes the intrinsic characteristics of different attribute types and utilizes inter-attribute dependencies to guide missing values imputation. In summary, CIMI clusters all incomplete objects, maximizing the utilization of available object information and avoiding the risk of information loss that occurs in most methods using only complete objects for imputation. Experimental results demonstrate that CIMI is capable of appropriately searching for optimal clustering tasks with imputing missing values. Furthermore, this paper focuses on the clustering performance of the data after imputation, exploring the impact of imputation methods on clustering outcomes. This provides a more insightful hint for researchers who consider clustering as a downstream task when selecting an appropriate imputation method.

References

1. Hornung R, Ludwigs F, Hagenberg J, Boulesteix AL. Prediction approaches for partly missing multi-omics covariate data: A literature review and an empirical comparison study. *WIREs Computational Statistics.* 2023;15(5):e1626.
2. Wei R, Wang J, Su M, et al. Missing value imputation approach for mass spectrometry-based metabolomics data. *Scientific Reports.* 2018;8(1):663.
3. Nekouie A, Moattar MH. Missing value imputation for breast cancer diagnosis data using tensor factorization improved by enhanced reduced adaptive particle swarm optimization. *Journal Of King Saud University-Computer And Information Sciences.* 2019;31(3):287–294.
4. Brock GN, Shaffer JR, Blakesley RE, Lotz MJ, Tseng GC. Which missing value imputation method to use in expression profiles: a comparative study and two selection schemes. *BMC Bioinformatics.* 2008;9(12):1–12.
5. Laña I, Olabarrieta II, Vélez M, Del Ser J. On the imputation of missing data for road traffic forecasting: New insights and novel techniques. *Transportation Research Part C: Emerging Technologies.* 2018;90:18–33.
6. Demirhan H, Renwick Z. Missing value imputation for short to mid-term horizontal solar irradiance data. *Applied Energy.* 2018;225:998–1012.
7. Washington BJ, Seymour L. An adapted vector autoregressive expectation maximization imputation algorithm for climate data networks. *WIREs Computational Statistics.* 2019;12(3):e1494.
8. Little RJ, Rubin DB. *Statistical Analysis With Missing Data*. 793. John Wiley & Sons, 2019.
9. Rässler S, Rubin DB, Zell ER. Imputation. *WIREs Computational Statistics.* 2013;5(1):20–29.
10. Zhang Z, Zhang Y, Zeng A, et al. Time-Series Data Imputation via Realistic Masking-Guided Tri-Attention Bi-GRU. In: European Conference on Artificial Intelligence. 2023:3074–3082.
11. Feng S, Zhao M, Huang Z, Ji Y, Zhang Y, Cheung YM. Robust Qualitative Data Clustering via Learnable Multi-Metric Space Fusion. In: International Conference on Acoustics, Speech and Signal Processing. 2024.
12. Zhao Y, Long Q. Variable Selection in the Presence of Missing Data: Imputation-based Methods. *Wiley Interdisciplinary Reviews: Computational Statistics.* 2017;9(5):e1402.
13. Jerez JM, Molina I, García-Laencina PJ, et al. Missing data imputation using statistical and machine learning methods in a real breast cancer problem. *Artificial Intelligence In Medicine.* 2010;50(2):105–115.
14. Myers TA. Goodbye, listwise deletion: Presenting hot deck imputation as an easy and effective tool for handling missing data. *Communication Methods And Measures.* 2011;5(4):297–310.

15. Pedersen AB, Mikkelsen EM, Cronin-Fenton D, et al. Missing data and multiple imputation in clinical epidemiological research. *Clinical Epidemiology.* 2017;9:157–166.

16. Zhang K, Gonzalez R, Huang B, Ji G. Expectation–maximization approach to fault diagnosis with missing data. *IEEE Transactions On Industrial Electronics.* 2014;62(2):1231–1240.

17. Troyanskaya O, Cantor M, Sherlock G, et al. Missing value estimation methods for DNA microarrays. *Bioinformatics.* 2001;17(6):520–525.

18. Singh N, Javeed A, Chhabra S, Kumar P. Missing value imputation with unsupervised kohonen self organizing map. In: Emerging Research in Computing, Information, Communication and Applications. 2015:61–76.

19. Quinlan JR. *C4.5: Programs for Machine Learning*. 16. Springer, 1993.

20. Oba S, Sato Ma, Takemasa I, Monden M, Matsubara Ki, Ishii S. A Bayesian missing value estimation method for gene expression profile data. *Bioinformatics.* 2003;19(16):2088–2096.

21. Zhang Y, Cheung Ym. Learnable weighting of intra-attribute distances for categorical data clustering with nominal and ordinal attributes. *IEEE Transactions On Pattern Analysis And Machine Intelligence.* 2021;44(7):3560–3576.

22. Azur MJ, Stuart EA, Frangakis C, Leaf PJ. Multiple imputation by chained equations: what is it and how does it work. *International Journal Of Methods In Psychiatric Research.* 2011;20(1):40–49.

23. Stekhoven DJ, Bühlmann P. MissForestnon-parametric missing value imputation for mixed-type data. *Bioinformatics.* 2012;28(1):112–118.

24. Zhang Y, Cheung YM. Graph-based dissimilarity measurement for cluster analysis of any-type-attributed data. *IEEE Transactions On Neural Networks And Learning Systems.* 2022;34(9):6530–6544.

25. Kim H, Golub GH, Park H. Missing value estimation for DNA microarray gene expression data: local least squares imputation. *Bioinformatics.* 2005;21(2):187–198.

26. Zhang S. Nearest neighbor selection for iteratively kNN imputation. *Journal Of Systems And Software.* 2012;85(11):2541–2552.

27. Yang F, Du J, Lang J, Lu L, Jin C, Kang Q. Missing Value Estimation Methods Research for Arrhythmia Classification Using the Modified Kernel Difference-Weighted KNN Algorithms. *BioMed Research International.* 2020;2020(1):7141725.

28. Pati SK, Das AK. Missing value estimation for microarray data through cluster analysis. *Knowledge And Information Systems.* 2017;52:709–750.

29. Raja P, Thangavel K. Missing value imputation using unsupervised machine learning techniques. *Soft Computing.* 2020;24(6):4361–4392.

30. Di Nuovo AG. Missing data analysis with fuzzy C-Means: A study of its application in a psychological scenario. *Expert Systems With Applications.* 2011;38(6):6793–6797.

31. Tang F, Ishwaran H. Random forest missing data algorithms. *Statistical Analysis And Data Mining: The ASA Data Science Journal.* 2017;10(6):363–377.

32. Minakshi G. Missing value imputation in multi attribute data set. *International Journal Of Computer Science And Information Technologies.* 2014;5(4):1–7.

33. Pawlak Z. Rough sets. *International Journal Of Computer & Information Sciences.* 1982;11:341–356.

34. Korrapati RB, Divakar C, Devi GL, et al. Rough set theory based missing value imputation. *Cognitive Science And Health Bioinformatics: Advances And Applications.* 2018:97–106.

35. Zhang W. Association-based multiple imputation in multivariate datasets: A summary. In: International Conference on Data Engineering. 2000.

36. Zhu X, Zhang S, Jin Z, Zhang Z, Xu Z. Missing value estimation for mixed-attribute data sets. *IEEE Transactions On Knowledge And Data Engineering.* 2010;23(1):110–121.

37. Rubin DB. Multiple imputations in sample surveys-a phenomenological Bayesian approach to nonresponse. In: American Statistical Association Alexandria, VA. 1978:20–34.

38. Little RJ, Schluchter MD. Maximum likelihood estimation for mixed continuous and categorical data with missing values. *Biometrika.* 1985;72(3):497–512.

39. Rigatti SJ. Random forest. *Journal of Insurance Medicine.* 2017;47(1):31–39.

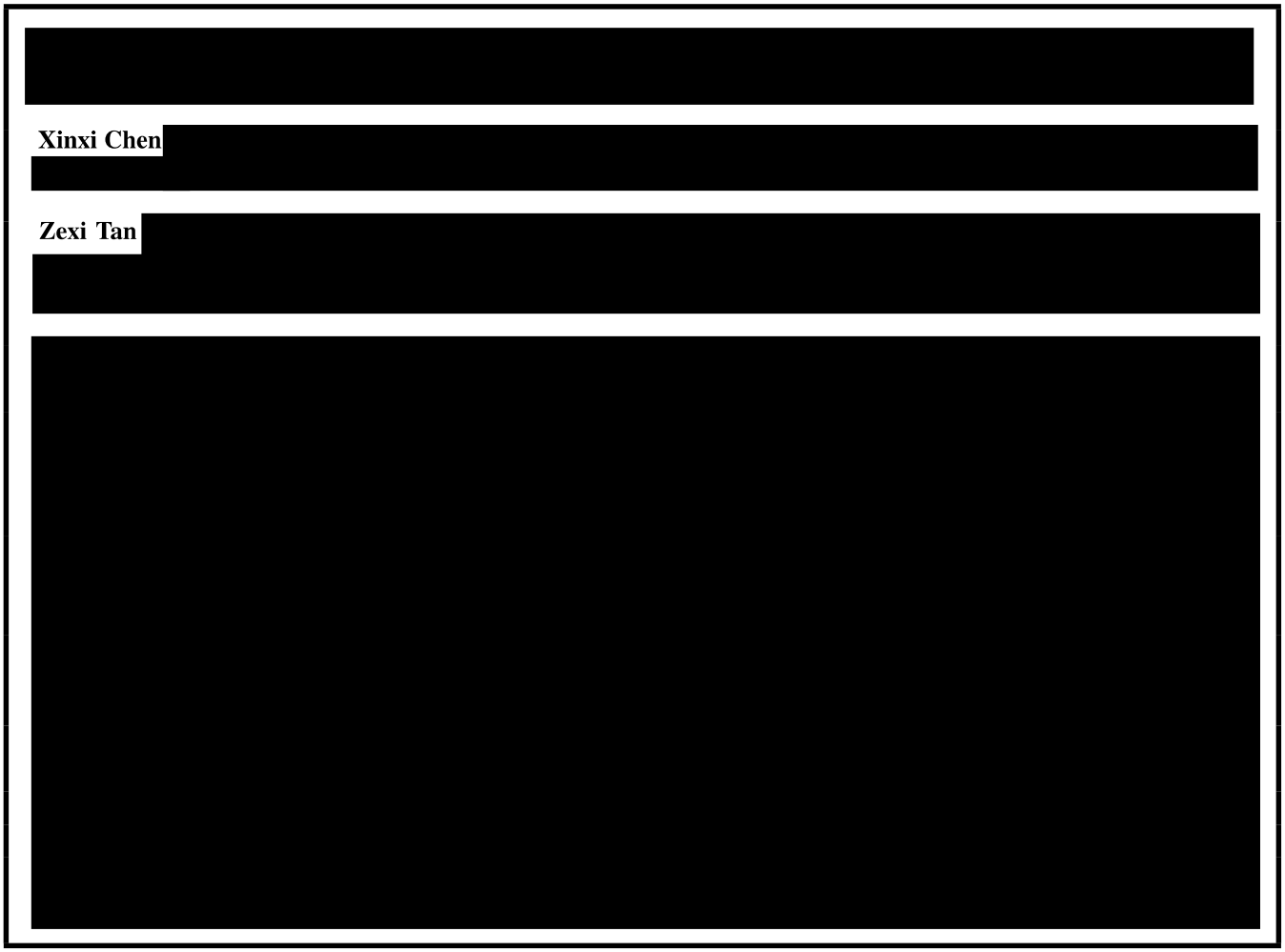
40. Loh WY. Classification and regression trees. *Wiley Interdisciplinary Reviews: Data Mining And Knowledge Discovery.* 2011;1(1):14–23.

41. Nikfalazar S, Yeh CH, Bedingfield S, Khorshidi HA. Missing data imputation using decision trees and fuzzy clustering with iterative learning. *Knowledge And Information Systems.* 2020;62:2419–2437.

42. Zhang Y, Cheung Ym, Zeng A. Het2Hom: Representation of Heterogeneous Attributes into Homogeneous Concept Spaces for Categorical-and-Numerical-Attribute Data Clustering. In: International Joint Conference on Artificial Intelligence. 2022:3758–3765.

43. Chen J, Ji Y, Zou R, Zhang Y, Cheung Ym. QGRL: quaternion graph representation learning for heterogeneous feature data clustering. In: Knowledge Discovery and Data Mining. 2024:297–306.
44. Zhu Q, Feng J, Huang J. Natural neighbor: A self-adaptive neighborhood method without parameter K. *Pattern Recognition Letters.* 2016;80:30–36.
45. Dixon JK. Pattern recognition with partly missing data. *IEEE Transactions On Systems, Man, And Cybernetics.* 1979;9(10):617–621.
46. Hruschka ER, Hruschka ER, Ebecken NF. Towards efficient imputation by nearest-neighbors: A clustering-based approach. In: Advances in Artificial Intelligence. 2005:513–525.
47. Lall U, Sharma A. A nearest neighbor bootstrap for resampling hydrologic time series. *Water Resources Research.* 1996;32(3):679–693.
48. Hartigan JA, Wong MA. Algorithm AS 136: A k-means clustering algorithm. *Journal Of The Royal Statistical Society. Series C (Applied Statistics).* 1979;28(1):100–108.
49. Huang Z. Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data Mining And Knowledge Discovery.* 1998;2(3):283–304.
50. Huang Z. Clustering Large Data Sets With Mixed Numeric And Categorical Values. In: Knowledge Discouvery And Data Mining. 1997.
51. Gu Y, Zhang S, Qiu L, Wang Z, Zhang L. A layered KNN-SVM approach to predict missing values of functional requirements in product customization. *Applied Sciences.* 2021;11(5):2420.
52. Hubert L, Arabie P. Comparing partitions. *Journal Of Classification.* 1985;2:193–218.
53. Starczewski A, Krzyżak A. Performance Evaluation of the Silhouette Index. In: Artificial Intelligence and Soft Computing. 2015:49–58.

AUTHOR BIOGRAPHY



Xinxi Chen

Zexi Tan

Asynchronous Federated Clustering with Unknown Number of Clusters

Yunfan Zhang[1] [redacted]

Abstract

Federated Clustering (FC) is crucial to mining knowledge from unlabeled non-Independent Identically Distributed (non-IID) data provided by multiple clients while preserving their privacy. Most existing attempts learn cluster distributions at local clients, then securely pass the desensitized information to the server for aggregation. However, some tricky but common FC problems are still relatively unexplored, including the heterogeneity in terms of clients' communication capacity and the unknown number of proper clusters. To further bridge the gap between FC and real application scenarios, this paper first shows that the clients' communication asynchrony and unknown proper cluster numbers are complex coupling problems, and then proposes an Asynchronous Federated Cluster Learning (AFCL) method accordingly. It spreads the excessive number of seed points to clients as a learning medium and coordinates them across clients to form a consensus. To alleviate the distribution imbalance cumulated due to the unforeseen asynchronous uploading from the heterogeneous clients, we also design a balancing mechanism for seeds updating. As a result, the seeds gradually adapt to each other to reveal a proper number of clusters. Extensive experiments demonstrate the efficacy of AFCL.

Code [redacted]

Introduction

Federated Learning (FL) is common in implementing distributed machine learning while preserving privacy (Banabilah et al. 2022; Zhang et al. 2021; Zhou et al. 2023). In unsupervised FL tasks, Federated Clustering (FC) that partitions a dataset into compact object clusters demonstrates great potential in mining data concepts and knowledge (Ma et al. 2019; Nelus, Glitza, and Martin 2021; Chung, Lee, and Ramchandran 2022). However, without label guidance, FC faces significant challenges brought by the privacy protection requirements and non-IID of heterogeneous clients.

Most existing methods address FC by first letting clients learn cluster distributions and then passing the privacy-protected cluster knowledge to the server for global aggregation (Ghosh et al. 2020; Kumar, Karthik, and Nair 2020). For example, k-FED (Dennis, Li, and Smith 2021) explores global cluster distributions with a higher security level through one-shot aggregation of the non-IID distributions learned by the clients. Two independent works, F-FCM (Pedrycz 2022) and FFCM (Stallmann and Wilbik 2022), share similar names and principles, and adopt fuzzy-c-means as their local clustering algorithm. Since the fuzzy object-cluster affiliation can more finely reflect the partition information of data objects, the information loss caused by the privacy constraints of FL can be considerably offset.

Notably, most existing works overlook the common problem of client asynchrony[1] attributed to the divergence of clients' communication capabilities. Due to the lack of data

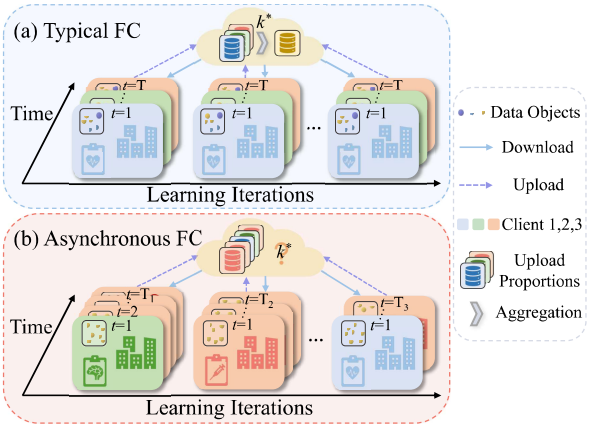


Figure 1: **AFCL (ours) vs. typical FC.** (a) Existing FC approaches typically assume that the clients are synchronous and the optimal k^* is known by both the clients and server. By contrast, (b) AFCL learns under a more realistic scenario that the clients can upload distribution information of completely non-overlapping and non-uniform numbers of clusters with unforeseen and imbalanced frequencies.

[1]The terms "asynchrony" and "asynchronous communication" in this paper indicate the non-uniform participation rates of different clients in each round of communication. Since the goal of FC is to aggregate the distributions of clients at the server, the asynchronous problem addressed in this work is conceptually different from the model update asynchronous problem in supervised FL.

*Coresponding Author

labels, such a problem can severely bias the learning towards the clients that upload their distributions more frequently. Accordingly, a recently proposed federated spectral clustering method (Qiao, Ding, and Fan 2024) copes with the asynchronous scenario by aggregating intermediate variables to construct a global similarity matrix, which is robust to the communication frequency bias. However, it still relies on the naive assumption that the true number of clusters k^* is known in advance, which limits the application domain of FC in many real applications with unforeseen k^*.

Automatic determination of the optimal k^* has been an attractive clustering research topic in recent decades. Silhouette Coefficient (Rousseeuw 1987) determines k^* by considering the intra-compactness and the inter-dispersion of clusters. Density-based clustering (Schubert et al. 2017; Zhang et al. 2025; Peng et al. 2025) performs cluster exploration by considering the distribution connection of objects, and can select k^* according to the quality of cluster partition. More advanced learning-based approaches (Ahalt et al. 1990; Cheung, Y.-m. 2005; Jia, Cheung, and Liu 2014; Cai et al. 2024; Zou et al. 2024) have been proposed to learn k^* by letting seeds compete or cooperate to eliminate redundant low-quality clusters. Recently, significance-based clustering (Hu et al. 2022, 2025) has been proposed using gap statistics to estimate k^*. Nevertheless, their learning process depends on detailed and sufficient data statistics, preventing them from being utilized in FC. Therefore, the absence of k^* brings difficulties to FC due to the lack of clustering guidance, and also incurs additional k^* learning objective, collectively making asynchronous FC a challenging task.

A more detailed comparison of asynchronous FC and typical FC are illustrated in Figure. 1. Unlike typical FC where the clients communicate synchronously with known k^*, our focused asynchronous FC poses the problem of learning clusters and cluster numbers from the uploaded complex distributions caused by the non-uniform communication of clients. More specifically, the difficulties of asynchronous FC lie in the cross-coupled k^* learning and asynchronous uploading of clients. That is, distributions learned at different clients are not associated with a uniform k^*, while the server struggles in the learning of k^* due to the asynchronously uploaded unreliable local distributions.

Therefore, we propose the Asynchronous Federated Cluster Learning (AFCL) method that can learn an optimal number of clusters with the asynchronously communicated clients in a self-adaptive way. It uniformly generates seed points for the clients, and accumulates the distribution information of their surrounding objects indicated by the difference between the seed and objects within each client to capture the clients' own distributions. Then the accumulated information is passed to the server to update the seeds for client-to-seed distribution information fusion. To gain a consensus among the clients, we let the neighboring seeds share their update intensity to achieve seed-to-seed information completion, as the non-IID clients may provide only a partial global distribution. A balancing mechanism has also been developed to evaluate and adjust the update intensity accumulated from different clients, to relieve the potential bias caused by their asynchronous participation. As a result, AFCL can automatically converge redundant neighboring seeds to learn an appropriate number of clusters under the challenging asynchronous federated scenario. Extensive experiments demonstrate the effectiveness of AFCL. The main contributions of this work are summarized into three points:

- We propose a new FC approach to learn a distribution consensus from asynchronously communicated clients without requiring the 'true' number of clusters. It serves to enhance the robustness and universality of FL.
- This paper first considers and attempts to solve a more realistic but challenging non-IID case in FC, i.e., a global cluster could be composed of completely non-overlapping sub-clusters belonging to different clients.
- A balancing mechanism is developed to allocate the contribution of clients with heterogeneous communication modes during the interactive learning on the server, even if they participate in the learning by only one-shot.

Related Work

Federated Clustering

An one-shot FC approach called k-FED (Dennis, Li, and Smith 2021) has been developed to relieve the leakage of information during communication, while FedKKM (Zhou and Wang 2022) has been proposed to improve communication efficiency by using a novel Lanczos algorithm in its distributed matrices. Meanwhile, F-FCM (Pedrycz 2022), and FFCM (Stallmann and Wilbik 2022) utilized fuzzy clustering techniques to enhance privacy protection by only transmitting object-cluster affiliation. Furthermore, A multi-view FC approach (Hu et al. 2023) has been proposed to extend multi-view clustering into the federated scenario by designing a strategy of consensus prototype learning. However, they employ clustering techniques under the assumption that the true cluster numbers are known by the clients and server in advance. Recently, an FC framework VKMC (Huang et al. 2022) has been proposed to improve the vertical FL based on coresets, while a density-based FC method HFDPC (Ding et al. 2023) has been proposed for improving the effectiveness of data partitioning by introducing a similar density chain. Most recently, Federated Subspace Clustering (Fed-SC) (Xie, Wu, and Liao 2023) and Federated Spectral Clustering (FedSC) (Qiao, Ding, and Fan 2024) have been proposed to address the FC of high-dimensional and noisy data, respectively. Although some of the above-mentioned FC methods have considered the non-IID or the asynchrony issues of clients in FC, most of their solutions heavily depend on the availability of the 'true' number of clusters k^*, which hinders their applications in real complex scenarios.

Clustering with Unknown Cluster Number

The more realistic unsupervised or weakly supervised learning has attracted much attention in recent years, especially for some significant application domains (Cheung and Zeng 2009; Wu et al. 2019; Zhang, Cheung, and Tan 2020; Wang et al. 2023). Clustering is a key unsupervised learning technique, where the traditional clustering methods determine the optimal number of clusters k^* manually (Rousseeuw

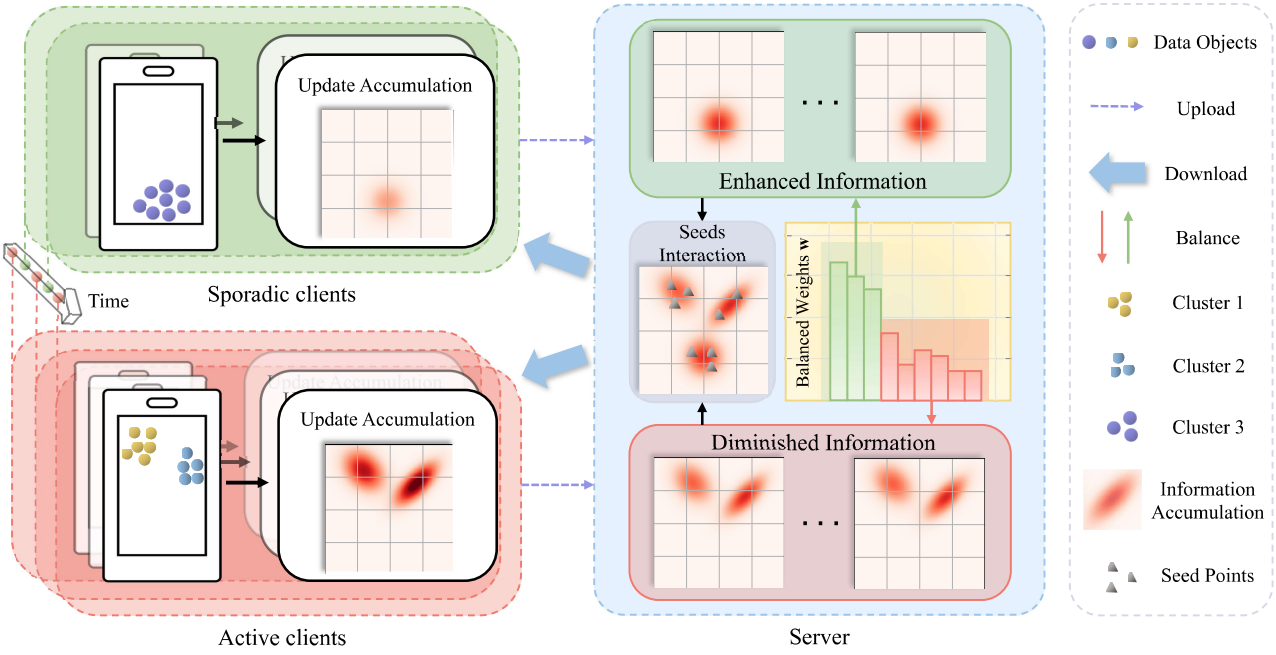


Figure 2: Overview of the proposed AFCL framework. Initialized seed points accumulate update intensity from different clients independently, then the server balances the update information to facilitate appropriate seeds interaction for fusing the clients' distributions. The heat map represents the intensity of update information of seeds accumulated from asynchronous clients.

1987; Thorndike 1953). To realize automated k^* selection, density-based clustering (Ester et al. 1996; Zhang et al. 2025; Peng et al. 2025) have been proposed to automatically determine k^* at a "knee point" during their cluster exploration. Recently, more advanced learning-based approaches (Cheung 2004; Cheung and Jia 2013; Cai et al. 2024) introduce a cooperative or competitive mechanism to excessive cluster centers. They simultaneously ensure the comprehensive representation of object distributions and make the elimination of redundant cluster centers learnable, thus achieving satisfactory clustering performance. Most recently, significance-based approaches (Hu et al. 2022, 2025) have been proposed to rigorously judge the significance of cluster distributions under the current k. Nevertheless, all the above-mentioned solutions require detailed statistics of the entire dataset, which hamper their applicability in FC.

Proposed Method

In this section, we first define the asynchronous FC problem, and then present the proposed AFCL algorithm composed of two key technical components: 1) CSUA: Client-Side Update Accumulation, and 2) SSSI: Server-Side Seeds Interaction. Frequently used notations are summarized in Table 1, and the overview of AFCL is shown in Figure. 2.

Problem Formulation

Assuming a federated network with p clients dividing the entire dataset $\mathbf{X}$ into the corresponding subsets $\{\mathbf{X}^{\{1\}}, \mathbf{X}^{\{2\}}, ..., \mathbf{X}^{\{g\}}, ..., \mathbf{X}^{\{p\}}\}$, where the subset of g-th client $\mathbf{X}^{\{g\}}$ has $n^{\{g\}}$ objects $\{\mathbf{x}_1^{\{g\}}, \mathbf{x}_2^{\{g\}}, ..., \mathbf{x}_{n^{\{g\}}}^{\{g\}}\}$ with $\sum_{g=1}^{p} n^{\{g\}} = n$, and each object $\mathbf{x}_i^{\{g\}} = [x_{i,1}^{\{g\}}, x_{i,2}^{\{g\}}, ..., x_{i,d}^{\{g\}}]^\top$ is a d-dimensional vector. We use a matrix $\mathbf{Q} \in \mathbb{R}^{n \times k}$ to indicate the object-cluster affiliation, and the conventional clustering objective is to minimize the overall intra-cluster dissimilarity between the objects and the cluster center (also called seed point or seed interchangeably hereinafter), which can be written as:

$$Z(\mathbf{Q}, \mathbf{M}) = \sum_{i=1}^{n} \sum_{l=1}^{k} q_{i,l} \Phi(\mathbf{x}_i, \mathbf{m}_l), \tag{1}$$

where $\Phi(\mathbf{x}_i, \mathbf{m}_l)$ denotes the Euclidean distance between i-th object $\mathbf{x}_i$ and l-th seed $\mathbf{m}_l$. All the k seeds can be organized as a matrix $\mathbf{M} \in \mathbb{R}^{d \times k}$ and $q_{i,l}$ is the (i,l)-th entry of $\mathbf{Q}$ satisfying $\sum_{l=1}^{k} q_{i,l} = 1$ and $q_{i,l} \in \{0, 1\}$. For federated

Notations	Explanations
$\mathbf{X}$	Global dataset
$\mathbf{X}^{\{g\}}$	Dataset of g-th client
$\mathbf{M}^{\{g\}}$	Seed points of g-th client
$\mathbf{m}_l$	l-th global seed point
$\mathbf{Q}^{\{g\}}$	Object-cluster affiliation matrix corresponding to g-th client
$\mathbf{B}^{\{g\}}$	Cluster center set of g-th client
$\mathbf{R}_l^{\{g\}}$	Update intensity of l-th seed on g-th client
k^*	The 'true' global cluster number of $\mathbf{X}$

Table 1: Summary of notations.

clustering, each g-th client can perform the above clustering locally on $\mathbf{X}^{\{g\}}$, and the ultimate goal is to minimize the objective function at the server with the entire dataset $\mathbf{X}$ and the consensus seed points $\mathbf{M}$ learned across all the clients.

CSUA: Client-Side Update Accumulation

To protect the privacy of clients while transmitting their distribution information to the server for global clustering, some intermediate quantities, e.g., update intensity of seeds, clustering center, and intra-cluster dissimilarity will be extracted by performing local clustering on each client and then uploaded to the server for seeds interaction.

For each object $\mathbf{x}_i^{\{g\}}$ in g-th client, its belonging cluster is determined by:

$$q_{i,l} = \begin{cases} 1, & \text{if } l = \arg\min_r \gamma_r \parallel \mathbf{x}_i^{\{g\}} - \mathbf{m}_r^{\{g\}} \parallel^2 \\ 0, & \text{otherwise,} \end{cases} \tag{2}$$

where $\mathbf{m}_r^{\{g\}}$ is the r-th seed in $\mathbf{M}^{\{g\}}$, and $\gamma_r^{\{g\}}$ is the weight of $\mathbf{m}_r^{\{g\}}$ computed by:

$$\gamma_r^{\{g\}} = \frac{s_r^{\{g\}}}{\sum_{l=1}^{k} s_l^{\{g\}}}. \tag{3}$$

Here, $s_r^{\{g\}}$ denotes the winning time of $\mathbf{m}_r^{\{g\}}$ within a single iteration, which is updated by:

$$s_c^{\{g\}} = s_c^{\{g\}} + 1, \tag{4}$$

for each $q_{i,c} = 1, i \in \{1, 2, ..., n^{\{g\}}\}$.

To facilitate the interaction of seeds across clients, we also compute the update intensity of seeds locally, but suspend the update until they are uploaded to the server. For the winner seed $\mathbf{m}_c^{\{g\}}$, its update intensity $\mathbf{r}_{c,i}^{\{g\}}$ contributed by the object $\mathbf{x}_i^{\{g\}}$ is computed by:

$$\mathbf{r}_{c,i}^{\{g\}} = \eta(\mathbf{x}_i^{\{g\}} - \mathbf{m}_c^{\{g\}}), \tag{5}$$

where $\mathbf{r}_{c,i}^{\{g\}} \in \mathbf{R}_c^{\{g\}}$ and η is the learning rate. By computing the update intensity of k seeds provided by all the $n^{\{g\}}$ objects, we obtain $R^{\{g\}} = \{\mathbf{R}_1^{\{g\}}, \mathbf{R}_2^{\{g\}}, ..., \mathbf{R}_k^{\{g\}}\}$ to upload to the server.

Remark 1. ***Privacy Protection w.r.t.*** $R^{\{g\}}$***:*** *AFCL uploads* $R^{\{g\}}$*s to the server to facilitate the interaction among seeds for the fusion of clients' information. Since the update intensity of a* r*-th seed provided by each of the objects treating it as a winner (i.e.,* $\mathbf{R}_r^{\{g\}}$*) will be uploaded, the risk of objects recovery and privacy leakage will be increased. Therefore, existing privacy-preserving techniques such as holomorphic encryption (Acar et al. 2018) and differential privacy (Wei et al. 2020; Li et al. 2023) can be incorporated in some scenarios with high privacy protection requirements. Specifically, these techniques can be utilized to perturb the update intensity provided by each object in* $R^{\{g\}}$ *while ensuring that the radius of cooperative seeds selection in Eq.* (13) *and the overall update of seeds in Eq.* (14) *unchanged. Note that this paper focuses on more robust FC, rather than improving its privacy protection level.*

To judge convergence at the server, we also compute the centers $\mathbf{B}^{\{g\}} = \{\mathbf{b}_1^{\{g\}}, \mathbf{b}_2^{\{g\}}, ..., \mathbf{b}_k^{\{g\}}\}$ of the k clusters partitioned by the seeds by:

$$\mathbf{b}_r^{\{g\}} = \frac{1}{o_r^{\{g\}}} \sum_{i=1}^{n^{\{g\}}} q_{i,r} \mathbf{x}_i^{\{g\}}, \tag{6}$$

where $o_r^{\{g\}}$ is the number of objects in the r-th cluster corresponding to $\mathbf{m}_r^{\{g\}}$. Based on $\mathbf{B}^{\{g\}}$, the contribution of the r-th cluster to the global objective Z can be computed by:

$$z_r^{\{g\}} = \sum_{i=1}^{n^{\{g\}}} q_{i,r} \parallel \mathbf{b}_r^{\{g\}} - \mathbf{x}_i^{\{g\}} \parallel^2, \tag{7}$$

and the contributions from all the seeds in g-th client can be collectively denoted as $\mathbf{z}^{\{g\}} = [z_1^{\{g\}}, z_2^{\{g\}}, ..., z_k^{\{g\}}]^\top$.

Remark 2. ***Necessity of Intermediate*** $\mathbf{B}^{\{g\}}$ ***and*** $\mathbf{z}^{\{g\}}$***:*** *Since the trajectories of seeds* $\mathbf{M}$ *is usually complex as shown in Figure. 3(d), directly calculating the objective function based on seeds may lead to constant fluctuations in the objective function value. Therefore, intermediate values* $\mathbf{B}^{\{g\}}$ *and* $\mathbf{z}^{\{g\}}$ *computed locally on each client are relatively stable and can timely reflect the goodness of current seeds in terms of each client, which are helpful to obtain a smooth and more approximate overall objective function value* Z*.*

SSSI: Server-Side Seeds Interaction

During the learning, communication frequencies of each client should be recorded as $\mathbf{\Theta} = [\theta^{\{1\}}, \theta^{\{2\}}, .., \theta^{\{p\}}]^\top$. Accordingly, weights $\mathbf{w} = [w^{\{1\}}, w^{\{2\}}, ..., w^{\{p\}}]^\top$ for balancing the seeds update bias caused by the asynchronous communication should be maintained. For g-th participating client, its balance weight is computed by:

$$w^{\{g\}} = \frac{\xi}{\xi + \frac{\theta^{\{g\}}}{\sum_{j=1}^{p} \theta^{\{j\}}}}, \tag{8}$$

where ξ is a hyper-parameter controlling the sensitivity of balance weight w.r.t. the communication frequencies $\mathbf{\Theta}$. Intuitively, a larger ξ makes the balance weight less sensitive to the frequency, and a client with a higher frequency will have a lower weight to weaken its contribution.

Upon receiving the uploading from clients, the server initially aggregates the cluster centers and their objective contributions with the balance weights by:

$$\mathbf{b}_r = \sum_{j=1}^{\bar{p}} \frac{w^{\{j\}} o_r^{\{j\}} \mathbf{b}_r^{\{j\}}}{\sum_{j=1}^{\bar{p}} o_r^{\{j\}}}, \quad z_r = \sum_{j=1}^{\bar{p}} \frac{w^{\{j\}} o_r^{\{j\}} z_r^{\{j\}}}{\sum_{j=1}^{\bar{p}} o_r^{\{j\}}}, \tag{9}$$

where $\bar{p}$ denotes the number of participating clients. $\mathbf{B} = \{\mathbf{b}_1, \mathbf{b}_2, ..., \mathbf{b}_k\}$ and $\mathbf{z} = [z_1, z_2, ..., z_k]^\top$ represent the aggregated centers and contributions to the objective Z from the server perspective. Accordingly, an approximation of the objective on the server can be derived from Eq. (1) as:

$$Z(\mathbf{B}, \mathbf{z}) = \frac{1}{k} \sum_{l=1}^{k} \max_{r \neq l} \left(\frac{z_l + z_r}{\parallel \mathbf{b}_l - \mathbf{b}_r \parallel^2} \right), \tag{10}$$

which is in the form of the DBI index (Ros, Riad, and Guillaume 2023) that simultaneously reflects cluster compactness (numerator) and clusters' dispersion (denominator).

To ensure that the seeds learned from data can consensually minimize Z, a cooperative set:

$$C_r = \{\mathbf{m}_l \mid\mid \| \mathbf{m}_r - \mathbf{m}_l \|^2 \leq \| \mathbf{m}_r - \mathbf{x}_i^{\{g\}} \|^2\} \tag{11}$$

is identified for each seed $\mathbf{m}_r$, and we let all the seeds in C_r collectively receive the updates from the samples as:

$$\mathbf{m}_u = \mathbf{m}_u + \eta(\mathbf{x}_i^{\{g\}} - \mathbf{m}_u), \tag{12}$$

where $\mathbf{m}_u \in C_r$. However, since original samples are unavailable at the server due to privacy constraints, we compute Eqs. (11) and (12) alternatively based on the update intensity $\mathbf{r}_{r,i}^{\{g\}}$ that has been uploaded to the server as:

$$C_r = \{\mathbf{m}_l \mid\mid \| \mathbf{m}_r - \mathbf{m}_l \|^2 \leq \| \frac{w^{\{g\}}\mathbf{r}_{r,i}^{\{g\}}}{\eta} \|^2\} \tag{13}$$

and

$$\mathbf{m}_u = \mathbf{m}_u + w^{\{g\}}\mathbf{r}_{r,i}^{\{g\}} + w^{\{g\}}\eta(\mathbf{m}_r - \mathbf{m}_u) \tag{14}$$

respectively, where $w^{\{g\}}$ is used to mitigate the bias of seeds update caused by the asynchronous uploading. Intuitively, a smaller $w^{\{g\}}$ corresponds to a client that uploads frequently. Thus its neighbor identifying radius $\| w^{\{g\}}\mathbf{r}_{r,i}^{\{g\}}/\eta \|^2$ in Eq. (13) will be smaller to avoid over-update of the seeds in C_r. In Eq. (14), $\mathbf{r}_{r,i}^{\{g\}}$ from Eq. (5) is already with the learning rate η, and thus its latter two terms are with the same coefficients $w^{\{g\}}\eta$, equivalent to update $\mathbf{m}_u$ by a small step towards the data object that yields the update intensity $\mathbf{r}_{r,i}^{\{g\}}$.

Remark 3. ***Interaction of Seeds:*** *According to Eqs.* (13) *and* (14)*, all seeds in C_r will move closer to the cluster distribution represented by $\mathbf{m}_r$ according to the distribution information accumulated on different clients, which facilitates the distribution completion across different clients.*

Overall AFCL Algorithm

In the AFCL algorithm, after the global initialization of the seed points and local update accumulation on all the clients, sufficient interaction among the clients through the server is iteratively performed until the convergence of Z. In this process (summarized as Algorithm 1), the client-side (indicated by pink color) mainly implements cluster distribution learning, and the server-side (indicated by blue color) is responsible for privacy-protected distribution information fusion. For each learning iteration of the AFCL algorithm, the time complexity is $\mathcal{O}(kn^{\{g\}}dp + n^{\{g\}}k^2d)$, which is efficient compared to the state-of-the-art FC methods. A more detailed analysis can be found in the Appendix.

Appendix — https://github.com/Yunfan-Zhang/AFCL

Algorithm 1: AFCL: Asynchronous Federated Clustering.

Input: $\mathbf{X}^{\{1\}}, \mathbf{X}^{\{2\}}, ..., \mathbf{X}^{\{p\}}$, k, ξ, η.

for all clients in parallel **do**
  Initialize k seeds using k-means++;
  Transfer initialized seeds to the server;
end for
Initialize k global seeds $\mathbf{M}$ using k-means++ according to the seeds received from all clients;
repeat
  Transfer global seeds $\mathbf{M}$ to participating clients;
  for g**-th participating client** in parallel **do**
    Update $\theta^{\{g\}} = \theta^{\{g\}} + 1$;
    Compute $\mathbf{Q}^{\{g\}}, R^{\{g\}}, \mathbf{B}^{\{g\}}, \mathbf{z}^{\{g\}}$, by Eqs. (2), (5), (6), and (7), respectively;
    Upload $\mathbf{Q}^{\{g\}}, R^{\{g\}}, \mathbf{B}^{\{g\}}, \mathbf{z}^{\{g\}}$ to the server;
  end for
  Compute $\mathbf{M}$, Z using Eqs. (14) and (10);
  Update $\mathbf{w}$ by Eq. (8);
until Convergence

Output: $\mathbf{M}$, $\mathbf{Q}$.

No.	Datasets	Abbrev.	n	d	k^*
1	Synthetic Dataset 1	SD1	2300	2	4
2	Synthetic Dataset 2	SD2	2900	2	5
3	Seeds	SE	210	7	3
4	Iris	IR	150	4	3
5	Avila	AL	10430	10	12
6	Abalone	AB	4177	7	29
7	Breast Cancer	CC	569	30	2
8	Accent	AC	329	12	6
9	Segment	SG	2100	19	7
10	Live	LI	7051	9	2
11	Parkinson	PA	197	22	2
12	Audit	AU	776	24	2
13	Transfusion	TF	748	4	2

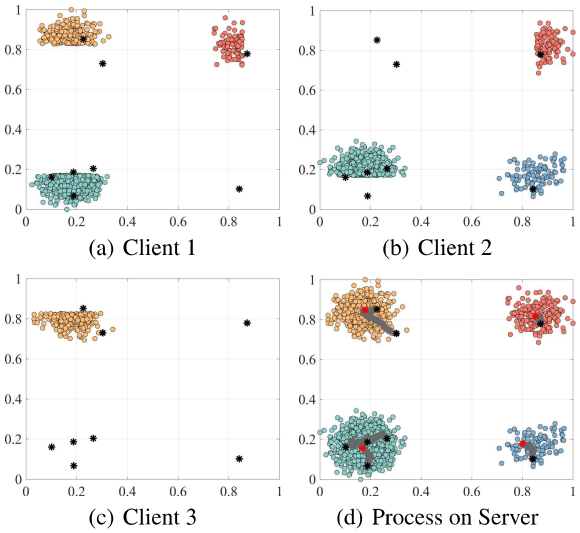
Table 2: Statistics of experimental datasets.

Experiments

Experimental Setup

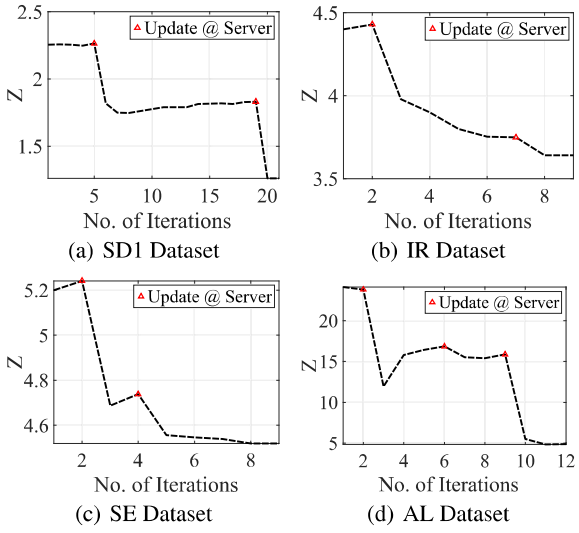
Six experiments have been conducted: (1) Visualization: Intuitive demonstration of the learning process of the proposed AFCL; (2) Convergence Evaluation: Objective function values at each learning iteration are recorded to illustrate the convergence and efficiency of AFCL; (3) Clustering Performance Evaluation: We compare AFCL with the conventional and state-of-the-art counterparts to demonstrate the superiority of AFCL; Due to space limitation, the remainder three experiments, i.e., significance test, ablation study, execution time evaluation, and more detailed experimental settings, are provided in the online appendix.

Six counterparts are compared: DK++ (Bahmani et al. 2012) is a conventional distributed learning approach that conforms to the settings of FL. Five state-of-the-art methods, i.e., the iterative learning approaches FFCM-avg1, FFCM-avg2 (Stallmann and Wilbik 2022), and FedSC (Qiao, Ding,



(a) Client 1 (b) Client 2 (c) Client 3 (d) Process on Server

Figure 3: Seed points and their trajectories on the server during the learning of AFCL. Black and red dots indicate the initial and final positions of the seed points, respectively.



(a) SD1 Dataset (b) IR Dataset (c) SE Dataset (d) AL Dataset

Figure 4: Values of the AFCL objective function on (a) SD1, (b) IR, (c) SE, and (d) AL datasets. Red triangles mark the iterations that the server update starts.

and Fan 2024), and the one-shot learning approaches k-FED (Dennis, Li, and Smith 2021) and Fed-SC (Xie, Wu, and Liao 2023), are compared. All their hyper-parameters (if any) are set according to the corresponding source papers.

13 datasets, including two Gaussian spherical synthetic datasets, and 11 public real datasets collected from the UCI machine learning repository (Asuncion and Newman 2007), are utilized for the experiments, and their statistics are shown in Table 2. All the real datasets are pre-processed by omitting the objects with missing values and normalized.

Three validity indices including the Silhouette Coefficient index (SC) (Rousseeuw 1987), Calinski-Harabasz index (CH) (Caliński and Harabasz 1974), and Bonferroni–Dunn (BD) test (Demšar 2006) are chosen for performance evaluation. Values of SC and CH are in the intervals [-1,1] and (0,$+\infty$), respectively, with a higher value indicating a better clustering performance. These two internal indices are insensitive to the number of clusters, thus facilitating a fair comparison of AFCL that learns its own k and the counterparts with a pre-set k^* as shown in Table 2. BD test is conducted on the performance ranks of the counterparts to validate if AFCL performs significantly better.

Visualization

To intuitively validate the effectiveness of AFCL, we split SD1 into three subsets for creating three extremely non-IID clients as shown in Figure. 3(a) - (c). Figure. 3(d) shows the global data distributions and update trajectories of the seeds on the server. It can be observed that even though the three clients have completely non-overlapping distributions, AFCL can still appropriately learn a set of seeds to represent the global cluster distributions. The trajectories also demonstrate that the seeds updating mechanism of AFCL can effectively facilitate interaction among the seeds with imbalanced update information uploaded by the asynchronous clients. After a certain number of learning iterations, redundant seeds are homogenized, i.e., overlapped at the center of several prominent clusters. This intuitively demonstrates the autonomous cluster number selection ability of AFCL.

To demonstrate the convergence efficiency of AFCL, we plot its objective function values on four datasets in Figure. 4. It can be observed that AFCL converges quickly with around 10 iterations in most cases. Moreover, the objective function always experiences a steep decline after the server updates, confirming that the designed seeds interaction mechanism is highly effective. It is also noteworthy that, since only limited statistics are permitted to be communicated between clients and server, a strict gradient descent cannot be guaranteed and thus the convergence curve in Figure. 4 is not monotonically decreasing. Such an effect is rational for FC because the clustering objective can be viewed as heterogeneous at different clients and server.

Clustering Performance Evaluation

The clustering performance of AFCL and the existing FC approaches are compared under the non-IID and asynchronous scenarios. For each dataset, we implement clustering by using each compared method by 20 times and report the average performance. For each of the 20 trails, we first implement k-means with $k = 5$, to divide the whole dataset into five clients to simulate the extreme non-overlapping distributions of non-IID clients. Then to simulate the asynchronous participation of clients, we randomly set each client with a different participation probability for controlling their upload in each iteration during the learning. As AFCL does not require k^*, the initial number of seed points k is randomly selected from the range $[k^*, 2k^*]$. The clustering performance in terms of SC and CH obtained un-

Dataset	DK++	k-FED	FFCM-avg1	FFCM-avg2	Fed-SC	FedSC	AFCL
SD1	0.5986±0.18	0.8494±0.00	0.5063±0.02	0.5036±0.02	0.8261±0.09	0.8539±0.00	**0.9714±0.00**
SD2	0.6127±0.11	0.7699±0.00	0.4773±0.03	0.4679±0.03	0.6005±0.14	0.7477±0.00	**0.8571±0.00**
SE	0.3229±0.00	0.3754±0.04	0.4323±0.05	0.4323±0.05	0.3619±0.00	0.3774±0.00	**0.5033±0.09**
IR	0.4955±0.01	0.4818±0.02	0.5672±0.02	0.6119±0.21	0.5384±0.04	0.5719±0.00	**0.6386±0.06**
AL	0.2096±0.02	0.0979±0.03	0.2721±0.09	0.2761±0.07	0.4422±0.05	0.3187±0.00	**0.6138±0.38**
AB	0.2314±0.02	0.1853±0.01	0.3664±0.34	0.4863±0.26	0.3009±0.03	0.3865±0.06	**0.5005±0.11**
CC	0.3778±0.00	0.3809±0.02	0.2873±0.05	0.3051±0.04	0.3672±0.04	0.3747±0.00	**0.5916±0.04**
AC	0.1831±0.02	0.0992±0.01	0.2656±0.13	0.2358±0.13	0.1376±0.02	0.1922±0.00	**0.4851±0.26**
SG	0.3197±0.02	0.3117±0.00	0.3819±0.11	0.3865±0.10	0.2677±0.04	0.2935±0.00	**0.6086±0.12**
LI	0.8241±0.00	0.8256±0.01	0.4805±0.16	0.4239±0.23	0.7980±0.10	0.8252±0.00	**0.8967±0.01**
PA	0.2763±0.00	0.4517±0.03	0.4419±0.15	0.4419±0.15	0.2443±0.00	**0.5138±0.00**	0.4903±0.11
AU	0.4046±0.05	0.3601±0.04	0.1440±0.08	0.2239±0.02	0.3411±0.07	0.4296±0.00	**0.5191±0.07**
TF	0.4935±0.00	0.5269±0.01	0.6402±0.16	0.6402±0.16	0.5172±0.25	0.5390±0.00	**0.6813±0.03**
Ave. Rank	4.6154	4.2308	4.4231	4.1923	5.4615	4.0000	**1.0769**

Table 3: Clustering performance evaluated by SC on all the 13 datasets.

Dataset	DK++	k-FED	FFCM-avg1	FFCM-avg2	Fed-SC	FedSC	AFCL
SD1	13933.3423	18933.7121	3136.2342	3140.0656	18901.0798	18958.6529	**19482.8610**
SD2	13187.1700	16936.0195	1462.2227	1474.5426	1575.4188	16913.4103	**17200.3823**
SE	192.6124	**251.1952**	83.3635	83.3635	193.6615	206.0449	230.9555
IR	**315.2151**	232.9987	65.6174	66.3026	211.5583	232.9386	310.7035
AL	1524.2576	1559.4321	1607.4565	1809.4353	2260.6359	1524.2576	**4880.6220**
AB	3756.1887	3791.0247	3821.2626	3890.4190	3072.4473	3244.3141	**5906.3378**
CC	202.4573	**290.0258**	104.0924	107.4325	222.6533	231.4785	231.5775
AC	112.3225	80.9425	82.4193	77.3857	73.2801	99.9817	**140.7156**
SG	973.7952	1121.9421	765.2036	1278.2297	781.12178	1057.1972	**1541.6809**
LI	5013.7432	5526.4145	1050.4084	806.5331	4075.7079	3608.0879	**6090.9982**
PA	40.2442	68.3886	79.5121	79.5121	56.2801	**83.3207**	78.1141
AU	304.9842	251.1567	65.7637	107.1965	201.6834	**433.3965**	306.7529
TF	393.1126	540.1820	436.6372	436.6372	420.2884	518.4788	**651.6017**
Ave. Rank	4.5000	3.0769	5.5769	4.8846	5.0000	3.4231	**1.5385**

Table 4: Clustering performance evaluated by CH on all the 13 datasets.

der the above settings is shown in Tables 3 and 4. The best and the second-best results are highlighted using boldface and underline, respectively. The 'Ave. Rank' rows report the average rank of different approaches across all datasets.

It can be observed that AFCL outperforms all its counterparts in general, indicating its superiority in asynchronous FC. Specifically, AFCL performs the best on almost all datasets w.r.t. the SC index, except on the PA dataset where AFCL still performs the second best. This is because AFCL can effectively minimize the intra-cluster dissimilarity and maximize the inter-cluster dispersion to search for the global optimal seeds. For the CH index, although AFCL performs the second-best on some datasets, i.e., SE, IR, CC, and AU, the best counterparts differ on these datasets while AFCL remains competitive in most cases. More specifically, for the cases where AFCL does not perform the best, the performance gap between AFCL and the best-performing counterpart is usually tiny, which demonstrates the effectiveness and robustness of AFCL on different datasets.

Concluding Remarks

This paper proposes a new FC approach called AFCL for mining global cluster distributions upon heterogeneous data distributions of asynchronously communicated clients. It advances FC to a more challenging but realistic scenario, i.e., clients can participate in the client-to-server uploading asynchronously, and all the clients and server can be extremely non-IID without knowing the 'true' number of clusters. AFCL achieves this by adopting a client-to-seed information fusion framework, which lets the seed points cooperate on the server to complete the non-IID distributions of clients and automatically learns to eliminate redundant seeds as well. A balance mechanism is also designed to relieve the non-uniform of the update information uploaded by the asynchronously participated clients. As a result, AFCL can effectively outline the global cluster distributions upon the seeds learned by the aggregated update intensity received from clients, even if the communication is extremely asynchronous and the distributions of clients are completely divergent. Comprehensive experiments have illustrated the efficacy of AFCL. Despite the superiority of AFCL, there are still some noteworthy potential limitations. That is, we assume FC on pure numerical data and the number of clients is relatively small. The next promising avenue would be the FC of datasets comprising both numerical and categorical attributes distributed on a large number of clients.

Acknowledgments

References

Acar, A.; Aksu, H.; Uluagac, A. S.; and Conti, M. 2018. A survey on homomorphic encryption schemes: Theory and implementation. *ACM Computing Surveys*, 51(4): 1–35.

Ahalt, S. C.; Krishnamurthy, A. K.; Chen, P.; and Melton, D. E. 1990. Competitive learning algorithms for vector quantization. *Neural Networks*, 3(3): 277–290.

Asuncion, A.; and Newman, D. 2007. UCI Machine Learning Repository.

Bahmani, B.; Moseley, B.; Vattani, A.; Kumar, R.; and Vassilvitskii, S. 2012. Scalable k-means+. *In 2021 Very Large Data Base Endowment*, (7): 1–12.

Banabilah, S.; Aloqaily, M.; Alsayed, E.; Malik, N.; and Jararweh, Y. 2022. Federated learning review: Fundamentals, enabling technologies, and future applications. *Information Processing & Management*, 59(6): 103061.

Cai, S.; Zhang, Y.; Luo, X.; Cheung, Y.-m.; Jia, H.; and Liu, P. 2024. Robust categorical data clustering guided by multi-granular competitive learning. In *2024 International Conference on Distributed Computing Systems*, 288–299.

Caliński, T.; and Harabasz, J. 1974. A dendrite method for cluster analysis. *Communications in Statistics-Theory and Methods*, 3(1): 1–27.

Cheung, Y.-m. 2004. A competitive and cooperative learning approach to robust data clustering. In *Neural Networks and Computational Intelligence*, 131–136.

Cheung, Y.-m.; and Jia, H. 2013. Categorical-and-numerical-attribute data clustering based on a unified similarity metric without knowing cluster number. *Pattern Recognition*, 46(8): 2228–2238.

Cheung, Y.-m.; and Zeng, H. 2009. Local kernel regression score for selecting features of high-dimensional data. *IEEE Transactions on Knowledge and Data Engineering*, 21(12): 1798–1802.

Cheung, Y.-m. 2005. On rival penalization controlled competitive learning for clustering with automatic cluster number selection. *IEEE Transactions on Knowledge and Data Engineering*, 17(11): 1583–1588.

Chung, J.; Lee, K.; and Ramchandran, K. 2022. Federated unsupervised clustering with generative models. In *2022 International Workshop on Trustable, Verifiable and Auditable Federated Learning*, 1–9.

Demšar, J. 2006. Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research*, 7: 1–30.

Dennis, D. K.; Li, T.; and Smith, V. 2021. Heterogeneity for the win: One-shot federated clustering. In *2021 International Conference on Machine Learning*, 2611–2620.

Ding, S.; Li, C.; Xu, X.; Guo, L.; Ding, L.; and Wu, X. 2023. Horizontal federated density peaks clustering. *IEEE Transactions on Neural Networks and Learning Systems*, 1–10.

Ester, M.; Kriegel, H.-P.; Sander, J.; and Xu, X. 1996. Density-based spatial clustering of applications with noise. In *1996 International Conference on Knowledge Discovery and Data Mining*, 6, 1–5.

Ghosh, A.; Chung, J.; Yin, D.; and Ramchandran, K. 2020. An efficient framework for clustered federated learning. In *2020 Advances in Neural Information Processing Systems*, 19586–19597.

Hu, L.; Jiang, M.; Liu, X.; and He, Z. 2025. Significance-based decision tree for interpretable categorical data clustering. *Information Sciences*, 690: 121588.

Hu, L.; Jiang, M.; Liu, Y.; and He, Z. 2022. Significance-based categorical data clustering. *arXiv preprint arXiv:2211.03956*, 1–17.

Hu, X.; Qin, J.; Shen, Y.; Pedrycz, W.; Liu, X.; and Liu, J. 2023. An efficient federated multi-view fuzzy c-means clustering method. *IEEE Transactions on Fuzzy Systems*, 1886–1899.

Huang, L.; Li, Z.; Sun, J.; and Zhao, H. 2022. Coresets for vertical federated learning: Regularized linear regression and k-means clustering. In *2022 Advances in Neural Information Processing Systems*, 29566–29581.

Jia, H.; Cheung, Y.-m.; and Liu, J. 2014. Cooperative and penalized competitive learning with application to kernel-based clustering. *Pattern Recognition*, 47(9): 3060–3069.

Kumar, H. H.; Karthik, V.; and Nair, M. K. 2020. Federated k-means clustering: A novel edge ai based approach for privacy preservation. In *2020 International Conference on Cloud Computing in Emerging Markets*, 52–56.

Li, Y.; Wang, S.; Chi, C.-Y.; and Quek, T. Q. 2023. Differentially private federated clustering over non-IID data. *IEEE Internet of Things Journal*, 6705–6721.

Ma, J.; Zhang, Q.; Lou, J.; Ho, J. C.; Xiong, L.; and Jiang, X. 2019. Privacy-preserving tensor factorization for collaborative health data analysis. In *2019 International Conference on Information and Knowledge Management*, 1291–1300.

Nelus, A.; Glitza, R.; and Martin, R. 2021. Unsupervised clustered federated learning in complex multi-source acoustic environments. In *2021 European Signal Processing Conference*, 1115–1119.

Pedrycz, W. 2022. Federated FCM: Clustering under privacy requirements. *IEEE Transactions on Fuzzy Systems*, 30(8): 3384–3388.

Peng, M.; Wu, Y.; Zhang, Y.; Lu, Y.; Li, M.; and Cheung, Y.-m. 2025. Weighted density for the win: Accurate subspace density clustering. In *2025 International Conference on Acoustics, Speech and Signal Processing*, 1–5.

Qiao, D.; Ding, C.; and Fan, J. 2024. Federated spectral clustering via secure similarity reconstruction. In *2024 Advances in Neural Information Processing Systems*, volume 36, 58520–58555.

Ros, F.; Riad, R.; and Guillaume, S. 2023. PDBI: A partitioning Davies-Bouldin index for clustering evaluation. *Neurocomputing*, 528: 178–199.

Rousseeuw, P. J. 1987. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20: 53–65.

Schubert, E.; Sander, J.; Ester, M.; Kriegel, H. P.; and Xu, X. 2017. DBSCAN revisited, revisited: Why and how you should (still) use DBSCAN. *ACM Transactions on Database Systems*, 42(3): 1–21.

Stallmann, M.; and Wilbik, A. 2022. Towards federated clustering: A federated fuzzy c-means algorithm (FFCM). arXiv:2201.07316.

Thorndike, R. L. 1953. Who belongs in the family? *Psychometrika*, 18(4): 267–276.

Wang, B.; Yang, Y.; Wu, J.; Qi, G.-j.; and Lei, Z. 2023. Self-similarity driven scale-invariant learning for weakly supervised person search. In *2023 International Conference on Computer Vision*, 1813–1822.

Wei, K.; Li, J.; Ding, M.; Ma, C.; Yang, H. H.; Farokhi, F.; Jin, S.; Quek, T. Q.; and Poor, H. V. 2020. Federated learning with differential privacy: Algorithms and performance analysis. *IEEE Transactions on Information Forensics and Security*, 15: 3454–3469.

Wu, J.; Yang, Y.; Liu, H.; Liao, S.; Lei, Z.; and Li, S. Z. 2019. Unsupervised graph association for person re-identification. In *2019 International Conference on Computer Vision*, 8321–8330.

Xie, S.; Wu, Y.; and Liao, e. a., Kewen. 2023. Fed-SC: One-shot federated subspace clustering over high-dimensional data. In *2023 International Conference on Data Engineering*, 2905–2918.

Zhang, C.; Xie, Y.; Bai, H.; Yu, B.; Li, W.; and Gao, Y. 2021. A survey on federated learning. *Knowledge-Based Systems*, 216: 106775.

Zhang, Y.; Cheung, Y.-m.; and Tan, K. C. 2020. A unified entropy-based distance metric for ordinal-and-nominal-attribute data clustering. *IEEE Transactions on Neural Networks and Learning Systems*, 31(1): 39–52.

Zhang, Y.; Zou, R.; Zhang, Y.; Zhang, Y.; Cheung, Y.-m.; and Li, K. 2025. Adaptive micro partition and hierarchical merging for accurate mixed data clustering. *Complex & Intelligent Systems*, 11: 1–14.

Zhou, W.; Li, P.; Han, Z.; Lu, X.; Li, J.; Ren, Z.; and Liu, Z. 2023. Privacy-preserving federated learning via disentanglement. In *2023 International Conference on Information and Knowledge Management*, 3606–3615.

Zhou, X.; and Wang, X. 2022. Memory and communication efficient federated kernel k-means. *IEEE Transactions on Neural Networks and Learning Systems*, 7114–7125.

Zou, R.; Zhang, Y.; Zhang, Y.; Lu, Y.; Li, M.; and Cheung, Y.-m. 2024. Federated clustering with unknown number of clusters. In *2024 International Conference on Data-driven Optimization of Complex Systems*, 671–677.

QGRL: Quaternion Graph Representation Learning for Heterogeneous Feature Data Clustering

Junyang Chen*
[redacted]

ABSTRACT

Clustering is one of the most commonly used techniques for unsupervised data analysis. As real data sets are usually composed of numerical and categorical features that are heterogeneous in nature, the heterogeneity in the distance metric and feature coupling prevents deep representation learning from achieving satisfactory clustering accuracy. Currently, supervised Quaternion Representation Learning (QRL) has achieved remarkable success in efficiently learning informative representations of coupled features from multiple views derived endogenously from the original data. To inherit the advantages of QRL for unsupervised heterogeneous feature representation learning, we propose a deep QRL model that works in an encoder-decoder manner. To ensure that the implicit couplings of heterogeneous feature data can be well characterized by representation learning, a hierarchical coupling encoding strategy is designed to convert the data set into an attributed graph to be the input of QRL. We also integrate the clustering objective into the model training to facilitate a joint optimization of the representation and clustering. Extensive experimental evaluations illustrate the superiority of the proposed Quaternion Graph Representation Learning (QGRL) method in terms of clustering accuracy and robustness to various data sets composed of arbitrary combinations of numerical and categorical features. The source code is opened at [redacted]

CCS CONCEPTS

• **Computing methodologies** → **Neural networks**; **Learning latent representations**; **Mixture modeling**; **Cluster analysis**.

KEYWORDS

Heterogeneous features, Graph neural network, Quaternion representation learning, Spectral clustering

ACM Reference Format:
Junyang Chen, [redacted]

*Co-first author.
†Corresponding author.

KDD '24, August 25–29, 2024, Barcelona, Spain

ACM ISBN 979-8-4007-0490-1/24/08
[redacted]

1 INTRODUCTION

Clustering is one of the most fundamental techniques in knowledge discovery and data mining tasks. It explores the potential distributions of data objects reflected by the feature values in an unsupervised way. With the explosive growth of various data, data sets composed of both numerical and categorical features are very common, which can be easily found in medical data analysis systems [24], citation relationship databases [6], to name a few. Exploring data object distributions jointly reflected by the numerical and categorical features is difficult due to the feature heterogeneity. That is, the heterogeneous numerical and categorical features are with quantitative and qualitative values, respectively, which describe object distributions in completely different ways [2, 39]. The loss of critical representation information across the heterogeneous features surely degrades the effectiveness of representation learning and the accuracy of downstream clustering.

Recent heterogeneous feature data clustering approaches attempt to develop similarity measures that take into account more data statistics and prior knowledge, including the occurrence frequency and semantic ordinal relationship of feature values, interdependence among features, etc. Compared with the conventional one-hot encoding [3] and Hamming distance [4] that simply consider the matching between two values for similarity representation, more advanced metrics [8, 20] that define distance structures for features by considering the occurrence frequency of intra-feature values have been proposed. Most recent works [39, 41, 42, 47] further exploit the statistical prior of inter-feature-coupling to achieve a more informative representation of categorical data. However,

these methods are based on the quantification of feature-level similarity without considering the relationship among data objects, and thus overlook the impact of object-level similarity to the clustering.

Thanks to the powerful ability of deep graph convolutional networks in revealing the relationship among graph nodes, deep graph representation learning-based clustering has attracted much attention and achieved competitive clustering performance [21, 35]. In the graph representation learning field, the dominant Graph Convolutional Network (GCN) [16] and its variants simultaneously embed the graph structure and feature values to obtain a more comprehensive data representation. Later, Graph Auto-Encoder (GAE) and its variants [15, 26, 35, 37] have also been specially developed for unsupervised representation learning of graph data. By adopting graph convolution layers as encoders, they considerably enhanced the performance of graph data clustering. Theoretically, the representation ability can be further improved by stacking more graph convolution layers. However, the embeddings tend to be homogeneous due to the common over-smoothing effect of stacking graph convolution layers. Accordingly, most graph representation learning models are restricted to a shallow graph convolutional network, preventing them from aggregating node relationship beyond local distributions. As a result, they fail to produce noise-robust embeddings, and will somewhat influence the clustering performance.

Therefore, in this work, a new graph representation learning method named Quaternion Graph Representation Learning (QGRL) has been proposed for heterogeneous feature data clustering. QGRL first constructs a graph on the heterogeneous feature data to capture the implicit value-level, feature-level, and object-level couplings, and then introduces a powerful quaternion representation learning mechanism [29] to circumvent the over-smoothing effect of the graph representation learning. More specifically, an adjacency matrix is derived from the data to form a graph structure, which is called Heterogeneous Data Graph (HDG). To ensure an informative graph construction, divergent statistical information of the data is encoded through the designed Hierarchical Coupling Encoding (HCE) strategy for the adjacency matrix computation. HDG acts to bridge the information pathway between heterogeneous feature data and the following representation learning. By generating four-view encodings from the constructed graph, the Hamilton product of Quaternion Representation Learning (QRL) can facilitate an efficient rotation of global features, bringing a higher degree of freedom to the representation learning. This compensates for the shallow graph convolutional network structure, and thus mitigates the over-smoothing of the learned node embeddings. By integrating the graph reconstruction and spectral clustering losses, the model is urged to learn clustering-friendly representations in the generated quaternion latent space. Extensive experiments on various heterogeneous feature data sets verify the superiority of the proposed method in terms of both representation learning and clustering. The main contributions are summarized into three-fold:

- A novel QRL framework is proposed for accurate and robust heterogeneous feature data clustering. It bridges the information pathway between heterogeneous features and representation learning using constructed graph, and also bridges the representation learning and clustering task by a joint learning scheme.
- To provide a high information fidelity basis for the representation learning, an encoding strategy is carefully designed to combine the statistical prior of data, including intra-feature probabilities, inter-feature dependencies, and inter-object distances computed by a metric unified on the heterogeneous features.
- This is the first attempt to introduce quaternion into unsupervised representation learning. Through our model design, an efficient decoupling for the representation of heterogeneous feature data has been formed, which is also of great reference value for applying quaternion to other unsupervised learning tasks.

2 RELATED WORK

This section overviews the related existing works in the fields of heterogeneous feature data clustering, graph representation learning, and quaternion representation learning.

2.1 Heterogeneous Feature Data Clustering

Existing approaches for heterogeneous feature data clustering can be roughly categorized into two types: 1) define a distance measure for categorical features for clustering, and 2) encode data into numerical data for clustering.

For the former type, a developing trend is to exploit the statistical information of data for more reasonable distance computation. Some studies [1, 14, 18] understand the similarity between two values from the perspective of probability. That is, if the probability of randomly picking two different values at the same time is higher, then they are considered more similar. Some other methods [20, 40, 41] compute the information entropy based on the probabilities, and judge the dissimilarity from the perspective of information theory. To extend the above ideas by taking into account the interdependence between features, conditional probability distributions (CPDs) corresponding to two possible values obtained from another feature are widely adopted to reflect the value-level distance by the existing methods [13, 23, 43]. By further exploiting the semantic order of categorical feature values [44], the method [39] defines distance metrics that are unified on numerical, nominal, and ordinal features for more universal clustering.

The latter type of method converts categorical values into numerical ones for clustering. As the conventional one-hot encoding overlooks the couplings within data, a more advanced encoding strategy that utilizes the adjacency matrix of data objects as the encoding has been proposed [31]. To adapt the encoding strategies to clustering, some recent advances [38, 42, 47] that make the encoding process learnable w.r.t. clustering objective have also been proposed. However, all the above-mentioned encoding strategies rely much on prior domain knowledge, which thus limits their efficacy. For the encoded data, conventional K-Means-type algorithms [12] or spectral clustering algorithms [22] can be directly applied to obtain the clustering results.

2.2 Deep Graph Representation Learning for Clustering

Inspired by the powerful feature extraction capability of convolutional neural networks [17], GCNs [16] have been proposed to

generalize the convolution operation to graph data, thereby integrating the graph structure and feature information for representation learning. Inheriting the powerful encoder-decoder representation learning backbone of AE [34] and VAE [7], Kipf *et al.* [15] proposed GAE and VGAE that project the inputs to a low-dimensional space and reconstruct the graph structure in a learnable manner to capture the key data features.

The variants of GAE-based methods [26, 28, 35, 37] further introduce different encoding enhancement mechanisms to improve the capability of embedding learning. The DAEGC [35] introduces the attention mechanism to integrate attribute information and graph architecture for more comprehensive representation learning. To further achieve joint clustering and representation learning, the work [37] relaxes the clustering objective and combines it into the training process of GAE. Later, to realize a more robust data representation learning, R-GAE [26] has been proposed to relieve the influence brought by the noise features, feature drifts, and feature randomness from a mathematics perspective. Although the above-mentioned GCN-based methods achieve considerable improvements in clustering, they still suffer from the intrinsic over-smoothing effect of graph convolution operation and have not taken into account the common issue of feature heterogeneity.

2.3 Quaternion Representation Learning

A quaternion is a hyper-complex number composed of four parts, and the Hamilton product of two quaternions can be viewed as their efficient rotation in the space spanned by the orthogonal imaginary axes. To leverage the efficient quaternion product in representation learning, some recent studies [9, 27, 29, 45, 48] have extended the feature encoding from real-value field to quaternion-value field for more sufficient feature coupling learning. The Quaternion Neural Networks (QNNs) [48] have demonstrated a great feature extraction ability in various supervised tasks, *e.g.*, few-shot segmentation [45], image classification [11], and speech recognition [30]. QCLNet [45] introduces quaternion representation learning to alleviate the computation burden brought by the high-dimensional correlated tensors, and also to explore the latent interactions between query and support images. The work [11] regards each RGB image as a quaternion, and embeds it with a learnable weight quaternion through the Hamilton product to achieve a more powerful representation learning. Benefiting from the orthogonal imaginary axes and the rotation nature of quaternion algebra, quaternion facilitates efficient feature coupling learning and thus is promising in enhancing representation learning of features with complex relationships.

3 PRELIMINARIES

This section introduces the definition of heterogeneous feature data and the problem setting of its clustering. Then the basic quaternion algebra is presented. Table 1 sorts out the frequently used notations and symbols in this paper.

A heterogeneous feature data set $\mathcal{S}$ is represented as a triplet $\mathcal{S} = < \mathcal{X}, \mathcal{A}, \mathcal{O} >$. The data object set $\mathcal{X} = \{\boldsymbol{x}_l | l = 1, 2, \ldots, n\}$ contains n objects, and each object $\boldsymbol{x}_l = [x_l^1, x_l^2, \ldots, x_l^d]^\top$ is represented by values from d features $\mathcal{A} = \{\boldsymbol{A}^r | r = 1, 2, \ldots, d\}$. For a heterogeneous feature data set, it is assumed that there are $d^{\{c\}}$ categorical and $d^{\{u\}}$ numerical features, and we have $\mathcal{A} = \mathcal{A}^{\{c\}} \cup \mathcal{A}^{\{u\}}$ with

Table 1: Frequently used notations and symbols.

Notations and Symbols	Explanations
Subscript *e.g.*, "i" of $\boldsymbol{x}_i$	Element index
Superscript *e.g.*, "r" of $\boldsymbol{A}^r$ and $\boldsymbol{O}^r$	Attribute index
Curly-bracketed superscript *e.g.*, "$\{c\}$" of $d^{\{c\}}$	Annotation
Calligraphic letters *e.g.*, $\mathcal{A}$, $\mathcal{O}$, and $\mathcal{X}$	Set
Uppercase, calligraphic font *e.g.*, $\mathcal{R}_{ij}^r$	Space
Uppercase, bold font, *e.g.*, $\mathbf{A}$ and $\mathbf{W}_i$	Matrix
Bold font, italic *e.g.*, $\boldsymbol{x}_l$ and $\boldsymbol{A}^r$	Vector

$d = d^{\{c\}} + d^{\{u\}}$, where $\mathcal{A}^{\{c\}}$ and $\mathcal{A}^{\{u\}}$ are the categorical and numerical feature set, respectively. Each feature can be written as an n-value vector $\boldsymbol{A}^r = [a_1^r, a_2^r, \cdots, a_n^r]$, and for a categorical feature $\boldsymbol{A}^r \in \mathcal{A}^{\{c\}}$, its n values are distributed on a limited number (*i.e.*, v^r for $\boldsymbol{A}^r$) of possible values, which can be written as a unique value set $\boldsymbol{O}^r = \{o_1^r, o_2^r, \ldots, o_{v^r}^r\}$ with $\boldsymbol{O}^r \in \boldsymbol{O}$. Our research goal is to perform QRL on the above-mentioned heterogeneous feature data sets to obtain satisfactory clustering performance. In this work, we focus on the common crisp partitional clustering task to divide a whole data set into a certain number of compact subsects containing closely distributed data objects.

The following introduces the quaternion operation rules in QRL. Quaternion Q is a type of hyper-complex number in the field $\mathbb{H}$, which can be represented as:

$$Q = r + x\mathbf{i} + y\mathbf{j} + z\mathbf{k}, \tag{1}$$

where r is for the real part and $x\mathbf{i}+y\boldsymbol{j}+z\mathbf{k}$ represents the imaginary parts. In $\mathbb{H}$, an orthogonal relationship holds for the imaginary parts, *i.e.*, $\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1$. Then we present the quaternion algebra involved in this paper: i) addition, ii) scale multiple, and iii) Hamilton product, in the following.

i) Addition: Given two quaternions Q_1 and Q_2, the addition operation adds up their corresponding parts by:

$$\begin{aligned} Q_1 + Q_2 = (r_1 + r_2) + (x_1 + x_2)\mathbf{i} \\ + (y_1 + y_2)\mathbf{j} + (z_1 + z_2)\mathbf{k}. \end{aligned} \tag{2}$$

ii) Scale Multiple: A quaternion Q can be scaled by a scalar λ as:

$$\lambda Q = \lambda r + \lambda x\mathbf{i} + \lambda y\mathbf{j} + \lambda z\mathbf{k}. \tag{3}$$

iii) Hamilton Product: The interaction between two quaternions Q_1 and Q_2 is specified by the Hamilton product known as the quaternion transformation. More specifically, Q_1 can be transformed by rotating it based on the quaternion Q_2 by:

$$\begin{aligned} Q_1 \otimes Q_2 = (r_1 r_2 - x_1 x_2 - y_1 y_2 - z_1 z_2) \\ + (r_1 x_2 + x_1 r_2 + y_1 z_2 - z_1 y_2)\mathbf{i} \\ + (r_1 y_2 - x_1 z_2 + y_1 r_2 + z_1 x_2)\mathbf{j} \\ + (r_1 z_2 + x_1 y_2 - y_1 x_2 + z_1 r_2)\mathbf{k}. \end{aligned} \tag{4}$$

Such an operation can form efficient interaction among feature components in the quaternion field, and can thus be utilized to facilitate a higher degree of freedom for learning models in representing complex data couplings.

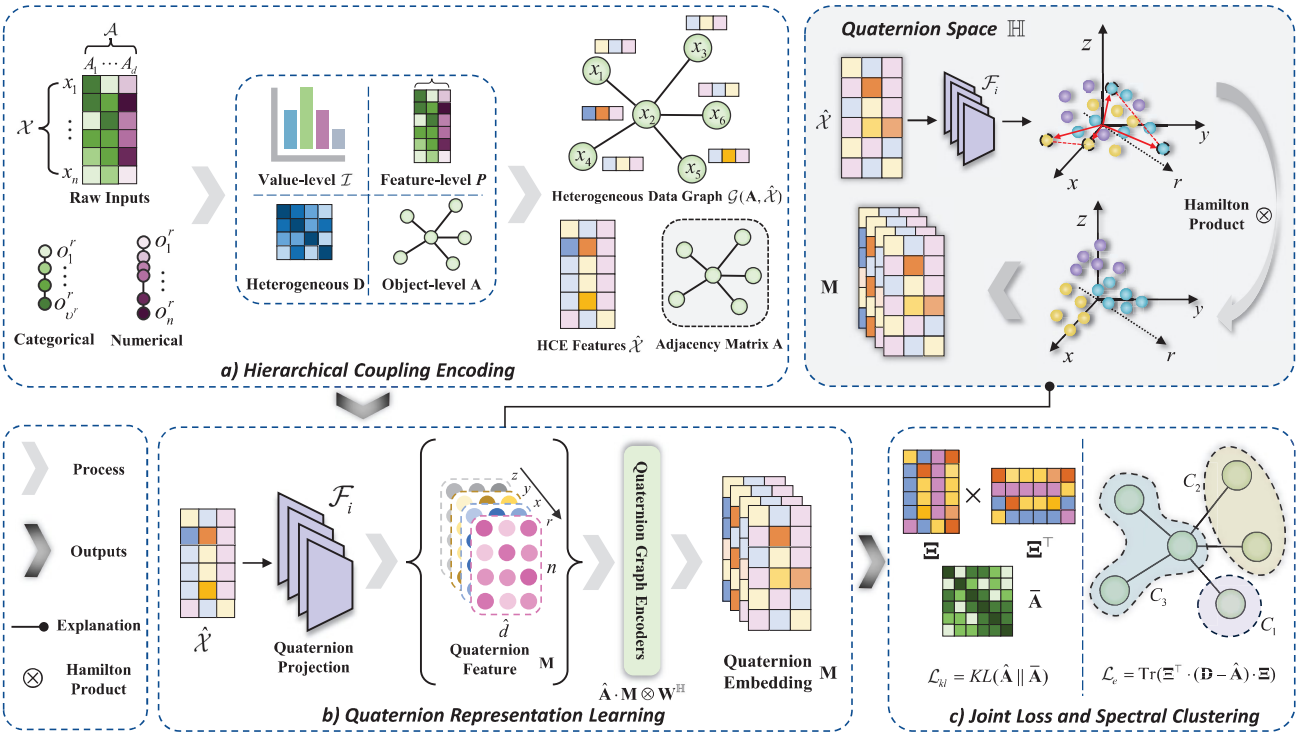


Figure 1: Overview of the proposed QGRL. Heterogeneous data is first encoded into a more informative attributed graph $\mathcal{G} = \{\mathrm{A}, \hat{\mathcal{X}}\}$ called Heterogeneous Data Graph (HDG) through the proposed Hierarchical Coupling Encoding (HCE) strategy. Then a multi-view projection is performed to convert the attributes of HDG into the quaternion space for quaternion representation learning. The obtained quaternion embedding Ξ is reconstructed to form adjacency matrix $\bar{\mathrm{A}}$ as the decoding operation. Finally, Ξ output by the trained QGRL model is utilized for spectral clustering.

4 PROPOSED METHOD

In this section, we first introduce Hierarchical Coupling Encoding (HCE) strategies to comprehensively encode the complex relation of heterogeneous feature data, and then present the proposed Quaternion Graph Representation Learning (QGRL) for clustering. The overall pipeline of QGRL is shown in Figure 1.

4.1 HCE: Hierarchical Coupling Encoding

There are four types of coupling of heterogeneous feature data: 1) value-level coupling, *i.e.*, the coupling among possible values within a categorical feature, 2) feature-level coupling, *i.e.*, the coupling between interdependent features, 3) heterogeneous coupling, *i.e.*, the coupling between different types of features, and 4) object-level coupling, *i.e.*, the coupling among data objects reflected by their similarities. By properly encoding these couplings, a coupling learning can be facilitated with a deep representation framework. In this subsection, we introduce the proposed coupling encoding strategies in the above-mentioned hierarchies.

4.1.1 Value-level coupling. The occurrence probabilities of possible values $O^r = \{o_1^r, o_2^r, \ldots, o_{v^r}^r\}$ of a categorical feature $\boldsymbol{A}^r$ can be viewed as a series of probabilities:

$$\mathcal{I}^r = \{P_i^r | i = 1, \ldots, v^r\} \tag{5}$$

where P_i^r is the occurrence probability of possible value o_i^r in $\boldsymbol{A}^r$:

$$P_i^r = \frac{\delta(\{\boldsymbol{A}^r\}_1^n = o_i^r)}{\delta(\{\boldsymbol{A}^r\}_1^n \neq \text{Null})}. \tag{6}$$

Here, $\delta(\{\boldsymbol{A}^r\}_1^n = o_i^r)$ counts the occurrence frequency of o_i^r in feature value set $\{\boldsymbol{A}^r\}_1^n$, and $\delta(\{\boldsymbol{A}^r\}_1^n \neq \text{Null})$ counts the number of non-empty values in $\{\boldsymbol{A}^r\}_1^n$ which is usually equal to n. Note that we use the uppercase P_i^r here to distinguish the occurrence probability of a value from the conditional probability that will be presented in the following. Since the probabilities satisfy:

$$\sum_{i=1}^{v^r} P_i^r = 1,$$

encoding the feature values by the occurrence probabilities of the corresponding possible values can surely capture the value-level couplings within each feature.

4.1.2 Feature-level coupling. The original features of a real data set are usually interdependent to a certain extent. To represent such inter-feature relations, we also define the Conditional Probability Distribution (CPD) of a feature $\boldsymbol{A}^m$ given a possible value o_i^r from another feature $\boldsymbol{A}^r$ as a v^m-dimensional vector:

$$\boldsymbol{P}_i^{m|r} = [p(o_1^m | o_i^r), p(o_2^m | o_i^r), \ldots, p(o_{v^m}^m | o_i^r)]^\top, \tag{7}$$

where the conditional probability $p(o_j^m|o_i^r)$ is computed by:

$$p(o_j^m|o_i^r) = \frac{\sigma(\mathcal{X}_j^m \cap \mathcal{X}_i^r)}{\sigma(\mathcal{X}_i^r)}. \tag{8}$$

Here, $\mathcal{X}_i^r = \{\boldsymbol{x}_l | x_l^r = o_i^r, l = 1, 2, \cdots, n\}$ is a subset of $\mathcal{X}$, which contains all the data objects with their r-th values equal to o_i^r. The function $\sigma(\cdot)$ counts the cardinality of a set. With $\boldsymbol{P}_i^{m|r}$, we can encode a value o_i^r according to different features $\boldsymbol{A}^m \in \mathcal{A}^{\{c\}}$ to preserve the inter-feature dependence.

4.1.3 Heterogeneous coupling. The above feature-level coupling encoding treats categorical features uniformly according to the CPDs. However, for heterogeneous feature data, heterogeneity of the distance structures of numerical and categorical features have not been represented yet. To effectively connect heterogeneous features while preserving their intrinsic distance structure, we propose to project categorical feature values onto a series of one-dimensional spaces, and then encode the categorical values according to their locations after the projection.

Remark 1. ***Connection of heterogeneous features.*** *The reason to project the categorical values onto one-dimensional spaces is to unify the categorical and numerical features by letting them reflect distances in the same manner. In this way, the basis of appropriately representing the coupling of the heterogeneous features is formed.*

The projection is performed according to a commonly used categorical feature distance metric based on CPDs defined in Section 4.1.2, where the distance between two possible values o_i^r and o_j^r of a feature $\boldsymbol{A}^r$ is computed according to each of the categorical features $\mathcal{A}^{\{c\}}$, which can be written as:

$$d(o_i^r, o_j^r) = \sum_{\boldsymbol{A}^m \in \mathcal{A}^{\{c\}}} \left\| \boldsymbol{P}_i^{m|r} - \boldsymbol{P}_j^{m|r} \right\|. \tag{9}$$

With this distance definition, we can project all the v^r possible values of feature $\boldsymbol{A}^r$ onto each of the $v^r(v^r-1)/2$ one-dimensional spaces spanned by the corresponding pairs of possible values. That is, given a one-dimensional space $\mathcal{R}_{ij}^r$ spanned by two possible values o_i^r and o_j^r, projection point of a value o_t^r can be determined by computing its distance to o_i^r (or to o_j^r) in the space $\mathcal{R}_{ij}^r$ by:

$$\phi(o_t^r, o_i^r; \mathcal{R}_{ij}^r) = \frac{|d(o_t^r, o_i^r)^2 - d(o_t^r, o_j^r)^2 + d(o_i^r, o_j^r)^2|}{2 \cdot d(o_i^r, o_j^r)} \tag{10}$$

following the Pythagorean theorem. For more projection details, readers can refer to [42]. After projecting all the v^r possible values, distance between each pair of the possible values in $\mathcal{R}_{ij}^r$ is obtained, and we organize the distances as a symmetric matrix $\mathbf{D}_{ij}^r \in \mathbb{R}^{v^r \times v^r}$ with its (t,l)-th entry $\mathbf{D}_{ij}^r(t,l)$ indicating the distance between o_t^r and o_l^r in the projection space $\mathcal{R}_{ij}^r$.

Remark 2. ***Comprehensiveness of projection.*** *Each categorical feature $\boldsymbol{A}^r$ is represented as a series of $v^r(v^r-1)/2$ one-dimensional distance structures from different endogenous views formed by different pairs of possible values. Combined with Remark 1, we know that the projection informatively preserves the intrinsic relationship among possible values, while the form of one-dimensional embedding ensures a homogeneous connection with numerical features.*

4.1.4 Encoding of couplings. With the above-mentioned three types of coupling encoding, all the categorical features $\mathcal{A}^{\{c\}}$ are represented to higher dimensions. Specifically, given the l-th value of a categorical feature $\boldsymbol{A}^r$ satisfying $a_l^r = o_i^r$, a_l^r will be encoded into a vector by concatenating its three types of coupling encoding as:

$$\hat{\boldsymbol{a}}_l^r = [P_i^r, \underbrace{\boldsymbol{P}_i^{1|r}, \boldsymbol{P}_i^{2|r}, \ldots,}_{d^{\{c\}}} \underbrace{\mathbf{D}_{11}^r(i,\cdot), \mathbf{D}_{12}^r(i,\cdot), \ldots]}_{v^r(v^r-1)/2} \tag{11}$$

where P_i^r indicates the value-level coupling, $\boldsymbol{P}_i^{1|r}, \boldsymbol{P}_i^{2|r}, \ldots$ is the feature-level coupling, and $\mathbf{D}_{11}^r(i,\cdot), \mathbf{D}_{12}^r(i,\cdot), \ldots$ stands for the heterogeneous coupling. Notation $\mathbf{D}_{ij}^r(t,\cdot)$ indicates the t-th row of matrix $\mathbf{D}_{ij}^r$ defined in Section 4.1.3. By encoding each feature value in $\mathcal{A}^{\{c\}}$, the encoded categorical feature set can be denoted as $\hat{\mathcal{A}}^{\{c\}}$, and the whole feature set is updated to $\hat{\mathcal{A}} = \hat{\mathcal{A}}^{\{c\}} \cup \mathcal{A}^{\{u\}}$. Accordingly, we denote the object set corresponding to $\hat{\mathcal{A}}$ as $\hat{\mathcal{X}}$.

We have presented the encoding of the three types of couplings so far, *i.e.*, value-level, feature-level, and heterogeneous couplings. The last object-level coupling encoding is performed by constructing a fully connected graph on the data objects, which will be separately discussed in the next subsection.

4.2 HDG: Heterogeneous Data Graph Construction

To construct an HDG on the data objects, we first define the object-level distance between two objects $\boldsymbol{x}_a$ and $\boldsymbol{x}_b$ as L2 norm by:

$$\Psi(\boldsymbol{x}_a, \boldsymbol{x}_b) = \left\| [\Phi^1(x_a^1, x_b^1), \Phi^2(x_a^2, x_b^2), \ldots, \Phi^d(x_a^d, x_b^d)]^\top \right\|_2, \tag{12}$$

where $\Phi^r(x_a^r, x_b^r)$ is the distance reflected by the r-th feature. To achieve a more reasonable distance measurement on heterogeneous feature data, we adopt graph-based unified dissimilarity proposed in [39] to compute $\Phi^r(x_a^r, x_b^r)$. Suppose $x_a^r = o_i^r$ and $x_b^r = o_j^r$ for $\boldsymbol{A}^r \in \mathcal{A}^{\{c\}}$, then the distance $\Phi^r(x_a^r, x_b^r)$ can be written as:

$$\Phi^r(x_a^r, x_b^r) = \begin{cases} \sum_{m=1}^{d} \phi^{r|m}(o_i^r, o_j^r) \cdot \omega^{r|m}, & \text{if } \boldsymbol{A}^{\mathrm{r}} \in \mathcal{A}^{\{\mathrm{c}\}} \\ |x_a^r - x_b^r|, & \text{if } \boldsymbol{A}^{\mathrm{r}} \in \mathcal{A}^{\{\mathrm{u}\}} \end{cases} \tag{13}$$

where

$$\phi^{r|m}(o_i^r, o_j^r) = \frac{\| |\boldsymbol{P}_i^{m|r} - \boldsymbol{P}_j^{m|r}| \|_1}{2} \tag{14}$$

is the dissimilarity between o_i^r and o_j^r reflected by $\boldsymbol{A}^m$, and $\omega^{r|m}$ weights the importance of $\boldsymbol{A}^m$ in indicating the distances of values from $\boldsymbol{A}^r$, which can be specified by users, and can also be computed according to the interdependence between $\boldsymbol{A}^m$ and $\boldsymbol{A}^r$. Following [39], we discretize each of the numerical features into five equal-length intervals, and then treat the discretized features as categorical features to complete the computation of Eq. (14), as $\boldsymbol{A}^m$ could be a numerical feature. Although the value-level distances in the two cases are with different formats in Eq. (13), they are unified from the perspective of transformation cost computed by the Earth Mover's Distance (EMD). Due to space limitation, we refer the readers to [39] for more details about the unification and weights computation.

Remark 3. ***Rationality of HDG construction.*** *In Section 4.1.4, an informative coupling encoding $\hat{\mathcal{X}}$ has been obtained, which can be utilized to compute object-level distances based on Euclidean distance. The reasons why we instead adopt the graph-based unified dissimilarity upon the original $\mathcal{X}$ for HDG construction are two-fold: 1) it treats the heterogeneous numerical and categorical features in a unified way to avoid information loss, and 2) categorical features have been greatly expanded in $\hat{\mathcal{X}}$, directly calculating distances upon it will lead to overemphasis on the categorical features.*

After computing the distances between each pair of objects by Eq. (12), an adjacency matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ is obtained with its (i, j)-th entry equals to $\Psi(\boldsymbol{x}_i, \boldsymbol{x}_j)$. So far we have completed the encoding for all four types of couplings, which can be compactly represented by the constructed HDG $\mathcal{G} = \{\mathbf{A}, \hat{\mathcal{X}}\}$. Then $\mathcal{G}$ is treated as the input of the proposed representation learning model, which will be detailed in the following subsections.

4.3 QGRL: Quaternion Graph Representation Learning

To convert the attributes $\hat{\mathcal{X}}$ of the constructed attributed graph $\mathcal{G}$ into quaternion space, a learnable quaternion projection module is designed to project $\hat{\mathcal{X}}$ into quaternion-value space via:

$$\mathcal{F}_i(\hat{\mathcal{X}}; \mathbf{W}_i^{\mathbb{P}}, \mathbf{B}_i^{\mathbb{P}}) = \mathbf{W}_i^{\mathbb{P}}\hat{\mathcal{X}} + \mathbf{B}_i^{\mathbb{P}} \tag{15}$$

where $\mathcal{F}_i(\cdot)$ is the linear projection function w.r.t. different quaternion components, *i.e.*, $i \in \{r, x, y, z\}$. Here we use the superscript $\mathbb{P}$ to distinguish the learnable parameters of the projection phase from the following parameters of the quaternion encoding phase indicated by $\mathbb{H}$. Instead of compressing the features in the real-value space, the quaternion projection aims to informatively convert features into the four-view quaternion-value space $\mathbb{H}$ to facilitate representation learning with a higher degree of freedom. After the projection, the encoded quaternion feature is formulated as:

$$\mathbf{M} = \mathbf{M}_r + \mathbf{M}_x\mathbf{i} + \mathbf{M}_y\mathbf{j} + \mathbf{M}_z\mathbf{k}, \tag{16}$$

where $\mathbf{M} \in \mathbb{H}^{n\times(4\times\hat{d})}$ denotes the quaternion feature matrix.

To learn the interdependencies between different quaternion components, we propose to use a *quaternion graph representation encoder* to capture the relations between quaternion embeddings:

$$\mathcal{H}_h(\hat{\mathbf{A}}, \mathbf{M}_h; \mathbf{W}_h^{\mathbb{H}}) = \varphi(\hat{\mathbf{A}} \cdot \mathbf{M}_h \otimes \mathbf{W}_h^{\mathbb{H}}), \tag{17}$$

where $\varphi(\cdot)$ is the ReLU function, $\hat{\mathbf{A}}$ denotes the normalized Laplacian matrix of $\mathbf{A}$, h indexes to the number of encoder layers, and the symbol $\otimes$ indicates the Hamilton product, which can be defined as:

$$\mathbf{M} \otimes \mathbf{W}^{\mathbb{H}} = \begin{bmatrix} \mathbf{M}_r \\ \mathbf{M}_x \\ \mathbf{M}_y \\ \mathbf{M}_z \end{bmatrix}^{\top} \begin{bmatrix} \mathbf{W}_r^{\mathbb{H}} & -\mathbf{W}_x^{\mathbb{H}} & -\mathbf{W}_y^{\mathbb{H}} & -\mathbf{W}_z^{\mathbb{H}} \\ \mathbf{W}_x^{\mathbb{H}} & \mathbf{W}_r^{\mathbb{H}} & -\mathbf{W}_z^{\mathbb{H}} & \mathbf{W}_y^{\mathbb{H}} \\ \mathbf{W}_y^{\mathbb{H}} & \mathbf{W}_z^{\mathbb{H}} & \mathbf{W}_r^{\mathbb{H}} & -\mathbf{W}_x^{\mathbb{H}} \\ \mathbf{W}_z^{\mathbb{H}} & -\mathbf{W}_y^{\mathbb{H}} & \mathbf{W}_x^{\mathbb{H}} & \mathbf{W}_r^{\mathbb{H}} \end{bmatrix}^{\top}, \tag{18}$$

where $\mathbf{W}^{\mathbb{H}}$ denotes the learnable parameters. For simplicity, we omit the subscript h of $\mathbf{M}$ and $\mathbf{W}^{\mathbb{H}}$ in the Eq. (18).

After propagating the feature representations with relations between quaternion components, the quaternion feature embeddings are further aggregated into a single feature matrix, which is utilized to calculate the loss of Graph reconstruction and clustering in the next stage. In practice, the quaternion feature embedding aggregation process can be formulated as:

$$\Xi = \mathrm{Re}(\mathbf{M}_L) \circledast \mathrm{Im}(\mathbf{M}_L) \tag{19}$$

where $\mathrm{Re}(\cdot)$ and $\mathrm{Im}(\cdot)$ are the real and imaginary parts of $\mathbf{M}_L$, respectively. $\mathbf{M}_L$ is the output of the last encoder layer $\mathcal{H}_L(\cdot)$, and the symbol $\circledast$ indicates quaternion fusion operation, which takes an average of the four quaternion embedding components to form a compact embedding for the downstream graph construction and clustering. Then, we reconstruct the adjacency matrix by:

$$\bar{\mathbf{A}} = \Xi \cdot \Xi^{\top} \tag{20}$$

where $\bar{\mathbf{A}}$ denotes the reconstructed matrix, which will be directly utilized to compute the training loss.

4.4 Joint Optimization for Graph Reconstruction and Clustering

Inspired by [19], we propose to jointly optimize unsupervised heterogeneous feature representation learning and spectral clustering by integrating the Kullback-Leibler (KL) divergence and relaxed spectral clustering objective as the loss function. Intuitively, the KL loss encourages the model to learn feature embeddings by recovering the original graph connectivity. The clustering loss aims to learn discriminative embeddings and facilitate clustering analysis by preserving the similarity between close objects in the embedding space. Concretely, the overall loss function is defined as:

$$\mathcal{L} = \mathcal{L}_{kl} + \alpha\mathcal{L}_{reg} + \beta\mathcal{L}_e, \tag{21}$$

where α and β are the hyper-parameters indicating loss weights. $\mathcal{L}_{reg}$ is the regularization term. In our implementation, L1 regularization is adopted to penalize the complexity of the model. The KL loss $\mathcal{L}_{kl}$ is expressed as:

$$\mathcal{L}_{kl} = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\hat{\mathbf{A}}_{ij}\log\frac{1}{\bar{\mathbf{A}}_{ij}}, \tag{22}$$

where $\hat{\mathbf{A}}$ and $\bar{\mathbf{A}}$ denote the normalized Laplacian adjacency matrix and the reconstructed adjacency matrix, respectively.

Similar to Graph Laplacian Eigenmaps [5], we introduce the loss term $\mathcal{L}_e$ into the loss function to preserve the graph property and penalize the quaternion embedding with higher similarity but lower connectivity in the graph. Formally, $\mathcal{L}_e$ is defined as:

$$\mathcal{L}_e = \mathtt{tr}(\Xi^{\top}\left(\mathbf{D} - \bar{\mathbf{A}}\right)\Xi), \tag{23}$$

where $\mathbf{D}$ denotes the degree matrix, $\bar{\mathbf{A}}$ is the reconstructed adjacency matrix, Ξ contains the learned quaternion embeddings, and $\mathtt{tr}(\cdot)$ computes the trace of matrix.

The KL loss $\mathcal{L}_{kl}$ encourages the consensus fusion of currently learned embedding and the original graph structure information in the process of reconstruction, while the eigenmap loss $\mathcal{L}_e$ makes the model prefer sparse graph structure with higher feature similarity of the connected nodes, which is consistent with the spectral clustering objective, and can thus be treated as a relaxed spectral clustering objective. In summary, they complement each other in terms of informativeness and clustering friendliness of the learned embeddings. Finally, the ultimate embeddings Ξ output by the trained QGRL model is treated as the input of spectral clustering to obtain a certain number of clusters [22].

Table 2: Statistics of ten data sets, where n, d_c, d_u, and k^* indicate the number of objects, categorical features, numerical features, and clusters, respectively.

No.	Data set	Abbrev.	n	d_c	d_u	k^*
1	Heart Failure	HF	299	5	7	2
2	Breast Cancer	BC	286	5	4	2
3	Autism-Adolescent	AA	104	7	2	2
4	Mammographic	MM	961	4	1	2
5	Zoo	ZO	101	16	0	7
6	Tic-Tac-Toe	TTT	958	9	0	2
7	Glass Identification	GI	214	0	9	6
8	Yeast	YE	1484	0	8	10
9	Iris	II	150	0	4	3
10	Wine	WI	178	0	13	3

5 EXPERIMENTS

In this section, we first outline the experimental settings, and then demonstrate experimental results with discussions.

5.1 Experimental Settings

5.1.1 Experiments Summary. In this section, we conduct four sets of experiments to comprehensively evaluate our proposed clustering framework. The experimental designs are as follows.

- Clustering Performance Evaluation: To demonstrate the superior performance of our proposed clustering framework, we compare the clustering performance of ten methods that include traditional, representative, and advanced counterparts on the heterogeneous feature data sets.
- Significance Test: To illustrate the superiority of our method, we perform significance tests in comparison with the state-of-the-art methods.
- Ablation Study: To demonstrate the effectiveness of the proposed hierarchical coupling encoding (HCE) and quaternion graph representation learning (QGRL) for heterogeneous data clustering, we conduct ablation studies by incrementally adding the proposed modules to a baseline model.
- Visualization Analysis: To intuitively demonstrate the effectiveness of QGRL, t-SNE is utilized to compare the cluster effect of embeddings generated by different models.

5.1.2 Data sets. Ten widely used public data sets are employed in the experiments. Most data sets are with a mixture of categorical and numerical features, and we also adopt pure categorical and pure numerical data sets for more comprehensive evaluation. All the data sets are from the UCI machine learning repository [25], and their detailed statistics are provided in Table 2.

5.1.3 Compared Methods and Implementation Details. Our method was compared with ten clustering counterparts, which can be broadly categorized into: 1) Traditional methods: K-Means [12] and Spectral clustering (SC) [22], 2) Representation learning methods: GAE [15] and VGAE [15], 3) State-of-the-art graph representation learning methods: ARGAE/ARVGAE [28], CCGC [36], DFCN [32], DAEGC [35], and the EGAE [37]. Some GCN methods require graph structure as input. For these methods, we apply one-hot encoding to the categorical features, thereby expanding the original features. We then construct the adjacency matrix based on the Euclidean distance between the expanded features. Our QGRL use the HDG $\mathcal{G} = \{\mathbf{A}, \hat{X}\}$ constructed by our HCE as input. The quaternion graph encoder is implemented by stacking two graph convolutional layers. For hyper-parameter settings, the length of latent vectors produced by the two layers is set at 1024 and 512, respectively. In practice, QGRL is pre-trained with only KL divergence and regularization loss and then trained in 50 epochs to obtain the representation and clustering results. After the training of each epoch, we perform matrix decomposition on the reconstructed adjacency matrix, and then extract the final clustering results from the matrix. To mitigate the effects of randomness, the mean and standard deviation are computed after running each method 10 times.

The settings of the compared methods are briefly described below. For GAE and VGA, we perform 200 epochs of unsupervised training, then use K-Means to cluster the generated embedding. The ARGAE, ARVGAE, CCGC, DFCN, DAEGC, and EGAE are implemented following their source code and original settings. All the experiments are implemented in PyTorch 1.8.0 and performed on a machine with an NVIDIA A5000 GPU, 64GB RAM.

5.1.4 Evaluation Metrics. In this paper, the clustering performance of the methods is evaluated by using three metrics [46], *i.e.*, Accuracy (ACC), Normalized Mutual Information (NMI), and Adjusted Rand Index (ARI). For the significance test discussed in Section 5.3, Wilcoxon signed-ranks test[10] is adopted for pairwise comparison of the counterparts.

5.2 Clustering Performance Evaluation

Based on the results in Table 3, our observations are as follows.

1) The proposed QGRL outperforms nearly all listed methods in three metrics across ten data sets, with an average ranking of 1.1, which illustrates its effectiveness in clustering in general.

2) On the most challenging heterogeneous feature data sets, *i.e.*, HF, BC, AA, and MM data sets, QGRL obviously outperforms its counterparts. Particularly, on the MM data set, QGRL achieves great improvements in comparison with the other methods. This indicates that the HCE encoding can effectively represent complex relation information of the heterogeneous features.

3) Although QGRL focuses on representation learning of heterogeneous feature data, its clustering performance is still extremely competitive on pure categorical data sets, *i.e.*, ZO and TTT. This is because the proposed HCE comprehensively performs four types of hierarchical encoding. For categorical data, encoding of the value-level, feature-level, and object-level couplings still acts to informatively represent the implicit complex relation within the categorical data, thus boosting the clustering performance.

4) QGRL outperforms the other counterparts on the TTT data set. TTT contains samples indicating the chessboard status in tic-tac-toe game, and there are two meaningful clusters in this data set, *i.e.*, winning and losing. QGRL performs well because its HCE considers the correlation between categorical features (*i.e.* Section 4.1.2). Since each placement position on the chessboard is a feature in the data set, the encoding of feature couplings is equivalent to considering the correlation between chessboard placement positions, which is crucial to distinguishing between winning and losing. Moreover,

Table 3: Average clustering performance on ten heterogeneous benchmark data sets evaluated by ACC, NMI, and ARI (Mean±Std), with best results bolded and Average Performance Ranks (AR) noted.

Data set	Metric	K-Means	SC	GAE	VGAE	ARGAE	ARVGAE	CCGC	DFCN	DAEGC	EGAE	QGRL (ours)
HF	ACC	0.6221±0.00	0.5485±0.00	0.6361±0.01	0.5438±0.03	0.6408±0.01	0.6890±0.00	0.6355±0.00	0.5742±0.02	0.6789±0.01	0.6900±0.02	**0.7137±0.03**
	NMI	0.0025±0.00	0.0020±0.00	0.0074±0.01	0.0028±0.01	0.0035±0.01	0.0267±0.00	0.0075±0.00	0.0002±0.00	0.0101±0.01	0.0296±0.03	**0.1092±0.03**
	ARI	0.0175±0.00	0.0042±0.00	0.0336±0.00	-0.0004±0.01	0.0220±0.00	0.0331±0.00	0.0337±0.00	-0.0004±0.00	0.0085±0.01	0.0263±3.36	**0.1663±0.04**
BC	ACC	0.5140±0.00	0.5210±0.00	0.7098±0.00	0.5430±0.05	**0.7227±0.01**	0.7154±0.01	0.7098±0.01	0.5531±0.05	0.6993±0.00	0.6647±0.06	0.7203±0.00
	NMI	0.0021±0.00	0.0014±0.00	0.0684±0.00	0.0048±0.01	0.0826±0.01	0.0756±0.01	0.0753±0.01	0.0166±0.02	0.0039±0.00	0.0266±0.03	**0.0866±0.01**
	ARI	-0.0025±0.00	-0.0012±0.00	0.1468±0.00	0.0020±0.02	0.1688±0.01	0.1572±0.01	0.1524±0.01	0.0164±0.04	-0.0040±0.00	0.0362±6.01	**0.1700±0.00**
AA	ACC	0.5096±0.00	0.6058±0.00	0.5192±0.00	0.5481±0.02	0.5663±0.01	0.5394±0.01	0.5788±0.02	0.5144±0.02	0.6125±0.01	0.6192±0.01	**0.6356±0.01**
	NMI	0.0006±0.00	0.0274±0.00	0.0014±0.00	0.0102±0.01	0.0066±0.01	0.0002±0.00	0.0154±0.00	0.0006±0.00	0.0250±0.00	0.0271±0.01	**0.0471±0.01**
	ARI	-0.0090±0.00	0.0353±0.00	-0.0127±0.00	-0.0063±0.01	0.0041±0.01	-0.0087±0.01	0.0148±0.01	-0.0081±0.01	0.0080±0.69	0.0162±0.02	**0.0375±0.02**
MM	ACC	0.6855±0.00	0.5398±0.00	0.5337±0.00	0.5119±0.01	0.6306±0.05	0.5559±0.01	0.6831±0.02	0.6855±0.00	0.5306±0.05	0.6852±0.01	**0.8296±0.00**
	NMI	0.1102±0.00	0.0328±0.00	0.0076±0.00	0.0007±0.00	0.0577±0.04	0.0103±0.01	0.1009±0.01	0.1065±0.00	0.0105±0.03	0.1030±0.01	**0.3487±0.01**
	ARI	0.1367±0.00	0.0054±0.00	0.0035±0.00	-0.0003±0.01	0.0752±0.05	0.0117±0.01	0.1338±0.02	0.1367±0.00	0.0109±0.03	0.1362±0.01	**0.4340±0.01**
ZO	ACC	0.7238±0.07	0.6109±0.01	0.5960±0.01	0.3772±0.03	0.7624±0.00	0.7634±0.01	0.6842±0.03	0.7168±0.08	0.3647±0.03	0.6891±0.03	**0.8020±0.02**
	NMI	0.7761±0.05	0.7237±0.01	0.5479±0.02	0.2357±0.03	0.7299±0.00	0.7315±0.03	0.6340±0.02	0.7068±0.05	0.1534±0.07	0.7111±0.03	**0.8212±0.02**
	ARI	0.6755±0.11	0.5515±0.01	0.3811±0.01	0.0797±0.03	0.6416±0.00	0.6446±0.02	0.5015±0.04	0.5744±0.10	-0.0410±0.01	0.5687±0.05	**0.7733±0.03**
TTT	ACC	0.5783±0.00	0.5167±0.00	0.5992±0.00	0.5295±0.01	0.6069±0.03	0.5461±0.01	0.6033±0.02	0.5335±0.01	0.6277±0.06	0.6531±0.01	**0.9574±0.03**
	NMI	0.0074±0.00	0.0002±0.00	0.0110±0.00	0.0016±0.01	0.0151±0.01	0.0011±0.01	0.0126±0.01	0.0025±0.00	0.0014±0.00	0.0084±0.01	**0.7633±0.10**
	ARI	0.0195±0.00	-0.0001±0.00	0.0308±0.00	0.0027±0.01	0.0388±0.02	0.0045±0.00	0.0343±0.01	0.0040±0.01	-0.0014±0.00	0.0092±0.02	**0.8381±0.09**
GI	ACC	0.5421±0.00	0.4720±0.00	0.4346±0.00	0.2785±0.02	0.4650±0.01	0.3902±0.01	0.4477±0.01	0.4556±0.01	0.3617±0.03	0.5505±0.01	**0.5509±0.03**
	NMI	0.4207±0.01	0.2914±0.00	0.3016±0.00	0.0405±0.01	0.3168±0.01	0.1949±0.01	0.3003±0.02	0.3148±0.02	0.0615±0.06	0.3763±0.01	**0.5004±0.06**
	ARI	0.2655±0.01	0.1594±0.00	0.1993±0.00	0.0029±0.01	0.2180±0.01	0.1289±0.00	0.1728±0.03	0.1770±0.01	0.0132±0.01	0.2587±0.01	**0.3476±0.08**
YE	ACC	0.3620±0.01	0.30.93±0.00	0.2323±0.01	0.1495±0.01	0.2573±0.01	0.2849±0.00	0.2240±0.03	0.2174±0.03	0.3127±0.00	0.3638±0.01	**0.4422±0.01**
	NMI	**0.2642±0.01**	0.0082±0.00	0.0874±0.00	0.0144±0.01	0.0914±0.01	0.0895±0.00	0.0758±0.02	0.1614±0.01	0.0110±0.00	0.2575±0.01	0.2182±0.01
	ARI	0.1395±0.01	-0.0025±0.00	0.0416±0.00	-0.0005±0.00	0.0474±0.00	0.0515±0.00	0.0326±0.02	0.0578±0.01	0.0005±0.00	0.1395±0.01	**0.1431±0.01**
II	ACC	0.8933±0.00	0.9067±0.00	0.8000±0.00	0.3947±0.03	0.8367±0.03	0.8827±0.04	0.9253±0.01	0.8373±0.01	0.3422±0.01	0.9167±0.01	**0.9627±0.02**
	NMI	0.7582±0.00	0.8057±0.00	0.5684±0.00	0.0254±0.03	0.7235±0.03	0.7654±0.05	0.7756±0.02	0.7012±0.01	0.0267±0.01	0.7786±0.02	**0.8695±0.04**
	ARI	0.7302±0.00	0.7592±0.00	0.5475±0.00	0.0076±0.02	0.6427±0.05	0.7185±0.08	0.7951±0.01	0.6354±0.02	0.0003±0.00	0.7795±0.03	**0.8923±0.04**
WI	ACC	0.4933±0.01	0.4944±0.00	0.3837±0.01	0.4062±0.03	0.5034±0.02	0.4382±0.00	0.5775±0.01	0.5000±0.01	0.3933±0.02	0.7472±0.00	**0.9646±0.00**
	NMI	0.1123±0.01	0.1412±0.00	0.0279±0.01	0.0266±0.02	0.0923±0.01	0.0500±0.00	0.1629±0.03	0.0875±0.01	0.0221±0.01	0.3655±0.01	**0.8742±0.01**
	ARI	0.1169±0.01	0.1250±0.00	-0.0089±0.00	0.0062±0.03	0.1094±0.01	0.0399±0.00	0.1720±0.02	0.1052±0.00	-0.0016±0.00	0.3899±0.00	**0.8945±0.01**
AR	-	5.43	7.13	7.53	9.53	4.93	6.26	5.06	6.80	8.40	3.60	1.10

a unified distance metric is adopted to perform a more reasonable graph construction (*i.e.* Section 4.2), which closely connects different chessboard statuses corresponding to the same cluster (*i.e.* winning or losing) to reduce the difficulty of clustering.

5) On pure numerical data sets, *i.e.*, GI, YE, II, and WI, QGRL still demonstrates its superiority as the adopted quaternion representation learning works well in learning the relationship among features and provides a discriminative data representation for clustering.

5.3 Significance Study

We conduct significance tests between our method and other methods using the Wilcoxon signed rank test based on the clustering performance reported in Table 3. We show the p-values of comparisons in Figure 2, and a darker color represents a smaller p-value. It can be observed that all comparisons demonstrate the significant superiority of our method at a 99% confidence level. It is noteworthy that the Wilcoxon signed rank test is performed based on the performance ranking of methods, so the p-values have a relatively large granularity. For example, since our method ranks first in ARI performance across all the data sets, the calculated p-value is always consistent at 0.00195 in Figure 2.

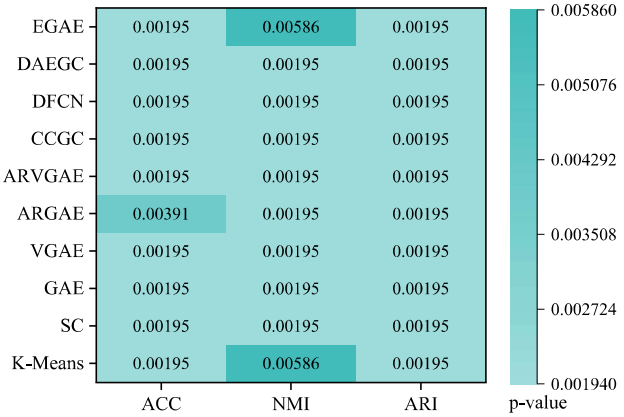


Figure 2: p-values of Wilcoxon signed rank test in comparing our method against the other methods on three metrics.

5.4 Ablation Study of QGRL

The results of ablation studies are illustrated in Table 4, where "Baseline" refers to the GAE model proposed in [15] combined with the KL loss (Section 4.4), one-hot encoding (Section 5.1.3), and Euclidean distance-based graph construction (Remark 3). Three models are further considered, including 1) "Baseline+HCE": Baseline with the HCE proposed in Section 4.1 and 4.2, 2) "Baseline+QRL": Baseline with the proposed QRL presented in Section 4.3, and 3) The full model of our proposed QGRL. The discussions are as follows.

Table 4: Clustering performance comparison of ablated versions of the proposed QGRL model.

Data set	Metric	Baseline	Baseline+HCE	Baseline+QRL	QGRL (ours)
HF	ACC	0.6261±0.01	0.7080±0.00	0.6528±0.02	0.7137±0.03
	NMI	0.0218±0.00	0.0606±0.00	0.0404±0.02	0.1092±0.03
	ARI	0.0513±0.01	0.1310±0.00	0.0811±0.03	0.1663±0.04
BC	ACC	0.7028±0.00	0.5245±0.00	0.5462±0.01	0.7203±0.00
	NMI	0.0627±0.00	0.0181±0.00	0.0085±0.01	0.0866±0.01
	ARI	0.1366±0.00	-0.0077±0.00	0.0060±0.01	0.1700±0.00
AA	ACC	0.5202±0.01	0.5347±0.02	0.6346±0.01	0.6356±0.01
	NMI	0.0013±0.00	0.0073±0.01	0.0449±0.02	0.0471±0.01
	ARI	-0.0123±0.00	0.0052±0.01	0.0386±0.02	0.0375±0.02
MM	ACC	0.5499±0.00	0.5346±0.02	0.6819±0.00	0.8296±0.00
	NMI	0.0077±0.00	0.0073±0.01	0.0997±0.00	0.3487±0.01
	ARI	0.0089±0.00	0.0052±0.01	0.1313±0.00	0.4340±0.01
ZO	ACC	0.6505±0.01	0.5357±0.02	0.7030±0.03	0.8020±0.02
	NMI	0.6345±0.04	0.0075±0.01	0.7571±0.02	0.8212±0.02
	ARI	0.4988±0.03	0.0053±0.01	0.5617±0.02	0.7733±0.03
TTT	ACC	0.6034±0.01	0.6138±0.00	0.6153±0.02	0.9574±0.03
	NMI	0.0124±0.00	0.0190±0.00	0.0461±0.03	0.7633±0.10
	ARI	0.0340±0.01	0.0444±0.00	0.0525±0.02	0.8381±0.09

1) Baseline+HCE does not achieve obvious improvements in comparison with Baseline. This is because HCE focuses on representing hierarchical couplings, which is the basis for quaternion representation. By only combining with the baseline GAE, the coupling relationships cannot be adequately learned, and may thus lead to redundant information in the final representation.

2) Baseline+QRL is observed to deliver superior results on some data sets, such as AA and MM, which shows that the proposed QRL can learn rich feature representations when dealing with data of limited information richness. However, the model performance deteriorated on the BC data set. This could be attributed to the inadequate encoding of the hierarchical coupling for heterogeneous features without HCE, consequently, preventing the proper learning of relationships and patterns in QRL.

3) The advantage of QGRL is not obvious compared to Baseline+QRL on AA data set. The reason is that most of the categorical features of AA are Boolean-valued and are relatively independent of each other. This makes the HCE of QGRL have no significant impact. In addition, the QRL module commonly adopted by QGRL and Baseline+QRL dominates the performance on AA dataset as QRL acts to enhance the degree of representation learning freedom. This is why Baseline+QRL and QGRL have similar performance, and why they both significantly outperform the two versions without QRL, *i.e.* Baseline and Baseline+HCE, on AA data set.

4) The complete QGRL achieves significant improvements on almost all the data sets, which indicates that the proposed HCE and QRL modules can cooperate to achieve better representation. That is, HCE provides an informative encoding basis as the model input, while QRL demonstrates powerful coupling learning on the basis. As a result, a comprehensive and discriminative representation is obtained for accurate clustering.

5.5 Visualization Analysis

In this part, we visualize the cluster distribution of embeddings obtained by two representative methods, *i.e.*, CCGC and EGAE, and the proposed QGRL via t-SNE [33], to intuitively show the effectiveness and superiority of QGRL. From Figure 3, it can be seen that QGRL better reveals the intrinsic clustering distributions than the two compared methods.

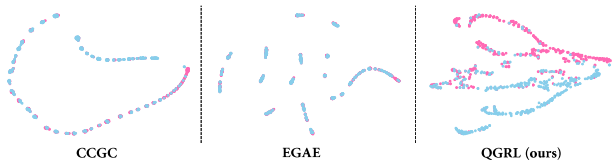


Figure 3: t-SNE visualization of representations obtained by CCGC, EGAE, and the proposed QGRL on MM data set. Pink and blue mark the objects with the "true" cluster labels.

6 CONCLUDING REMARKS

This paper proposes a novel quaternion graph representation learning method called QGRL for heterogeneous feature data clustering. To more comprehensively learn the representations of heterogeneous numerical and categorical features, the complex hierarchical couplings are carefully encoded into the form of attributed graphs to uncover the value-level, feature-level, heterogeneous, and object-level relation in the data. To obtain a more concise form of the data representations for clustering, we integrate the powerful quaternion representation learning and spectral clustering objective to perform unsupervised representation learning. It turns out that clustering-friendly representations can be adequately learned on the basis of informative heterogeneous feature encoding. As QGRL is designed for representing heterogeneous features, it is surely feasible for any-feature-type data, including numerical data, categorical data, and heterogeneous data. Extensive comparison and ablation studies illustrate the superiority of QGRL in clustering.

While QGRL demonstrates its effectiveness, it is not exempt from limitations, including the requirement of a given number of clusters, relatively high computation cost, inflexibility in processing dynamic or streaming data, *etc.* Our further research will focus on deriving a lightweight network structure from QGRL, proposing a continual learning strategy for the model training, and integrating the search for an optimal number of clusters into the learning process. Moreover, transferring QGRL to other tasks driven by heterogeneous feature data, *e.g.*, classification, anomaly detection, dimensionality reduction, *etc.*, is also promising.

ACKNOWLEDGEMENTS

[redacted]

REFERENCES

[1] Amir Ahmad and Lipika Dey. 2007. A method to compute distance between two categorical values of same attribute in unsupervised learning for categorical data set. *Pattern Recognition Letters* 28, 1 (2007), 110–118.

[2] Amir Ahmad and Shehroz S. Khan. 2019. Survey of State-of-the-Art Mixed Data Clustering Algorithms. *IEEE Access* 7 (2019), 31883–31902.

[3] Madhavi Alamuri, Bapi Raju Surampudi, and Atul Negi. Jul. 2014. A survey of distance/similarity measures for categorical data. In *Processings of the International Joint Conference on Neural Networks*. 1907–1914.

[4] Vladimir Batagelj, Hans-Hermann Bock, Anuska Ferligoj, and Ales Ziberna (Eds.). 2006. *Data Science and Classification*.

[5] Mikhail Belkin and Partha Niyogi. 2001. Laplacian Eigenmaps and Spectral Techniques for Embedding and Clustering. In *Proceedings of the 14th International Conference on Neural Information Processing Systems*. 585–591.

[6] Deyu Bo, Xiao Wang, Chuan Shi, Meiqi Zhu, Emiao Lu, and Peng Cui. Apr. 2020. Structural Deep Clustering Network. In *Proceedings of The Web Conference 2020*. 1400–1410.

[7] Samuel R Bowman, Luke Vilnis, Oriol Vinyals, Andrew M Dai, Rafal Jozefowicz, and Samy Bengio. 2015. Generating sentences from a continuous space. *arXiv preprint arXiv:1511.06349* (2015).

[8] Yiu-ming Cheung and Hong Jia. 2013. Categorical-and-numerical-attribute data clustering based on a unified similarity metric without knowing cluster number. *Pattern Recognition* 46, 8 (2013), 2228–2238.

[9] Danilo Comminiello, Marco Lella, Simone Scardapane, and Aurelio Uncini. Oct. 2019. Quaternion Convolutional Neural Networks for Detection and Localization of 3D Sound Events. In *Processings of the Conference on Acoustics, Speech and Signal Processing*. 8533–8537.

[10] Janez Demsar. 2006. Statistical Comparisons of Classifiers over Multiple Data Sets. *Journal of Machine Learning Research* 7 (2006), 1–30.

[11] Chase J. Gaudet and Anthony S. Maida. Jul. 2018. Deep Quaternion Networks. In *Processings of the International Joint Conference on Neural Networks*. 1–8.

[12] Greg Hamerly and Charles Elkan. 2003. Learning the k in k-means. In *Proceedings of the 16th International Conference on Neural Information Processing Systems*. 281–288.

[13] Dino Ienco, Ruggero G. Pensa, and Rosa Meo. Sep. 2009. Context-Based Distance Learning for Categorical Data Clustering. In *Processings of the 8th International Symposium on Intelligent Data Analysis*, Vol. 5772. 83–94.

[14] Hong Jia, Yiu-ming Cheung, and Jiming Liu. 2016. A New Distance Metric for Unsupervised Learning of Categorical Data. *IEEE Transactions on Neural Networks and Learning Systems* 27, 5 (2016), 1065–1079.

[15] T. N. Kipf. 2016. Variational graph auto-encoders. *arXiv preprint arXiv:1611.07308* (2016).

[16] Thomas N. Kipf and Max Welling. Apr. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *Processings of the 5th International Conference on Learning Representations*.

[17] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Dec. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In *Proceedings of the 26th International Conference on Neural Information Processing Systems*. 1106–1114.

[18] Si Quang Le and Tu Bao Ho. 2005. An association-based dissimilarity measure for categorical data. *Pattern Recognition Letters* 26, 16 (2005), 2549–2557.

[19] Xuelong Li, Hongyuan Zhang, and Rui Zhang. 2022. Adaptive Graph Auto-Encoder for General Data Clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44, 12 (2022), 9725–9732.

[20] Dekang Lin. Jul. 1998. An Information-Theoretic Definition of Similarity. In *Processings of the 15th International Conference Machine Learning*. 296–304.

[21] Yue Liu, Jun Xia, Sihang Zhou, Siwei Wang, Xifeng Guo, Xihong Yang, Ke Liang, Wenxuan Tu, Stan Z. Li, and Xinwang Liu. 2022. A Survey of Deep Graph Clustering: Taxonomy, Challenge, and Application. *CoRR* abs/2211.12875 (2022).

[22] Ulrike Luxburg. 2007. A tutorial on spectral clustering. *Statistics and Computing* 17 (2007), 395–416.

[23] Panagiotis Mandros, David Kaltenpoth, Mario Boley, and Jilles Vreeken. 2020. Discovering Functional Dependencies from Mixed-Type Data. In *Processings of the 26th Conference on Knowledge Discovery and Data Mining*. 1404–1414.

[24] Razvan V. Marinescu, Arman Eshaghi, Marco Lorenzi, Alexandra L. Young, Neil P. Oxtoby, Sara Garbarino, Sebastian J. Crutch, and Daniel C. Alexander. 2019. DIVE: A spatiotemporal progression model of brain pathology in neurodegenerative disorders. *NeuroImage* 192 (2019), 166–177.

[25] Kolby Nottingham Markelle Kelly, Rachel Longjohn. 2023. The UCI Machine Learning Repository. https://archive.ics.uci.edu.

[26] Nairouz Mrabah, Mohamed Bouguessa, Mohamed Fawzi Touati, and Riadh Ksantini. 2023. Rethinking Graph Auto-Encoder Models for Attributed Graph Clustering. *IEEE Transactions on Knowledge and Data Engineering* 35, 9 (2023), 9037–9053.

[27] Dai Quoc Nguyen, Tu Dinh Nguyen, and Dinh Q. Phung. Nov. 2021. Quaternion Graph Neural Networks. In *Processings of the Asian Conference on Machine Learning*, Vol. 157. 236–251.

[28] Shirui Pan, Ruiqi Hu, Guodong Long, Jing Jiang, Lina Yao, and Chengqi Zhang. Jul. 2018. Adversarially Regularized Graph Autoencoder for Graph Embedding. In *Processings of the 27th International Joint Conference on Artificial Intelligence*. 2609–2615.

[29] Titouan Parcollet, Mohamed Morchid, and Georges Linarès. 2020. A survey of quaternion neural networks. *Artificial Intelligence Review* 53, 4 (2020), 2957–2982.

[30] Titouan Parcollet, Ying Zhang, Mohamed Morchid, Chiheb Trabelsi, Georges Linarès, Renato de Mori, and Yoshua Bengio. Sep. 2018. Quaternion Convolutional Neural Networks for End-to-End Automatic Speech Recognition. In *Processings of the 19th Conference of the International Speech Communication Association*. 22–26.

[31] Yuhua Qian, Feijiang Li, Jiye Liang, Bing Liu, and Chuangyin Dang. 2016. Space structure and clustering of categorical data. *IEEE Transactions on Neural Networks Learning Systems* 27, 10 (2016), 2047–2059.

[32] Wenxuan Tu, Sihang Zhou, Xinwang Liu, Xifeng Guo, Zhiping Cai, En Zhu, and Jieren Cheng. 2021. Deep Fusion Clustering Network. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence*. 9978–9987.

[33] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research* 9, 86 (2008), 2579–2605.

[34] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. 2008. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th International Conference on Machine Learning*. 1096–1103.

[35] Chun Wang, Shirui Pan, Ruiqi Hu, Guodong Long, Jing Jiang, and Chengqi Zhang. Aug. 2019. Attributed Graph Clustering: A Deep Attentional Embedding Approach. In *Processings of the 28th International Joint Conference on Artificial Intelligence*. 3670–3676.

[36] Xihong Yang, Yue Liu, Sihang Zhou, Siwei Wang, Wenxuan Tu, Qun Zheng, Xinwang Liu, Liming Fang, and En Zhu. Feb. 2023. Cluster-Guided Contrastive Graph Clustering Network. In *Processings of the 37th AAAI Conference on Artificial Intelligence*. 10834–10842.

[37] Hongyuan Zhang, Pei Li, Rui Zhang, and Xuelong Li. 2023. Embedding Graph Auto-Encoder for Graph Clustering. *IEEE Transactions on Neural Networks and Learning Systems* 34, 11 (2023), 9352–9362.

[38] Yiqun Zhang and Yiu-ming Cheung. 2022. Learnable Weighting of Intra-Attribute Distances for Categorical Data Clustering with Nominal and Ordinal Attributes. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44, 7 (2022), 3560–3576.

[39] Yiqun Zhang and Yiu-ming Cheung. 2023. Graph-Based Dissimilarity Measurement for Cluster Analysis of Any-Type-Attributed Data. *IEEE Transactions on Neural Networks and Learning Systems* 34, 9 (2023), 6530–6544.

[40] Yiqun Zhang and Yiu-ming Cheung. Oct. 2018. Exploiting Order Information Embedded in Ordered Categories for Ordinal Data Clustering. In *Processings of the 24th International Symposium on Methodologies for Intelligent Systems*, Vol. 11177. 247–257.

[41] Yiqun Zhang, Yiu-ming Cheung, and Kay Chen Tan. 2020. A Unified Entropy-Based Distance Metric for Ordinal-and-Nominal-Attribute Data Clustering. *IEEE Transactions on Neural Networks and Learning Systems* 31, 1 (2020), 39–52.

[42] Yiqun Zhang, Yiu-ming Cheung, and An Zeng. Jul. 2022. Het2Hom: Representation of Heterogeneous Attributes into Homogeneous Concept Spaces for Categorical-and-Numerical-Attribute Data Clustering. In *Processings of the 31st International Joint Conference on Artificial Intelligence*. 3758–3765.

[43] Yiqun Zhang and Yiu-ming Cheung. 2020. A new distance metric exploiting heterogeneous interattribute relationship for ordinal-and-nominal-attribute data clustering. *IEEE Transactions on Cybernetics* 52, 2 (2020), 758–771.

[44] Yiqun Zhang and Yiu-ming Cheung. 2020. An ordinal data clustering algorithm with automated distance learning. In *Processings of the 34th AAAI Conference on Artificial Intelligence*, Vol. 34. 6869–6876.

[45] Zewen Zheng, Guoheng Huang, Xiaochen Yuan, Chi-Man Pun, Hongrui Liu, and Wing-Kuen Ling. 2023. Quaternion-Valued Correlation Learning for Few-Shot Semantic Segmentation. *IEEE Transactions on Circuits and Systems for Video Technology* 33, 5 (2023), 2102–2115.

[46] Sheng Zhou, Hongjia Xu, Zhuonan Zheng, Jiawei Chen, Zhao Li, Jiajun Bu, Jia Wu, Xin Wang, Wenwu Zhu, and Martin Ester. 2022. A Comprehensive Survey on Deep Clustering: Taxonomy, Challenges, and Future Directions. *CoRR* abs/2206.07579 (2022).

[47] Chengzhang Zhu, Longbing Cao, and Jianping Yin. 2022. Unsupervised Heterogeneous Coupling Learning for Categorical Representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44, 1 (2022), 533–549.

[48] Xuanyu Zhu, Yi Xu, Hongteng Xu, and Changjian Chen. Sep. 2018. Quaternion Convolutional Neural Networks. In *Processings of the 15th European Conference on Computer Vision*, Vol. 11212. 645–661.

Robust Categorical Data Clustering Guided by Multi-Granular Competitive Learning

Shenghong Cai[1], [redacted]

2024 IEEE 44th International Conference on Distributed Computing Systems (ICDCS) | 979-8-3503-8605-9/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICDCS60910.2024.00035

***Abstract*—Data set composed of categorical features is very common in big data analysis tasks. Since categorical features are usually with a limited number of qualitative possible values, the nested granular cluster effect is prevalent in the implicit discrete distance space of categorical data. That is, data objects frequently overlap in space or subspace to form small compact clusters, and similar small clusters often form larger clusters. However, the distance space cannot be well-defined like the Euclidean distance due to the qualitative categorical data values, which brings great challenges to the cluster analysis of categorical data. In view of this, we design a Multi-Granular Competitive Penalization Learning (MGCPL) algorithm to allow potential clusters to interactively tune themselves and converge in stages with different numbers of naturally compact clusters. To leverage MGCPL, we also propose a Cluster Aggregation strategy based on MGCPL Encoding (CAME) to first encode the data objects according to the learned multi-granular distributions, and then perform final clustering on the embeddings. It turns out that the proposed MGCPL-guided Categorical Data Clustering (MCDC) approach is competent in automatically exploring the nested distribution of multi-granular clusters and highly robust to categorical data sets from various domains. Benefiting from its linear time complexity, MCDC is scalable to large-scale data sets and promising in pre-partitioning data sets or compute nodes for boosting distributed computing. Extensive experiments with statistical evidence demonstrate its superiority compared to state-of-the-art counterparts on various real public data sets.**

***Index Terms*—Cluster analysis, categorical feature, competitive learning, number of clusters, cluster granularity**

I. Introduction

Clustering is one of the most popular unsupervised learning techniques that divides objects into a certain number of clusters where each cluster is usually composed of similar objects [1], [2]. Clustering can be utilized as a learner for recognition tasks, including anomaly detection [3], recommendation [4], risk detection [5], etc. It can also be utilized as a general tool for mining knowledge from data, e.g., latent object distribution [6], potential feature association [7], etc. However, most existing clustering is based on numerical data where all the feature values are quantitative and can directly attend arithmetic calculation [8]–[10]. Another common type of data, i.e., categorical data [11], [12] composed of qualitative feature values as shown in Fig. 1, is usually overlooked.

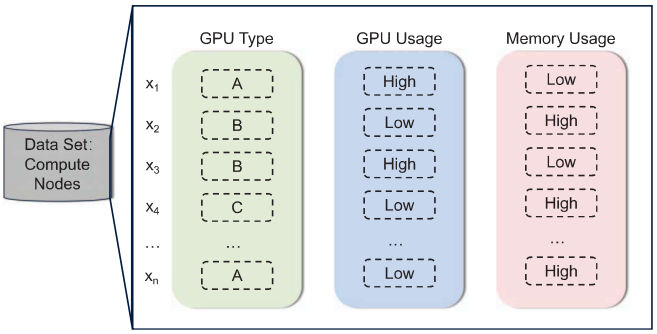


Fig. 1: Three categorical features (i.e., "GPU Type", "GPU Usage" and "Memory Usage") of a data set describing different compute nodes.

A common way for categorical data clustering is to encode the qualitative values into quantitative numerical values. However, the encoding process may easily cause information loss [13] as the implicit distribution of categorical data is difficult to be appropriately mapped into the Euclidean distance space. By contrast, some conventional categorical data clustering approaches [13]–[16] directly perform clustering by adopting distance metrics that are specifically defined for categorical data. Nevertheless, categorical data set is usually composed of features from various domains, which brings great challenges to defining a universal distance metric. Although hierarchical clustering [17] that outputs a dendrogram reflecting nested data object relationship can be utilized to understand the complex distribution of categorical data, the construction of dendrogram is laborious and may even fail due to the lack of powerful categorical data distance metric. Below we discuss the research progress of the above three streams of methods, i.e., 1) encoding-based, 2) distance defining-based, and 3) hierarchical clustering methods, and then refine the specific cutting-edge problems to be solved in this paper.

For the encoding-based stream, besides the most commonly used one-hot encoding [18], more advanced strategies [19], [20] that further consider the value-, object-, and feature-level couplings have been proposed for more informative represen-

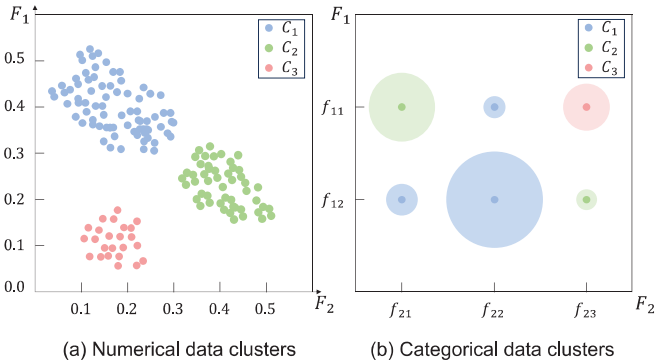
2575-8411/24/$31.00 ©2024 IEEE
DOI 10.1109/ICDCS60910.2024.00035

tation. To make the representation learnable with the downstream clustering tasks, representation learning approaches [21]–[23] have also been proposed and obtained better categorical data clustering performance. Later, the research [24] further presents novel learning strategies to circumvent the non-trivial hyper-parameter selection of representation learning. Most recently, [25] introduces a multi-view projection technique to extract a more comprehensive representation of categorical feature values. Although the above-mentioned approaches have successively refreshed the clustering performance, they all focus on the improvement of encoding and learning techniques rather than the understanding of complex distributions and cluster effects of categorical data.

For the distance defining-based stream, the most popular Hamming distance [26] assigns distances "0" and "1" to pairs of identical and different values, respectively, from the perspective of the most basic value matching perspective. Entropy-based measures [27]–[31] and probability-based metrics [32]–[36], propose to quantify object-cluster affiliation based on more informative data statistics, e.g., occurrence frequency and conditional probability distribution of possible values, and have achieved more satisfactory clustering performance. Noticing the damage to the clustering performance caused by the heterogeneity of numerical and categorical features, some more advanced metrics [37]–[39] further unify the distance definitions of the two types of features from the perspective of probability. However, they mainly focus on the unification issue, and their performance will degrade when processing pure categorical data. The above measures and metrics are often combined with a partitional clustering algorithm for categorical data clustering. Since most of these algorithms require a given number of sought clusters k^*, they are incompetent in exploring and understanding the natural cluster distribution of categorical data.

Hierarchical clustering is considered a promising way for data distribution visualization and understanding, as it produces a tree-like nesting of data objects by recursively linking the current most similar object pairs. Representative link strategies include the conventional average-, complete-, and single-linkage [17], while recent advanced strategies [40]–[42] have also been explored. A link strategy that is specifically designed for categorical data has also been introduced in [43]. However, since the adopted dissimilarity metric acts as the basis for linkage computation, and hierarchical clustering lacks a learning mechanism capable of optimizing the metric, metric inappropriateness will all be unavoidably inherited. Moreover, as objects are treated as basic units during the recursive merging, implementing hierarchical clustering to large categorical data will be laborious. Although the most recent work [44] attempts to utilize statistical tests to guide the detection of significant clusters, unfortunately, the inherent bias of statistical tests makes them incapable of simultaneously detecting the multi-granular clusters that are prevalent in categorical data as shown in Fig. 2.

It can be seen that the limited feature values of categorical data make the objects overlapped at several points (e.g., the six points in Fig. 2(b)) in the distance space. The overlapping objects can be viewed as fine-grained clusters. Several such clusters form a larger cluster at a more coarse granularity, and so on, forming the nesting of multi-granular clusters. Based on the aforementioned analysis, it can be concluded that there is still a lack of cluster analysis methods that can effectively reveal the complex nested multi-granular cluster effect and are universally applicable to categorical data sets composed of qualitative features from various conceptual domains.



(a) Numerical data clusters (b) Categorical data clusters

Fig. 2: Comparison of clusters of numerical and categorical data. Since categorical data objects overlap on six points in (b), spheres with different radii are used to indicate the occurrence frequency of overlapping objects. The natural distance structure of categorical data leads to the nested multi-granular cluster effect (e.g., the green cluster is composed of two clusters with different granularity), which brings difficulties to cluster analysis.

This paper, therefore, proposes a new cluster analysis framework called MGCPL-guided Categorical Data Clustering (MCDC). First, a Multi-Granular Competitive Penalization Learning (MGCPL) mechanism is designed to automatically learn object partitions at different granularities by making the learning converge at different numbers of clusters. As MGCPL treats small clusters as the basic unit, the laborious nested relationship analysis is thus greatly alleviated. Compared with hierarchical clustering, the introduced learning mechanism facilitates intelligent multi-granular cluster detection. To leverage the analysis results of MGCPL, Cluster Aggregation based on MGCPL Encoding (CAME) has also been proposed to obtain partitional clustering results based on a given number of sought clusters. As its name suggests, clusters explored by MGCPL at each granularity are encoded to obtain informative embeddings, where the multi-granular information may complement each other and thus achieve more accurate clustering. It is worth noting that most existing clustering algorithms can be applied to the embeddings to obtain performance improvements. It turns out that the proposed method is robust and accurate on real categorical data sets from various domains, and its linear time complexity makes it highly scalable. Extensive experimental evaluations provide sufficient evidence of its effectiveness and efficiency.

The main contributions can be summarized into three-fold:

TABLE I: Frequently used symbols in the paper.

Symbols	Explanations
X	Data set
F	Features
C	Clusters
n	Number of data objects
d	Number of features
Q	Partition matrix of data objets
C_v	The winning cluster
C_h	The rival nearest winner
η	Learning rate
u	Cluster weights during competitive learning
σ	Number of granularity levels learned by MGCPL
κ	A series of ks learned by MGCPL
Γ	Data representation guided by MGCPL
Θ	Feature weights of representation Γ
k^*	The true number of clusters
Z	Mode of clusters

- To the best of our knowledge, this is the first attempt to reveal the complex but ubiquitous nested multi-granular cluster effect in categorical data, which is promising in inspiring subsequent works on categorical data analysis.
- A new cluster analysis mechanism called MGCPL is proposed to explore the nested multi-granular clusters of categorical data. MGCPL is efficient, and can provide rich data representation information for downstream tasks.
- An aggregation strategy called CAME is designed to fuse the multi-granular results of MGCPL. CAME achieves more accurate clustering, and its representation can also enhance existing categorical data clustering methods.

II. Preliminaries

This section first briefly introduces basic notations and frequently used symbols (see Table I), and then presents categorical data distance measurement and competitive learning mechanism that are highly related to the proposed method.

Given a categorical data set $X = \{x_i | i = 1, 2, ..., n\}$ with n data objects. Each object x_i is represented by d features $\{F_r | r = 1, 2, ..., d\}$. Thus, x_i can be represented as $x_i = [x_{i1}, x_{i2}, ..., x_{id}]^\top$ with $x_{ir} \in dom\,(F_r)$ and $r = 1, 2, ..., d$ where $dom\,(F_r) = \{f_{r1}, f_{r2}, ..., f_{rm_r}\}$ contains all the m_r possible values that can be chosen by feature F_r. In the common partitional clustering tasks, X should be divided into k clusters $C = \{C_l | l = 1, 2, ..., k\}$, i.e., a collection of k disjoint subsets of X, where C_l is the set of objects in the lth cluster and $X = \bigcup_{l=1}^{k} C_l$. Since distance measurement plays a key role in most existing categorical data clustering algorithms, we present an object-cluster distance measure for categorical data in the following.

A. Categorical Data Distance Measure

To achieve better adaptability between distance definition and clustering task, an object-cluster similarity denoted as $s\,(x_i, C_l)$ can be defined as

$$s\,(x_i, C_l) = \frac{1}{d}\left[\sum_{r=1}^{d} s(x_{ir}, C_l)\right] \tag{1}$$

where

$$s(x_{ir}, C_l) = \frac{\Psi_{F_r = x_{ir}}(C_l)}{\Psi_{F_r \neq NULL}(C_l)} \tag{2}$$

is the similarity reflected by the rth feature. Note that $\Psi_{F_r = x_{ir}}(C_l)$ counts the number of objects in cluster C_l that have value x_{ir} in feature F_r, and $\Psi_{F_r \neq NULL}(C_l)$ means the number of objects in cluster C_l that have values in the feature F_r. Intuitively, $s\,(x_i, C_l)$ is the average of similarities reflected by different features. Then we introduce how the competitive learning mechanism works on categorical data based on the above-defined distance to explore clusters.

B. Competitive Learning Algorithm

Most existing clustering methods assume the known true cluster number k^*, which is usually unavailable in real data analysis tasks, especially for data sets with complex distributions like categorical data. Competitive learning [45] mechanism is thus designed to learn the true number of clusters. The core idea of competitive learning is to make the initialized clusters compete with each other to eliminate clusters with less importance, and thus its objects will be carved up by the remaining clusters. In this way, by setting a relatively large initial k, the algorithm can gradually converge to k^* with a more stable and prominent cluster distribution. Such a learning mechanism is to maximize the overall intra-cluster similarity $S(Q)$:

$$S(Q) = \sum_{l=1}^{k}\sum_{i=1}^{n} u_l q_{il} s(x_i, C_l) \tag{3}$$

where q_{il} is the (i, l)th entry of Q, and is computed by

$$q_{il} = \begin{cases} 1, \text{ if } s(x_i, C_l) \geq s(x_i, C_t) \;\; \forall 1 \leq t \leq k \\ 0, \text{ otherwise.} \end{cases} \tag{4}$$

u_l is the weight of cluster C_l satisfying $0 \leq u_l \leq 1$ with $l = 1, 2, ..., k$. It measures the importance of C_j, and a higher weight indicates that the corresponding cluster is more prominent with less possibility to be eliminated.

During clustering, competitive learning is facilitated as follows. For each input x_i, the winning cluster C_v selected from initialized cluster candidates by

$$v = \arg\max_{1 \leq l \leq k} [u_l s(x_i, C_l)] \tag{5}$$

is updated toward x_i by a small step. To avoid the effect that some seed points located in marginal positions will immediately become dead units without learning chance in the subsequent learning process, wining chance of a frequent winning seed point will be gradually reduced. Accordingly, the winning frequency of different clusters can be computed to adjust the selection chance of the winner, and thus Eq. (5) can be re-written as

$$v = \arg\max_{1 \leq l \leq k} [(1 - \rho_l) u_l s(x_i, C_l)] \tag{6}$$

where

$$\rho_l = \frac{g_l}{\sum_{t=1}^{k} g_t} \tag{7}$$

is a winning ratio computed based on g_l, which is the winning times of cluster C_l in the last learning iteration. Accordingly, the weight of cluster C_v is updated by a small step controlled by a small learning rate η, which can be written as

$$u_l^{new} = u_l^{old} + \eta. \tag{8}$$

Note that the value of initial k should be set at a larger value than k^*, i.e., $k \geq k^*$, to ensure that the redundant clusters can be gradually eliminated during the learning process.

III. Proposed Method

This section first presents the MGCPL algorithm for exploring nested multi-granular distribution of clusters, and then introduces the CAME aggregation strategy to combine the multi-granular information provided by MGCPL to obtain the clustering results. The overall pipeline of the cluster analysis method composed of MGCPL and CAME is demonstrated in Fig. 3. Time complexity analysis, discussions on convergence and distributed computing issues, are provided at the end of this section.

A. MGCPL: Multi-granular Competitive Penalization Learning

In most real data cluster analysis tasks, it is not the case that a true number of cluster k^* can be known in advance, especially for categorical data with complex non-Euclidean distance space that are difficult to intuitively understand. Therefore, one of the most important issues in clustering is to estimate the most appropriate number of clusters. However, it is common that there are several ks suitable for the same data set, as the clusters can exist at different granularities, which is called the multi-granular effect. Such an effect is particularly evident in categorical data, because the categorical features are with limited number of possible values, making data objects overlap in the distance space as discussed in the Introduction.

Hence, for a categorical data set, it is necessary to explore a series of suitable numbers of clusters $\kappa = \{k_1, k_2, ..., k_\sigma\}$ where k_σ is the one corresponding to the partition of data objects with the most coarse granularity. As existing competitive learning algorithms aim to find k^* only, we proposed MGCPL to find all the suitable ks in κ. The basic idea is to start the competitive learning with a relatively large initial k_0, and let the k_0 clusters compete with each other to eliminate less important ones to obtain k_1. By inheriting the previously learned k_1 as the initialization, the learning is relunched by clearing the parameters that guide the convergence. Such a process is recursively implemented until the overall MGCPL converges at k_σ where the coarse-grained clusters are prominent enough and no more clusters can be learned to eliminate.

The competitive learning mechanism described by Eq. (3)-Eq. (6) only awards the winning cluster while neglecting the rival cluster, which makes the winners gradually absorb the surrounding seed points and thus not conducive to exploring multi-granular clusters. To avoid this, a rival penalization mechanism is introduced. Specifically, for each input x_i, the winning cluster C_v selected from the initialized cluster candidates is updated toward x_i, while the rival nearest C_h to the winner C_v is determined by

$$h = \arg \max_{1 \leq l \leq k, l \neq v} [(1 - \rho_l) u_l s(x_i, C_l)] \tag{9}$$

where ρ_l is defined in Eq. (7). For each data object x_i, when the winning cluster and its rival nearest are determined, x_i will be assigned to the winning cluster C_v, and the corresponding winning time is updated by

$$g_v = g_v + 1. \tag{10}$$

In Eq. (9), u_l is the weight of C_l computed by

$$u_l = \frac{1}{1 + e^{(-10\delta_l + 5)}} \tag{11}$$

with $l \in \{1, 2, ..., k\}$. Such a commonly used Sigmoid function form is to ensure a more sensitive updating of rival weights and make its values in the interval [0,1]. Accordingly, the updating of u_l can be accomplished by changing the value of δ_l instead. Subsequently, the winner C_v is awarded with

$$\delta_v^{new} = \delta_v^{old} + \eta \tag{12}$$

and the rival C_h is penalized with

$$\delta_h^{new} = \delta_h^{old} - \eta s(x_i, C_l) \tag{13}$$

where η is a small learning rate. As a result, the rival is penalized a step away from the winner, and thus the rivals obtain more opportunities to explore the cluster distributions in the distance space.

It is usually assumed that all the categorical features have the same contribution during the object-cluster similarity measurement. But in practice, as features are of different importance in forming clusters of different data sets, we improve Eq. (1) with a weighting mechanism by

$$s(x_i, C_l) = \frac{1}{d} \left[\sum_{r=1}^{d} \omega_{rl} s(x_{ir}, C_l) \right] \tag{14}$$

where ω_{rl} with $0 \leq \omega_{rl} \leq 1$ is the weight of the rth feature to cluster C_l, and we have $\sum_{r=1}^{d} \omega_{rl} = 1$ with $l \in \{1, 2, ..., k\}$. Since feature-weight ω_{rl} is changing with the change of feature-cluster contribution, we use H_{rl} to indicate the contribution of feature F_r to cluster C_l. To compute H_{rl}, we should first introduce two important terms, i.e., inter-cluster difference α_{rl} and intra-cluster similarity β_{rl}, where α_{rl} measures the ability of feature F_r in distinguishing cluster C_l from the others, while β_{rl} evaluates whether the cluster C_l along the feature F_r has a compact structure. We formulate α_{rl} by

$$\alpha_{rl} = \frac{1}{\sqrt{2}} \sqrt{\sum_{t=1}^{m_r} \left(\frac{\Psi_{F_r = f_{rt}}(C_l)}{\Psi_{F_r \neq NULL}(C_l)} - \frac{\Psi_{F_r = f_{rt}}(X \setminus C_l)}{\Psi_{F_r \neq NULL}(X \setminus C_l)} \right)^2} \tag{15}$$

and calculate β_{rl} by

$$\beta_{rl} = \frac{1}{n_l} \sum_{x_i \in C_l} \frac{\Psi_{F_r = x_{ir}}(C_l)}{\Psi_{F_r \neq NULL}(C_l)} \tag{16}$$

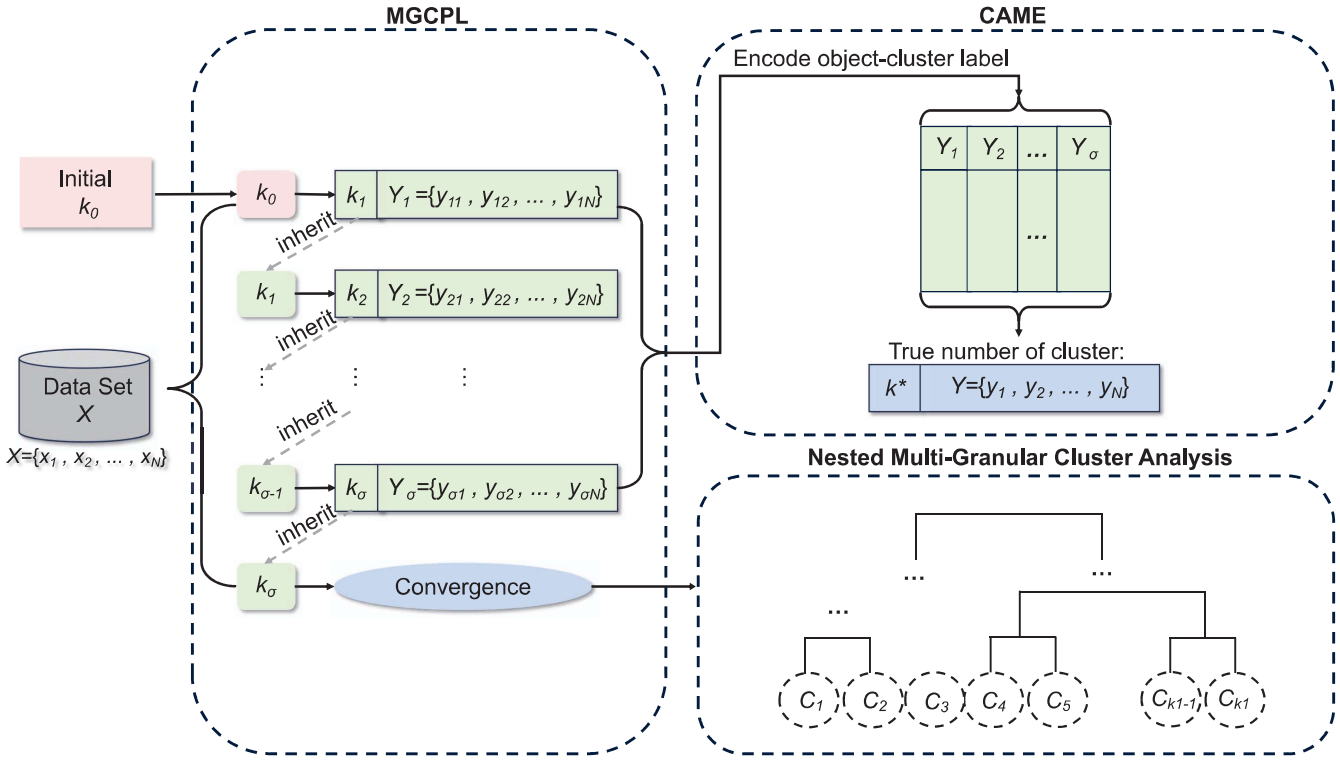


Fig. 3: Pipeline of the proposed method. MGCPL starts its learning with a relatively large initial k_0. The initialized clusters compete with each other to eliminate less important ones and obtain k_1. By inheriting k_1 as the initialization, the learning is re-lunched by clearing the parameters that guide the convergence. Such a process is recursively implemented until converges at k_σ where the k_σ prominent coarse-grained clusters cannot be further eliminated. The multi-granular results can be utilized for nested cluster distribution analysis, and can also be aggregated by CAME to accurately partition X into k clusters.

where n_l is the number of objects in C_l. When both α_{rl} and β_{rl} reach large values, it implies the important contribution of feature F_r in detecting C_l, and thus H_{rl} can be obtained by

$$H_{rl} = \alpha_{rl}\beta_{rl} \tag{17}$$

accordingly. Then the corresponding probabilistic feature weight ω_{rl} can be calculated by

$$\omega_{rl} = \frac{H_{rl}}{\sum_{t=1}^{d} H_{tl}} \tag{18}$$

with $r \in \{1, 2, ..., d\}$ and $l \in \{1, 2, ..., k\}$.

To facilitate the learning of multi-granular clusters by using the above-described learning process, we initialize a larger number of clusters k_0 to launch the learning. When the above-defined competitive penalization learning converges with k_1, i.e., an appropriate number of clusters at a fine granularity corresponding to k_1 have been explored, we let the learning mechanism inherit k_1 and re-launch the learning to explore coarser-grained clusters. To re-launch the learning in a new epoch, all the previous statistics are reset by $g_l = 0$, $u_l = 1/d$, and $\delta_l = 1$ with $l = \{1, 2, ..., k_\sigma\}$. Competitive penalization learning is recursively launched until it obtains the same partition as the previous epoch. In this way, we can obtain a series of numbers of clusters with decreasing values $\kappa = \{k_1, k_2, ..., k_\sigma\}$ where k_σ is the obtained k with the smallest value. At the same time, we obtain a series of partitions, i.e., clustering results, which can be represented as a collection of object labels, i.e., $\Gamma = \{Y_1, Y_2, ..., Y_\sigma\}$ with $Y_\sigma = \{y_{\sigma1}, y_{\sigma2}, ..., y_{\sigma n}\}$. The whole MGCPL algorithm is summarized as Algorithm 1.

B. CAME: Cluster Aggregation based on MGCPL Encoding

The multi-granular cluster distribution information obtained by MGCPL can be utilized to form an informative representation of categorical data. We thus propose a new encoding strategy that uses the object-cluster affiliation Γ as the data representation. Then, we implement categorical data clustering on the new representation. The advantage of Γ encoding is that it can make full use of the information provided by each granularity of the data set and convert the heterogeneous information provided by the features from different domains into the object-cluster affiliation learned by MGCPL. Moreover, since the features in Γ provide object-cluster affiliations at different granularities, their contributions to the final clustering of CAME are usually different in terms of the sought number of clusters k. Therefore, we formulate the cluster aggregation in the form of feature importance learning to minimize the

Algorithm 1 MGCPL: Multi-Granular Competitive Penalization Learning Algorithm

Input: Data set X, learning rate η, initialized k_0.
Output: Multi-granular partitions $\Gamma = \{Y_1, Y_2, ..., Y_\sigma\}$ and corresponding numbers of clusters $\kappa = \{k_1, k_2, ..., k_\sigma\}$.

Initialize convergence = false, $k^{initial} = k_0$.
while convergence = false **do**
  change = true, randomly select $k^{initial}$ objects to represent clusters in C.
  while change = true **do**
    for i = 1 to n **do**
      Compute v and h by Eqs. (6) and (9), update q_{iv} by Eq. (4), update learning variables by Eqs. (7) and (10)-(13).
    end for
    if $Q^{new} = Q^{old}$ **then**
      change = false.
    end if
    Update ω_{rl} by Eq. (15)-(18).
  end while
  Set $k^{initial} = k^{new}$, reset g_l = 0, u_l = 1/d and δ_l = 1 with $l = \{1, 2, ..., k^{new}\}$.
  if $k^{new} = k^{old}$ **then**
    convergence = true.
  end if
end while

objective function $P(Q, \Theta)$ as follows

$$P(Q, \Theta) = \sum_{l=1}^{k} \sum_{i=1}^{n} \sum_{r=1}^{\sigma} q_{il} \theta_r d(x_{ir}, Z_{lr}) \tag{19}$$

where q_{il} is the (i, l)th entry of Q defined as

$$q_{il} = \begin{cases} 1, \text{ if } \sum_{r=1}^{\sigma} \theta_r d(x_{ir}, Z_{lr}) \leq \sum_{r=1}^{\sigma} \theta_r d(x_{ir}, Z_{tr}) \\ \qquad\qquad\qquad\qquad \text{for } \forall\ t \in \{1, 2, ..., k\} \\ 0, \text{ otherwise.} \end{cases} \tag{20}$$

In Eq. (19), the variable $\Theta = \{\theta_1, \theta_2, ..., \theta_\sigma\}$ is a set of feature weights to be updated during learning. Z_l represents the mode of lth cluster with $l = \{k_1, k_2, ..., k_\sigma\}$. $s(x_{ir}, Z_{lr})$ is the Hamming distance between feature value of object x_{ir} and feature value of cluster Z_{lr}. Note that we use Z_{lr} here to indicate that this is the cluster mode value from representation Γ rather than the original data set X. In this subsection, all the data values are from Γ.

The weight θ_r that reflects the importance of F_r in Γ is updated by

$$\theta_r = \frac{I_r}{\sum_{t=1}^{\sigma} I_r} \tag{21}$$

where I_r is the overall intra-cluster similarity contributed by F_r, which can be written as

$$I_r = \sum_{l=1}^{k} \sum_{i=1}^{n} \sum_{r=1}^{\sigma} \left[1 - d(x_{ir}, Z_{lr})\right]. \tag{22}$$

Algorithm 2 CAME: Cluster Aggregation based on MGCPL Encoding

Input: Data representation Γ, number of clusters k.
Output: Partition Q, features importance Θ.

Initialize convergence = false and $\theta_r = 1/\sigma$ with $r = \{1, 2, ..., \sigma\}$.
Compute $\tilde{Q}$ according to Eq. (20).
while convergence = false **do**
  Set $Q = \tilde{Q}$, compute $\tilde{\Theta}$ by Eqs. (21) and (22).
  Set $\Theta = \tilde{\Theta}$, compute $\tilde{Q}$ by Eq. (20).
  if $Q = \tilde{Q}$ **then**
    convergence = true.
  end if
end while

A higher intra-cluster similarity of a feature indicates that this feature contributes more to forming clusters with more similar data objects.

Clustering with the above feature weighting can be treated as an optimization problem to minimize Eq. (19). More specifically, we can iteratively solve the following two minimization problems:

1) Fix object partition $Q = \tilde{Q}$, update feature weights $\tilde{\Theta}$;
2) Fix feature weights $\Theta = \tilde{\Theta}$, compute object partition $\tilde{Q}$.

Such a learning process will converge to a minimal solution in a finite number of iterations, and the final clustering result Q can be obtained. We summarize CAME as Algorithm 2.

C. Time Complexity Analysis

Theorem 1. *The time complexity of MGCPL is* $O(dnk_0)$.

Proof. *To analyze the complexity in the worst case, we adopt* k_0 *as the initial* k, I *is the maximum number of iterations to make the competitive penalization learning converge. During the object-cluster similarity computation, as* $n \times k_0$ *pairs of distances should be computed on* d *features, the time complexity is thus* $O(Idnk_0)$ *for similarity computation. Similarly, there are* $d \times k_0$ *weights in total that should be updated based on the statistics obtained by going through all the* n *data objects, and thus the time complexity for weights updating is* $O(dnk_0)$. *As for the updating of* g_l, u_l, *and* δ_l, *their time complexity can be omitted compared to that of similarity computation and weights updating. Since the above parts will be implemented by* σ *times in Algorithm 1, the overall time complexity of MGCPL is* $O(\sigma Idnk_0)$. *As* σ *and* I *are both much smaller than* n, d, *and* k_0 *in practice, the overall time complexity of MGCPL is thus* $O(dnk_0)$. □

Theorem 2. *The time complexity of CAME is* $O(dnk)$.

Proof. *Assume that the clustering process of CAME needs* T *iterations to converge. In each iteration, weights of* σ *features are updated by considering the* n *data objects in* k *clusters, and thus the time complexity of feature weighing is* $O(dnk)$. *In each iteration, all the* n *data objects should also be partitioned into* k *clusters by computing the object-cluster*

distances reflected by σ features, and thus the time complexity is also $O(dnk)$. For T iterations in total, the overall time complexity is $O(Tdnk)$. Since T can be viewed as a small constant in most cases, the overall time complexity of CAME is thus $O(dnk)$. □

D. Discussions on Convergence and Distributed Computing

The proposed whole clustering approach MCDC is composed of MGCPL in Algorithm 1 and CAME in Algorithm 2. The MGCPL component can be viewed as repeatedly implementing competitive penalization learning [38], which is a strict gradient decent process that is guaranteed to converge. For the CAME component, it is actually a process of features weighted k-modes clustering, which has also been proven to converge in [21]. Although we adopt an approximation to more intuitively update the weights by Eq. (21), such an update strategy is still consistent with the minimization of the objective function in Eq. (19), as features that contribute less on the minimization of the objective function are assigned with smaller importance in the next iteration. Therefore, MCDC well converges on all the data sets we used for the experiments. If strict convergence is required in some scenarios, the weights updating mechanism described by Eqs. (21) and (22) can be simply replaced by the updating strategy described in [21] derived via Lagrange multiplier.

The potential contributions of the proposed multi-granular clustering algorithm to distributed computing systems are mainly two-fold:

1) It can be utilized to pre-partition data points into compact subsets to more reasonably allocate them to distributed computing nodes. Specifically, data points described by categorical features are automatically divided into relatively independent and compact micro-clusters, which are automatically merged into larger-scale clusters of different granularities. The multi-granular information obtained in this process can well guide the central server to allocate data sample subsets of different granularities to suitable nodes, flexibly realizing parallel computing without causing significant loss of local correlation information of the data objects.
2) It can be utilized to pre-divide compute nodes described as the data set shown in Fig. 1 to form performance-consistent node networks that are more suitable for certain computing tasks. That is, computing nodes are automatically grouped into multi-granular clusters according to their categorical features. The nodes in the same cluster have relatively consistent computing performance and features, and can thus collaborate more efficiently to complete distributed computing tasks. Therefore, the obtained multi-granular computing node clusters can flexibly guide the selection of uniform nodes according to computing task requirements.

IV. Experiment

This section introduces the experimental design and the selection of counterparts, validity indices, and data sets. Then five parts of experimental results are demonstrated with in-depth discussions for performance evaluation of the proposed MGCPL-guided Categorical Data Clustering (MCDC).

A. Experimental Settings

Five Experiments are conducted to evaluate the proposed method from different perspectives, which are summarized below.

- Clustering performance evaluation: The proposed MCDC method is compared with existing representative clustering approaches by quantifying their clustering performance using different mainstream validity indices.
- Significance test: Wilcoxon signed rank test is conducted on the performance of the compared approaches. A rejection of the null hypothesis indicates a significant outperforming of our proposed method against the counterparts.
- Ablation Study: MCDC is ablated into different versions by successively removing its main technical components, and the performance of these versions is compared to illustrate the effectiveness of the MCDC components.
- Learning process evaluation: MGCPL will converge to different ks during its learning. We visualize the changing of k with the optimal k^* to illustrate the effectiveness of the multi-granular cluster learning mechanism.
- Computational efficiency evaluation: MGCPL can be viewed as an efficient alternative to hierarchical clustering. We thus plotting and comparing its execution time under different ns, ds, and ks with the counterparts.

Nine Counterparts are compared in the comparative experiments, including six representative clustering methods, two variants of MCDC that adopt and enhance two existing categorical data clustering algorithms, and MCDC itself. The six representative counterparts are the conventional k-modes [14] proposed for partitional clustering and ROCK [43] for hierarchically clustering categorical data. Four recent advanced clustering methods, i.e., WOCIL [38], GUDMM [30], FKMAWCW [36], and ADC [39] are also chosen for more convincing comparison. WOCIL is proposed for automatically learning the clusters of mixed data, i.e., data composed of both categorical and numerical features. GUDMM introduces a generalized multi-aspect distance measure based on mutual information. FKMAWCW is a fuzzy k-modes-based approach that learns weights of features to clusters during clustering. ADC utilizes a graph-based dissimilarity measurement for cluster analysis of data composed of any type of features. GUDMM and FKMAWCW are also applied to the multi-granular encoding output of our proposed MCDC. The formed clustering approaches are named MCDC+GUDMM and MCDC+FKMAWCW, respectively. For simplicity, they are abbreviated as MCDC+G. and MCDC+F. hereinafter. Hyper-parameters of the compared methods (if any) are set according to the corresponding source paper. Learning rate η and k_0 of MCDC is set at $\eta = 0.03$ and $k_0 = \sqrt{n}$, respectively. For all the compared methods, the sought number of clusters is set at k^* corresponding to each data set as shown in Table II.

TABLE II: Statistics of the 8 data sets.d, n, and k^* indicate the number of features, the number of objects, and the true number of clusters, respectively.

No.	Data Set	Abbrev.	d	n	k^*
1	Car Evaluation	Car.	6	1728	4
2	Congressional	Con.	16	435	2
3	Chess	Che.	36	3196	2
4	Mushroom	Mus.	22	8124	2
5	Tic Tac Toe	Tic.	9	958	2
6	Vote	Vot.	16	232	2
7	Balance	Bal.	4	625	3
8	Nursery	Nur.	8	12960	5
9	Synthetic (with large n)	Syn_n	10	200000	3
10	Synthetic (with large d)	Syn_d	1000	20000	3

Four Validity Indices[1] are utilized to measure the clustering performance. Clustering Accuracy (ACC) is a commonly used index that ranges from 0 to 1. It computes the ratio of the number of correctly clustered objects to the total number of objects. Adjusted Rand Index (ARI) calculates the consistency of obtained clustering results and true labels by comparing their pairwise matching. Its values range from -1 to 1. Adjusted Mutual Information (AMI) is based on mutual information, which quantifies the matching between obtained clustering results and true labels from the perspective of information theory. Its values also range from -1 to 1. The Fowlkes-Mallows (FM) score is defined as the geometric mean of the pairwise precision and recall, and ranges from 0 to 1. All the adopted indices reflect a better clustering performance with a higher value.

Ten Data Sets are utilized to conduct a comprehensive evaluation. Among them, eight data sets are representative ones downloaded from the UCI Machine Learning Repository[2]. Two synthetic data sets with large n and d are generated with well-separated clusters for the efficiency evaluation. All the data sets are categorical ones, and data objects with missing values are omitted before conducting experiments. Detailed statistics of the data sets are shown in Table II.

B. Clustering Performance Evaluation

The clustering performance of the proposed MCDC is compared with the counterparts on all eight categorical data sets in Table III. Each result in the table is obtained by executing the corresponding method by 50 times and taking the average performance with standard deviation. Please note that MCDC+G. and MCDC+F. are the two variants of MCDC adopting GUDMM and FKMAWCW as the clustering algorithms, respectively. Comparing their performance with the original GUDMM and FKMAWCW can reflect the effectiveness of the proposed MCDC in enhancing the clustering performance of existing clustering methods. In the table, the best and the second-best results on each data set are highlighted using **boldface** and underline, respectively.

[1]https://scikit-learn.org/stable/modules/clustering.html#clustering-evaluation

[2]https://archive.ics.uci.edu/

It can be observed from Table III that MCDC outperforms its counterparts on most data sets. Although MCDC does not perform the best on some data sets, it still ranks second or with a performance that is very close to the best performer. Thus, it can be concluded that, in general, MCDC demonstrates its superiority in terms of both accuracy and robustness. From the table, we can also see that ROCK, FKMAWCW, and GUDMM have unsatisfactory performance on some data sets. This is because they sometimes cannot obtain the pre-set number of clusters and are judged as failed. Moreover, ROCK, WOCIL, and three MCDC variants perform very stable because Rock is a hierarchical clustering approach without random initialization, and WOCIL adopts a very stable initialization mechanism. The performance of MCDC is also very stable because the learned multi-granular information complements each other to form a comprehensive and stable representation of different data sets.

As for MCDC+G. and MCDC+F., it can be seen that the performance of the corresponding GUDMM and FKMAWCW is obviously boosted in most cases by MCDC. This indicates that the proposed MCDC is effective in enhancing different categorical data clustering methods. It can be seen that MCDC, MCDC+G., and MCDC+F. achieve obviously better clustering performance than the other counterparts. Moreover, MCDC+F. performs the best in general. This may be because the corresponding FKMAWCW is a fuzzy clustering algorithm that suits categorical data better. More specifically, fuzzy algorithms can more appropriately describe the object-cluster similarity based on the statistics during clustering, and can thus better exploit the multi-granular information in the embeddings provided by MCDC.

C. Significance Test

To provide statistical evidence of the superiority of MCDC, we conduct a significance test using the Wilcoxon signed-rank test based on the clustering performance shown in Table III, and demonstrate the test results in Table IV. The best-performing version of MCDC, i.e., MCDC+F., is compared with each of the counterparts at a 90% confidence interval. We use "+" to indicate a rejection of the null hypothesis, which means MCDC+F. significantly outperforms the corresponding counterpart w.r.t. certain validity index.

It can be seen from Table IV that MCDC significantly outperforms its counterparts under almost all the indices, which obviously illustrates the superiority of MCDC. Although the test does not show a significant advantage of MCDC in comparison with K-MODES and ROCK in terms of AMI, MCDC+F. still outperforms them on most data sets as shown in Table III.

D. Ablation Study

To further verify the effectiveness of different main components of the proposed MCDC method, we ablate it into the following four versions: 1) MCDC[4] is the version that replaces the feature weighting mechanism (described by Eqs. (21)-(22) in Section III-B) of CAME with fixed identical weights, 2)

TABLE III: Clustering performance w.r.t., ACC, ARI, AMI, and FM on categorical data sets. MCDC+G. and MCDC+F. are the variants of MCDC adopting GUDMM and FKMAWCW, respectively. The best and second-best results on each data set are highlighted using **boldface** and underline, respectively.

Index	Data	K-MODES	ROCK	WOCIL	FKMAWCW	GUDMM	ADC	MCDC	MCDC+G.	MCDC+F.
ACC	Car.	0.372±0.00	0.326±0.00	0.270±0.00	0.371±0.00	0.372±0.00	0.361±0.00	0.373±0.00	0.270±0.00	**0.414±0.00**
	Con.	0.866±0.00	0.506±0.00	**0.874±0.00**	0.796±0.01	0.818±0.00	**0.874±0.00**	**0.874±0.00**	**0.874±0.00**	**0.874±0.00**
	Che.	0.551±0.00	0.505±0.00	0.531±0.00	0.561±0.00	0.554±0.00	0.548±0.00	0.578±0.00	0.547±0.00	**0.585±0.00**
	Mus.	0.740±0.02	0.509±0.00	0.678±0.00	0.000±0.00	0.501±0.00	0.752±0.02	0.710±0.00	0.613±0.00	**0.784±0.00**
	Tic.	0.557±0.00	**0.674±0.00**	0.526±0.00	0.538±0.00	0.507±0.00	0.535±0.00	0.602±0.00	0.642±0.00	0.646±0.00
	Vot.	0.869±0.00	0.500±0.00	0.888±0.00	0.778±0.01	0.828±0.00	0.888±0.00	**0.905±0.00**	**0.905±0.00**	**0.905±0.00**
	Bal.	0.448±0.00	0.496±0.00	0.419±0.00	0.463±0.00	0.000±0.00	0.442±0.00	0.464±0.00	0.453±0.00	**0.506±0.00**
	Nur.	0.332±0.00	0.000±0.00	0.239±0.00	0.315±0.00	0.000±0.00	0.337±0.00	0.340±0.00	0.305±0.00	**0.432±0.00**
ARI	Car.	0.027±0.00	0.023±0.00	0.001±0.00	-0.002±0.00	**0.054±0.00**	0.017±0.00	0.051±0.00	0.001±0.00	0.027±0.00
	Con.	0.536±0.00	-0.004±0.00	**0.557±0.00**	0.385±0.05	0.394±0.00	**0.557±0.00**	**0.557±0.00**	**0.557±0.00**	**0.557±0.00**
	Che.	0.014±0.00	-0.001±0.00	0.003±0.00	0.020±0.00	0.012±0.00	0.015±0.00	0.024±0.00	0.008±0.00	**0.028±0.00**
	Mus.	0.303±0.07	-0.001±0.00	0.125±0.00	0.000±0.00	-0.003±0.00	0.321±0.06	0.186±0.01	0.051±0.00	**0.323±0.00**
	Tic.	0.017±0.00	**0.120±0.00**	0.000±0.00	-0.002±0.00	-0.001±0.00	0.007±0.00	0.038±0.00	0.079±0.00	0.062±0.00
	Vot.	0.543±0.00	-0.004±0.00	0.600±0.00	0.349±0.05	0.427±0.00	0.600±0.00	**0.655±0.00**	**0.655±0.00**	**0.655±0.00**
	Bal.	0.027±0.00	**0.080±0.00**	0.005±0.00	0.055±0.00	0.000±0.00	0.025±0.00	0.052±0.00	0.016±0.00	0.079±0.00
	Nur.	0.049±0.00	0.000±0.00	0.002±0.00	0.028±0.00	0.000±0.00	0.052±0.00	0.051±0.00	0.004±0.00	**0.166±0.00**
AMI	Car.	0.049±0.00	0.050±0.00	0.003±0.00	0.082±0.00	0.117±0.00	0.047±0.00	**0.123±0.00**	0.003±0.00	0.015±0.00
	Con.	0.473±0.00	0.001±0.00	**0.484±0.00**	0.337±0.03	0.380±0.00	**0.484±0.00**	**0.484±0.00**	**0.484±0.00**	**0.484±0.00**
	Che.	0.012±0.00	0.000±0.00	0.003±0.00	**0.021±0.00**	0.011±0.00	0.015±0.00	0.020±0.00	0.005±0.00	0.020±0.00
	Mus.	0.280±0.05	0.000±0.00	0.235±0.00	0.000±0.00	0.044±0.00	**0.347±0.04**	0.209±0.01	0.036±0.00	0.248±0.00
	Tic.	0.012±0.00	**0.120±0.00**	0.007±0.00	0.005±0.00	0.000±0.00	0.006±0.00	0.020±0.00	0.058±0.00	0.023±0.00
	Vot.	0.457±0.00	0.000±0.00	0.522±0.00	0.301±0.03	0.417±0.00	0.522±0.00	**0.566±0.00**	**0.566±0.00**	**0.566±0.00**
	Bal.	0.026±0.00	0.071±0.00	0.008±0.00	0.048±0.00	0.000±0.00	0.026±0.00	0.083±0.00	0.017±0.00	**0.089±0.00**
	Nur.	0.060±0.00	0.000±0.00	0.004±0.00	0.043±0.00	0.000±0.00	0.061±0.00	0.077±0.00	0.022±0.00	**0.208±0.00**
FM	Car.	0.409±0.00	0.394±0.00	0.369±0.00	0.406±0.00	0.413±0.00	0.401±0.00	0.407±0.00	0.369±0.00	**0.434±0.00**
	Con.	0.774±0.00	0.518±0.00	**0.784±0.00**	0.711±0.01	0.754±0.00	**0.784±0.00**	**0.784±0.00**	**0.784±0.00**	**0.784±0.00**
	Che.	0.544±0.00	0.525±0.00	0.507±0.00	**0.578±0.00**	0.554±0.00	0.555±0.00	0.573±0.00	0.519±0.00	0.532±0.00
	Mus.	0.667±0.02	0.525±0.00	0.657±0.00	0.000±0.00	0.687±0.00	**0.721±0.01**	0.640±0.00	0.544±0.00	0.662±0.00
	Tic.	0.538±0.00	0.581±0.00	0.527±0.00	0.547±0.00	0.524±0.00	0.526±0.00	0.548±0.00	0.562±0.00	**0.612±0.00**
	Vot.	0.772±0.00	0.500±0.00	0.800±0.00	0.696±0.01	0.734±0.00	0.800±0.00	**0.827±0.00**	**0.827±0.00**	**0.827±0.00**
	Bal.	0.426±0.00	0.441±0.00	0.437±0.00	0.424±0.00	0.000±0.00	0.425±0.00	**0.464±0.00**	0.460±0.00	0.452±0.00
	Nur.	0.303±0.00	0.000±0.00	0.260±0.00	0.306±0.00	0.000±0.00	0.305±0.00	0.309±0.00	0.321±0.00	**0.396±0.00**

TABLE IV: Results of two-tailed Wilcoxon signed-rank test conducted with confidence interval 90% (i.e. α = 0.1). The symbol "+" indicates that MCDC+F. performs significantly better than a certain counterpart, while "-" indicates that there is no significant difference between the two methods.

Method	ACC	ARI	AMI	FM
K-MODES	+	+	-	+
ROCK	+	+	-	+
WOCIL	+	+	+	+
FKMAWCW	+	+	+	+
GUDMM	+	+	+	+
ADC	+	+	-	-

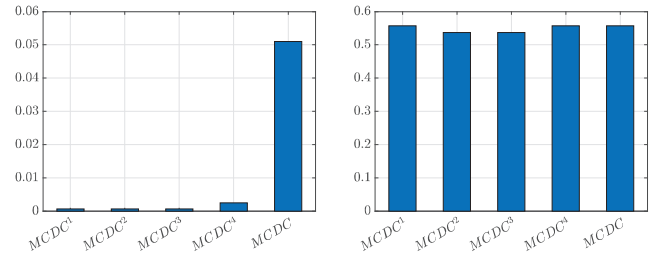
MCDC^3 is obtained by removing the whole CAME module from MCDC and use the k_σ learned by MGCPL for clustering, 3) MCDC^2 is the version obtained by replacing MGCPL of MCDC^3 with the conventional competitive learning with $k^* + 2$ as the initialization described in Section II-B, and 4) MCDC^1 is the version formed by further removing the competitive learning mechanism from MCDC^2 and only adopt the object-cluster distance described in Section II-A. Since the version of MCDC^1 that replaces object-cluster distance with the conventional Hamming distance metric is equivalent to k-modes, which has been compared in Table III, We do not further ablate MCDC^1 to avoid duplicated results.

It can be seen from the results shown in Fig. 4 that the ARI performance of MCDC, MCDC^4, MCDC^3, MCDC^2, MCDC^1 sequentially decreases in general, which intuitively illustrate the effectiveness of all the proposed main technical components of MCDC.

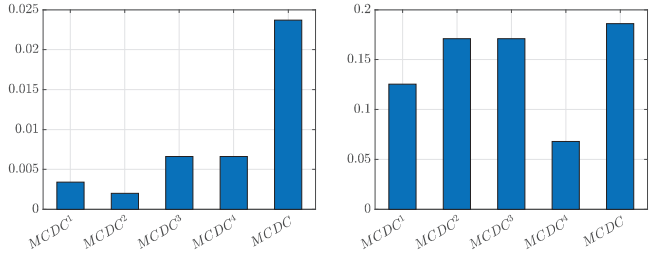
More specifically, it can be observed that MCDC always outperforms MCDC^4. This indicates that the feature weighting mechanism in CAME (i.e., Eqs. (21)-(22) in Section III-B) is effect in learning the importance of features in the embeddings output by CAME, and also illustrates that the encoding strategy of CAME is effective in fusing the multi-granular information provided by MGCPL.

It can also be observed that MCDC^4 performs not worse than MCDC^3 on five data sets, but outperformed by MCDC^3 on three data sets, i.e., Mus., Vot., and Bal. This is because identical weights of MCDC^4 cannot appropriately reflect the importance of the encoded features of these three data sets, which again highlights the necessity of weights learning.

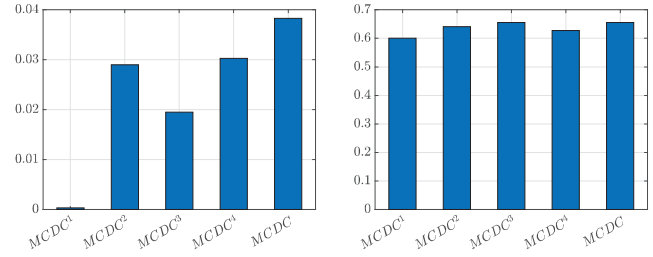
The effectiveness of the proposed MGCPL is illustrated by the fact that MCDC^3 outperforms MCDC^2 on almost all the data sets. The reason is that MGCPL learns the cluster distributions from different ks. Although the result of MCDC^3



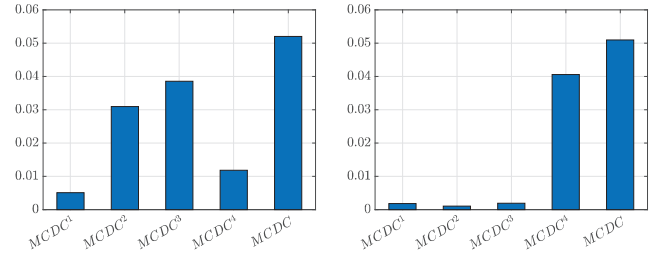
(a) ARI on Car. (b) ARI on Con.



(c) ARI on Che. (d) ARI on Mus.

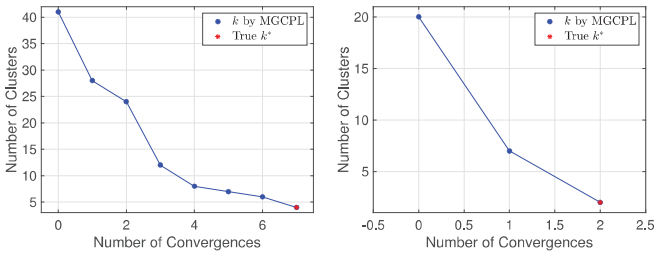


(e) ARI on Tic. (f) ARI on Vot.

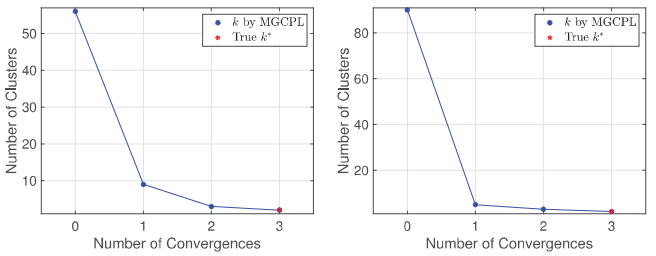


(g) ARI on Bal. (h) ARI on Nur.

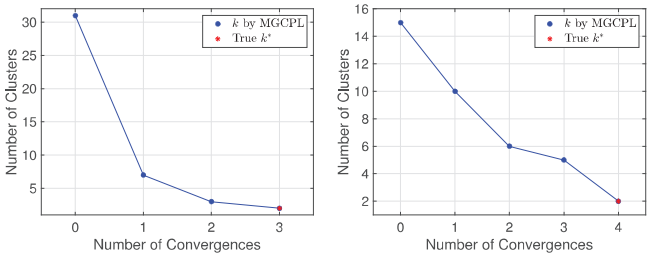
Fig. 4: Comparison of MCDC and its four ablated versions, i.e., MCDC^4, MCDC^3, MCDC^2, and MCDC^1, which are obtained by removing the weighting mechanism of CAME, the whole CAME, multi-granular learning mechanism of MGCPL, and the whole MGCPL from MCDC in turn.



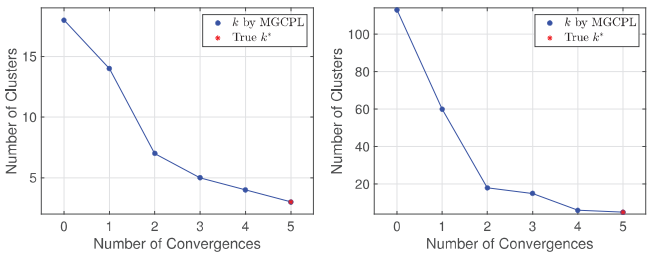
(a) ks learned for Car. (b) ks learned for Con.



(c) ks learned for Che (d) ks learned for Mus.



(e) ks learned for Tic. (f) ks learned for Vot.



(g) ks learned for Bal. (h) ks learned for Nur.

Fig. 5: Different numbers of clusters learned by MGCPL. Blue dots indicate the number of clusters when MGCPL temporarily converges under the current cluster granularity. Red stars indicate the true number of clusters k^*.

is obtained at the final k_σ, the previous $k_{\sigma-1}$ learned by MGCPL provides a more reasonable initialization for the last round learning compared to the initialized k of MCDC^2 where $k = k^* + 2$.

By comparing MCDC^2 and MCDC^1, it can be found that MCDC^2 has no significant advantage over MCDC^1. The reason is that MCDC^1 requires k^* to be given in advance for clustering, while MCDC^2 automatically learns to find k^*. In other words, although MCDC^2 adopting a competitive learning mechanism is more powerful, its advantage is obscured because k^* is unfairly leaked to MCDC^1.

E. Learning Process Evaluation

Numbers of clusters, i.e., $\kappa = \{k_1, k_2, ..., k_\sigma\}$, learned by MGCPL are demonstrated in Fig. 5 where blue dots indicate the number of clusters when MGCPL temporarily converges

under the current cluster granularity, and red stars indicate the true number of clusters k^* as shown in Table II. Please note that the number of clusters corresponding to "0" on the x-axis indicates the initialized k.

It can be observed that MGCPL converges in stages during its learning, which reflects that MGCPL can automatically learn clusters with different granularities. It can also be observed that almost all the final k_σ learned by MGCPL equal to the true number of clusters k^*, which indicates that MGCPL is competent in searching for the optimal number of clusters k^* without prior clustering knowledge.

F. Computational Efficiency Evaluation

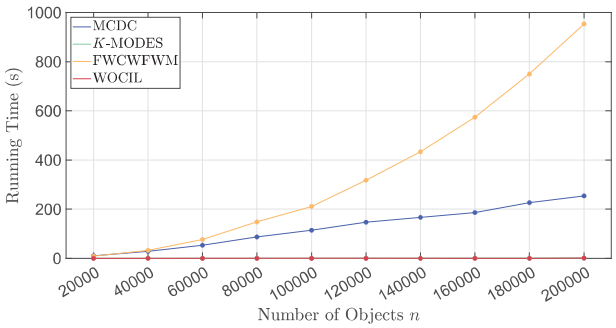
The execution time of MCDC on three synthetic data sets is shown in Fig. 6. We implement several representative counterparts on each synthetic data set with different ns, ks, and ds to verify the time complexity of MCDC. Note that k in this experiment is the number of sought clusters k in Algorithm 2. The execution time is averaged on ten runs of the corresponding method.

Intuitively, the execution time of MCDC increases linearly with the increasing of data size n and feature scale d, which confirms our analysis at the end of Section III-B that MCDC is with linear time complexity w.r.t. n and d. Moreover, it can also be observed that MCDC has linear time complexity w.r.t. k, which indicates that MCDC can be easily applied to different clustering tasks of customized k. In general, MCDC is scalable to large-scale categorical data.

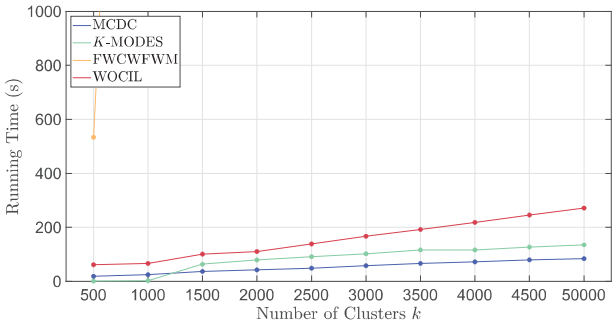
V. Concluding Remarks

This paper proposes a new method called MCDC for cluster analysis of categorical data. MCDC is composed of MGCPL for learning nested multi-granular cluster distribution, and CAME for aggregating the learned nested distribution to obtain partitional clustering results. As the learning process of MGCPL is fully automatic and highly interpretable, the complex cluster distribution of categorical data can be intuitively revealed. Accordingly, CAME encodes the multi-granular distribution learned by MGCPL to obtain informative embeddings of data objects for clustering. Since the two main components of MCDC, i.e., MGCPL and CAME, are both with linear time complexity, MCDC is scalable to large-scale categorical data. Extensive experiments illustrate the superiority of MCDC in terms of clustering accuracy, robustness to data sets in different fields, and computational efficiency.

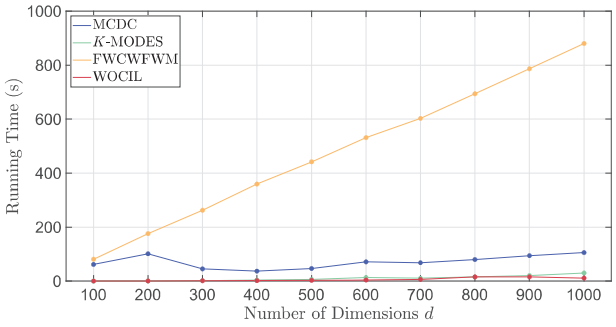
Some limitations of this work include: 1) the proposed method has not been extended to more complex heterogeneous feature data and multi-modal data and 2) we assumed that the data is static and has not yet considered more complex dynamically distributed data clustering. Building on this research, some promising future research orientations are: 1) applying MGCPL to discover implicit cluster distributions in different fields, 2) extending the whole MCDC to process streaming and dynamic data, and 3) leveraging the advantages of MGCPL to active learning for reducing the workload of human experts in manually labeling large-scale categorical data sets.



(a) Time on Syn_n w.r.t. n



(b) Time on Syn_n w.r.t. k



(c) Time on Syn_d w.r.t. d

Fig. 6: Execution time of different methods on (a) Syn_n, (b) Syn_n, and (c) Syn_d with increasing n, k, and d, respectively.

Acknowledgements

[redacted]

REFERENCES

[1] T. Li, A. Rezaeipanah, and E. M. T. El Din, "An ensemble agglomerative hierarchical clustering algorithm based on clusters clustering technique and the novel similarity measurement," *Journal of King Saud University-Computer and Information Sciences*, vol. 34, no. 6, pp. 3828–3842, 2022.

[2] D. Xu and Y. Tian, "A comprehensive survey of clustering algorithms," *Annals of Data Science*, vol. 2, pp. 165–193, 2015.

[3] J. Li, H. Izakian, W. Pedrycz, and I. Jamal, "Clustering-based anomaly detection in multivariate time series data," *Applied Soft Computing*, vol. 100, p. 106919, 2021.

[4] Z. Abbasi-Moud, H. Vahdat-Nejad, and J. Sadri, "Tourism recommendation system based on semantic clustering and sentiment analysis," *Expert Systems with Applications*, vol. 167, p. 114324, 2021.

[5] G. Caruso, S. Gattone, F. Fortuna, and T. Di Battista, "Cluster analysis for mixed data: An application to credit risk evaluation," *Socio-Economic Planning Sciences*, vol. 73, p. 100850, 2021.

[6] F. Chang, S. Yasmin, H. Huang, A. H. Chan, and M. M. Haque, "Injury severity analysis of motorcycle crashes: A comparison of latent class clustering and latent segmentation based models with unobserved heterogeneity," *Analytic Methods in Accident Research*, vol. 32, p. 100188, 2021.

[7] X.-F. Song, Y. Zhang, D.-W. Gong, and X.-Z. Gao, "A fast hybrid feature selection based on correlation-guided clustering and particle swarm optimization for high-dimensional data," *IEEE Transactions on Cybernetics*, vol. 52, no. 9, pp. 9573–9586, 2021.

[8] Y. Yang, J. Cai, H. Yang, and X. Zhao, "Density clustering with divergence distance and automatic center selection," *Information Sciences*, vol. 596, pp. 414–438, 2022.

[9] S. Chawla and A. Gionis, "K-means–: A unified approach to clustering and outlier detection," in *Proceedings of the 2013 SIAM International Conference on Data mining*. SIAM, 2013, pp. 189–197.

[10] N. Liu, Z. Xu, X.-J. Zeng, and P. Ren, "An agglomerative hierarchical clustering algorithm for linear ordinal rankings," *Information Sciences*, vol. 557, pp. 170–193, 2021.

[11] A. Agresti, *Categorical data analysis*. John Wiley & Sons, 2012, vol. 792.

[12] R. Azen and C. M. Walker, *Categorical data analysis for the behavioral and social sciences*. Routledge, 2021.

[13] A. E. Ezugwu, A. M. Ikotun, O. O. Oyelade, L. Abualigah, J. O. Agushaka, C. I. Eke, and A. A. Akinyelu, "A comprehensive survey of clustering algorithms: State-of-the-art machine learning applications, taxonomy, challenges, and future research prospects," *Engineering Applications of Artificial Intelligence*, vol. 110, p. 104743, 2022.

[14] Z. Huang, "A fast clustering algorithm to cluster very large categorical data sets in data mining," *Dmkd*, vol. 3, no. 8, pp. 34–39, 1997.

[15] R.-J. Kuo, Y. Zheng, and T. P. Q. Nguyen, "Metaheuristic-based possibilistic fuzzy k-modes algorithms for categorical data clustering," *Information Sciences*, vol. 557, pp. 1–15, 2021.

[16] F. Yuan, Y. Yang, and T. Yuan, "A dissimilarity measure for mixed nominal and ordinal attribute data in k-modes algorithm," *Applied Intelligence*, vol. 50, pp. 1498–1509, 2020.

[17] F. Murtagh and P. Contreras, "Algorithms for hierarchical clustering: an overview," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 2, no. 1, pp. 86–97, 2012.

[18] M. Alamuri, B. R. Surampudi, and A. Negi, "A survey of distance/similarity measures for categorical data," in *2014 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2014, pp. 1907–1914.

[19] Y. Qian, F. Li, J. Liang, B. Liu, and C. Dang, "Space structure and clustering of categorical data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 10, pp. 2047–2059, 2015.

[20] S. Jian, G. Pang, L. Cao, K. Lu, and H. Gao, "Cure: Flexible categorical data representation by hierarchical coupling learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, no. 5, pp. 853–866, 2018.

[21] J. Z. Huang, M. K. Ng, H. Rong, and Z. Li, "Automated variable weighting in k-means type clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 5, pp. 657–668, 2005.

[22] C. Zhu, L. Cao, and J. Yin, "Unsupervised heterogeneous coupling learning for categorical representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 1, pp. 533–549, 2020.

[23] L. Bai and J. Liang, "A categorical data clustering framework on graph representation," *Pattern Recognition*, vol. 128, p. 108694, 2022.

[24] Y. Zhang and Y.-m. Cheung, "Learnable weighting of intra-attribute distances for categorical data clustering with nominal and ordinal attributes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 7, pp. 3560–3576, 2021.

[25] Y. Zhang, Y.-m. Cheung, and A. Zeng, "Het2hom: Representation of heterogeneous attributes into homogeneous concept spaces for categorical-and-numerical-attribute data clustering," in *Proceedings of the 31st International Joint Conference on Artificial Intelligence*, 2022, pp. 1–8.

[26] P. Arabie, N. D. Baier, C. F. Critchley, and M. Keynes, "Studies in classification, data analysis, and knowledge organization," 2006.

[27] D. Barbará, Y. Li, and J. Couto, "Coolcat: An entropy-based algorithm for categorical clustering," in *Proceedings of the Eleventh International Conference on Information and Knowledge Management*, 2002, pp. 582–589.

[28] T. Li, S. Ma, and M. Ogihara, "Entropy-based criterion in categorical clustering," in *Proceedings of the Twenty-first International Conference on Machine Learning*, 2004, p. 68.

[29] Y. Zhang, Y.-M. Cheung, and K. C. Tan, "A unified entropy-based distance metric for ordinal-and-nominal-attribute data clustering," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 1, pp. 39–52, 2019.

[30] E. Mousavi and M. Sehhati, "A generalized multi-aspect distance metric for mixed-type data clustering," *Pattern Recognition*, vol. 138, p. 109353, 2023.

[31] Y. Zhang and Y.-M. Cheung, "A new distance metric exploiting heterogeneous interattribute relationship for ordinal-and-nominal-attribute data clustering," *IEEE Transactions on Cybernetics*, vol. 52, no. 2, pp. 758–771, 2020.

[32] S. Q. Le and T. B. Ho, "An association-based dissimilarity measure for categorical data," *Pattern Recognition Letters*, vol. 26, no. 16, pp. 2549–2557, 2005.

[33] A. Ahmad and L. Dey, "A method to compute distance between two categorical values of same attribute in unsupervised learning for categorical data set," *Pattern Recognition Letters*, vol. 28, no. 1, pp. 110–118, 2007.

[34] D. Ienco, R. G. Pensa, and R. Meo, "From context to distance: Learning dissimilarity for categorical data clustering," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 6, no. 1, pp. 1–25, 2012.

[35] H. Jia, Y.-m. Cheung, and J. Liu, "A new distance metric for unsupervised learning of categorical data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 5, pp. 1065–1079, 2015.

[36] A. G. Oskouei, M. A. Balafar, and C. Motamed, "Fkmawcw: Categorical fuzzy k-modes clustering with automated attribute-weight and cluster-weight learning," *Chaos, Solitons & Fractals*, vol. 153, p. 111494, 2021.

[37] Y.-m. Cheung and H. Jia, "Categorical-and-numerical-attribute data clustering based on a unified similarity metric without knowing cluster number," *Pattern Recognition*, vol. 46, no. 8, pp. 2228–2238, 2013.

[38] H. Jia and Y.-M. Cheung, "Subspace clustering of categorical and numerical data with an unknown number of clusters," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 8, pp. 3308–3325, 2017.

[39] Y. Zhang and Y.-M. Cheung, "Graph-based dissimilarity measurement for cluster analysis of any-type-attributed data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 9, 2022.

[40] Y. Jeon, J. Yoo, J. Lee, and S. Yoon, "Nc-link: A new linkage method for efficient hierarchical clustering of large-scale data," *IEEE Access*, vol. 5, pp. 5594–5608, 2017.

[41] Y.-m. Cheung and Y. Zhang, "Fast and accurate hierarchical clustering based on growing multilayer topology training," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 3, pp. 876–890, 2018.

[42] A. Dogan and D. Birant, "K-centroid link: A novel hierarchical clustering linkage method," *Applied Intelligence*, pp. 1–24, 2022.

[43] S. Guha, R. Rastogi, and K. Shim, "Rock: A robust clustering algorithm for categorical attributes," *Information Systems*, vol. 25, no. 5, pp. 345–366, 2000.

[44] L. Hu, M. Jiang, Y. Liu, and Z. He, "Significance-based categorical data clustering," *ArXiv Preprint ArXiv:2211.03956*, 2022.

[45] S. C. Ahalt, A. K. Krishnamurthy, P. Chen, and D. E. Melton, "Competitive learning algorithms for vector quantization," *Neural Networks*, vol. 3, no. 3, pp. 277–290, 1990.

Clustering by Learning the Ordinal Relationships of Qualitative Attribute Values

[redacted] Yunfan Zhang[a‡], [redacted]

2024 International Joint Conference on Neural Networks (IJCNN) | 979-8-3503-5931-2/24/$31.00 ©2024 IEEE | DOI: 10.1109/IJCNN60899.2024.10650134

***Abstract*—In many real-world clustering tasks, data objects are described by both quantitative and qualitative attributes. Attributes with semantically ordered qualitative values are very common and are usually coded according to their order (i.e., consecutive integers) for clustering. However, semantic order is not always globally interdependent with a certain clustering task. An intuitive case is that level of income (attribute) is not always positively correlated with the level of mental health (label). Using mismatched order surely forms a bottleneck to clustering performance, and conversely, the unsupervised clustering process prevents understanding of "true" order. Therefore, we proposed a novel learning paradigm to tune the value order. More specifically, we adjust the intra-attribute orders, and let this process learn mutually with object clustering, thus bridging the gap between value order and clustering task. To the best of our knowledge, this is the first attempt to learn ordinal relationships among qualitative attribute values. Extensive experiments with significance tests show that our method outperforms the existing relevant clustering approaches on qualitative attribute data.**

I. Introduction

Clustering is a fundamental data analysis technique, which is commonly used in many machine learning and data mining tasks. Current diverse data acquisition pathways allow data objects to be described by both the numerical attributes with quantitative and the categorical attributes with qualitative values, where the semantically ambiguous qualitative values are often not well encoded as numerical values by experts prior to cluster analysis. This has gradually shifted clustering research from numerical method [1]–[3] to addressing problems posed by categorical attributes [4]–[6], especially how to define their distances [7], [8], in recent years.

For categorical attributes, a taxonomy further distinguishes the attributes into nominal and ordinal ones based on whether there is a semantic order of possible values, e.g., "strong-accept", "accept", "weak accept", etc., that reviewers might recommend for this submission. The ordinal values are usually simply treated as consecutive integers and thus form an identical distance "1" between each pair of adjacent values. Such distance structure is based on external semantics, and thus cannot serve different clustering tasks adequately. An intuitive survey example [9] is that the mental health classes (i.e., healthy and unhealthy) corresponding to high-, moderate-, and low-income groups are neutral between healthy and unhealthy, tend to be healthy, and tend to be unhealthy, respectively, as shown in Fig. 1. This indicates that the semantic order of attribute "income" somewhat mismatches the task of cluster analysis of mental health patterns, and the mismatch effect accumulates when all the ordinal attributes participate in clustering based on their external semantic orders. Although attributes interactions can somewhat mitigate the effect caused by inaccurate orders, clustering performance can be further enhanced by using the "true" order of the attribute values.

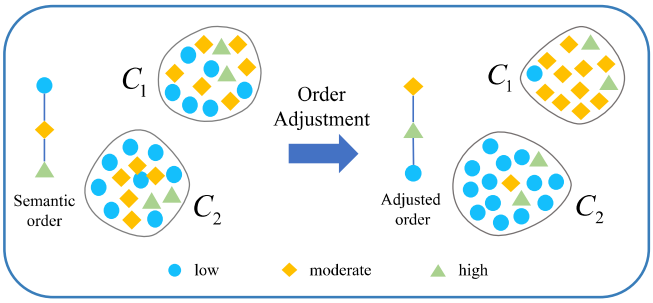


Fig. 1. An example illustrating that order adjustment matters the clustering. The C_1 and C_2 are the healthy and unhealthy clusters, respectively. After the order adjustment, the clustering result better conforms with reality.

Many advanced approaches in the literature are devoted to more comprehensively utilizing the information of categorical attributes, and have achieved considerable success in improving clustering accuracy. According to the way they exploit the information of categorical attributes in clustering, we can roughly divide them into statistical knowledge-based and learning-based approaches. Then we proceed from the perspective of how they handle ordinal attributes.

The statistical knowledge-based approaches attempt to obtain reasonable attribute representations based on the data statistics rather than the approaches that rely solely on semantic knowledge of values, e.g., conventional Hamming and order distance-based clusterings. Among this stream, entropy-based measures [10], [11] adopt statistical information entropy of the values to reflect their affiliations to the clusters. Later, more approaches [12]–[14] have been proposed on the basis of probability, adopting the common basic principle that two values with similar statistical context (e.g., occurrence frequency or conditional probability on the other attributes) should have a higher similarity. However, clustering based on all the above approaches relies fully on the data statistics but ignores

the semantic order. Therefore, some recent advances [15], [16] especially consider order information during distance measurement, thus achieving better clustering performance. Nevertheless, all the above measures work independently of the clustering, thereby limiting the clustering performance.

The learning-based approaches have thus appeared in the literature to jointly learn representations of attributes and clustering of objects. The conventional learning-based approaches either learn object-cluster similarities [17], [18] or attributes importance [19], [20] for clustering. However, they facilitate the learning based on a priori assumption on the distances among intra-attribute values, which still prevents them from approaching "true" distance representations w.r.t. clustering. Recently, a more advanced categorical data clustering approach [21] that introduces multiple kernels to comprehensively represent the attributes has been proposed. Since it does not consider ordinal attributes, the most recent approaches [22], [23] further represent and adaptively adjust the distance structures of ordinal attributes during clustering.

To the best of our knowledge, all the existing feasible solutions are based on the original semantic order, which may not suit the clustering as demonstrated by the left case in Fig. 1. Thus the original orders bottleneck the clustering performance, while the unsupervised setting conversely hinders the understanding of the "true" orders. These cross-coupled factors form the crux of the generally poor clustering performance on ordinal-attributed categorical data. Motivated by this, a method that lets the semantic order tuning and the clustering learn from each other is in urgent need.

This paper, therefore, proposes a novel method for clustering categorical data that bridges the gap between the semantic orders and the orders preferred by the clustering task. The key innovation is that we simultaneously remove the restrictions brought by the macro semantic order and the micro attribute value-level distances to the clustering through one learning paradigm. That is, we let the three objectives, i.e., (1) orders of values, (2) distances of values, and (3) partitions of objects, iteratively learn from each other through the proposed optimization algorithm. Three main contributions of this work are summarized below:

- A new paradigm, which is efficient, parameter-free, interpretable, and can be easily extended to most clustering methods for enhancement is proposed. It facilitates a high degree of freedom for learning representations of categorical data, thereby adequately eliminating the deterioration brought by the information ambiguity to clustering accuracy.
- We design an order understanding strategy to extract "hints for better orders" of each ordinal attribute from the current optimal clustering results. Through the strategy, a new order that better suits the current clusters can be inferred to provide a re-launch point that is closer to the global optimum in the next learning epoch.
- To efficiently fine-tune the newly learned order relationships, an inter-value distance learning mechanism is elegantly incorporated into the learning paradigm by concisely representing the v^2-scale distances of each attribute using v-scale weights between adjacent values (v for the number of possible values of an attribute).

II. Related Work

This section overviews existing statistical knowledge-based and learning-based categorical attribute representations for clustering, as these two topics are highly related to our work.

A. Statistical Knowledge-based Representation

Entropy-based measures [10], [11] quantify the object-cluster affiliations by adopting information entropy as a measure. [15] further preserves the order relationships of ordinal attributes by successively computing the entropy on each pair of adjacent possible values. To exploit the inter-dependence among attributes, approaches [12], [13], [24] measure the distance between two values as the differences between their corresponding conditional probability distributions reflected by the other attributes. Later, the measure [14] further selects a set of highly related attributes as context for more reasonable distance measurement. To cope with independent attributes, the recent work [16] proposes to simultaneously consider intra- and inter-attribute information, which achieves sufficient clustering performance improvement. Most recently, [25] uses entropy-based similarity and hamming distance-based metric to measure the numerical and categorical attributes.

However, all the above approaches represent object-cluster similarities independently of clustering, thus limiting the clustering performance. To address this issue, learning-based representations have been proposed in the literature.

B. Learning-based Representation

Conventional approaches [19], [26] learn the data representation from the perspective of attributes, i.e., update the weight of each attribute during clustering to obtain a better representation where the clusters appear in a more compact way. The two flexible subspace learning approaches [20], [27] learn the weights of an attribute w.r.t. different clusters. There are also approaches proposed from the perspective of objects [17], which learns object-cluster similarity based on the occurrence probability of object values in different clusters during clustering. The work [18] further considers the importance of different attributes, and achieves a better clustering performance, and [28] proposed an innovative loss function based on consensus clustering.

To more finely learn the value-level representations, [21] first represents each attribute using multiple kernels and then jointly learns the representations with clustering, which achieves very competitive clustering performance. The works [22], [29] have also been proposed to learn the inter-value distances of categorical attributes. Most recently, [23] proposes to first convert heterogeneous nominal and ordinal attributes into a homogeneous form, and then learn the inter-value distances in a decent way. However, all the learning-based approaches are still based on the semantic relationship among the possible values, which prevents them from achieving more satisfactory clustering accuracy.

III. Proposed Method

This section first introduces basic problem settings, and then presents how to infer the orders of attribute values according to a given partition of objects. Finally, the inference strategy is combined with the clustering task to form a learning paradigm.

A. Preliminaries

Given a categorical dataset denoted as $X = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n\}$ with data volume n, each of the n data objects is described by the values from d attributes $A = \{a_1, a_2, ..., a_d\}$. A data object $\mathbf{x}_i$ can be denoted in the form of a vector as $\mathbf{x}_i = [x_{i,1}, x_{i,2}, ..., x_{i,d}]^\top$. An attribute a_r describes a value domain $V_r = \{o_{r,1}, o_{r,2}, ..., o_{r,v_r}\}$ with v_r possible values. If a_r is an ordinal attribute, its values are with extra semantic order (also called rank hereinafter) in comparison with nominal attributes, and the order can be written as $\phi(o_{r,1}) > \phi(o_{r,2}) > ... > \phi(o_{r,v_r})$, where the function $\phi(\cdot)$ fetches the rank of a possible value by

$$\phi(o_{r,i}) = i. \tag{1}$$

Since a nominal attribute can be encoded into boolean-valued attributes by one-hot encoding, and the encoded attribute can be treated as ordinal attributes, we discuss by assuming all the attributes are ordinal hereinafter.

Partitional clustering is to divide the data object set X into k non-overlapping subsets called clusters $C = \{C^1, C^2, ..., C^k\}$. A cluster C^l can be represented by a d-dimensional vector $\mathbf{c}^l = [c_1^l, c_2^l, ..., c_d^l]^\top$ with values from the value domains corresponding to the d attributes. The general goal of clustering is to minimize the overall difference among intra-cluster data objects. The object-cluster affiliation is reflected by an $n \times k$ partition matrix $\mathbf{Q}$ with its the (i,j)th entry indicating if the ith object belongs to the jth cluster by

$$q_{i,j} = \begin{cases} 1, & \text{if } j = \arg\min\limits_{1 \le h \le k} \Gamma(\mathbf{x}_i, C^h) \\ 0, & \text{otherwise} \end{cases}, \tag{2}$$

where $\Gamma(\mathbf{x}_i, C^j)$ is the dissimilarity between x_i and cluster C^j, which can be generally written as

$$\Gamma(\mathbf{x}_i, C^j) = \sum_{r=1}^{d} \gamma(x_{i,r}, c_r^j) \tag{3}$$

with $\gamma(x_{i,r}, c_r^j)$ being the distance between $\mathbf{x}_i$ and C^j reflected by attribute a_r. Accordingly, the objective function can be written as

$$E(\mathbf{Q}) = \sum_{i=1}^{n} \sum_{j=1}^{k} q_{i,j} \cdot \Gamma(\mathbf{x}_i, C^j) \quad \text{s.t.} \quad \sum_{j=1}^{k} q_{i,j} = 1. \tag{4}$$

For categorical data, distance $\Gamma(\mathbf{x}_i, C^j)$ is typically defined based on the fixed semantic order of ordinal attribute values, which bottlenecks the clustering performance. To remove such restriction, we denote the orders of each attribute as $\Phi = \{\Phi_1, \Phi_2, ..., \Phi_d\}$ where $\Phi_r = \{\phi(o_{r,1}), \phi(o_{r,2}), ..., \phi(o_{r,v_r})\}$ stores the ranking of possible values for a_r, and design a learning mechanism to make the rank values in Φ learnable with clustering tasks in the following subsections.

Algorithm 1 Order Updating of Possible Values

Input: X, $\mathbf{Q}$.
Output: Φ.

```
for r = 1, 2, ..., d do
    Obtain G^r by Eq. (5) and Eq. (6);
    set D_r = ∅;
    repeat
        Find i* and j* according to Eq. (7);
        D_r = D_r ∪ {o_{r,i*}, o_{r,j*}}, update Φ_r by inserting o_{r,i*}
        and o_{r,j*} between the last found pair;
    until V_r \ D_r = ∅;
end for
```

B. Order Inference

Our proposed approach aims to represent categorical data, thus providing more appropriate distances $\Gamma(\mathbf{x}_i, C^j)$ and $\gamma(x_{i,r}, c_r^j)$ for clustering. As discussed in Section I, orders between values of an ordinal attribute dominate the distance measurement, and thus our goal is to obtain the reasonable order of the possible values to better match the clustering task. Therefore, this subsection presents how to determine the current optimal order Φ^* based on a given partition of objects.

From the perspective of the given clusters C, two possible values are more similar if they co-occur more frequently in the same cluster, and vice versa. Accordingly, dissimilarity between a pair of possible values $o_{r,i}$ and $o_{r,j}$ is defined as

$$g_{i,j}^r = \sum_{l=1}^{k} \frac{|p_{r,i}^l - p_{r,j}^l| \cdot \sigma(C^l)}{n}, \tag{5}$$

which is specified as the (i,j)-th entry of a gap matrix $\mathbf{G}^r$ corresponding to a_r. $|p_{r,i}^l - p_{r,j}^l|$ is the difference between the occurrence probabilities of $o_{r,i}$ and $o_{r,j}$ in cluster C^l with $p_{r,i}^l$ defined as

$$p_{r,i}^l = \frac{\sigma(X_{r,i}^l)}{\sigma(C^l)}, \tag{6}$$

where $X_{r,i}^l = \{\mathbf{x}_h | x_{h,r} = o_{r,i}, \mathbf{x}_h \in C^l\}$, and $\sigma(\cdot)$ counts the cardinality of a set.

To determine the orders of the possible values of an ordinal attribute a_r, we first determine the two values $o_{r,i^*}, o_{r,j^*} \in V_r$ with the current largest difference, which satisfy

$$\begin{aligned} i^*, j^* &= \arg\max_{i,j} g_{i,j}^r \\ \text{s.t.} \quad & V_r \backslash D_r \neq \emptyset \text{ and } i, j \in \{1, 2, ..., v_r\}, \end{aligned} \tag{7}$$

where D_r stores the possible values that have been ordered. By putting the two possible values $o_{r,i}^*$ and $o_{r,j}^*$ at the two sides of the obtained order, and update D_r by $D_r = D_r \cup \{o_{r,i^*}, o_{r,j^*}\}$. We then consider the rest $v_r - 2$ possible values in $V_r \setminus D_r$ according to Eq. (7), and insert the current most different pairs into the order with updating D_r until $V_r \setminus D_r = \emptyset$ or there is only one possible value in $V_r \setminus D_r$ due to the odd number of possible values in V_r. For the latter case, the only one possible value is directly inserted between the last found pair of possible values. After determining the new orders of all

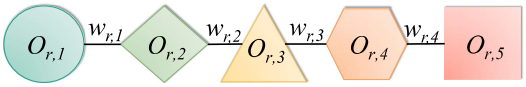


Fig. 2. Distance structure of an attribute represented by inter-value weights. In this toy example, we have the number of possible values $v_r = 5$.

the possible values of each attribute, we obtain Φ, and the above process is summarized in **Algorithm 1**. To learn Φ from data partitions, we treat it as a variable to participate in the optimization of the clustering objective in **Algorithm 2**, which will be discussed in the following subsections.

C. Distance Learning

Although we obtain new order Φ, it is still insufficient to fully utilize the information of the current object partition, and the distance structures will also change depending on Φ during clustering. Therefore, this subsection proposes an approach for learning the distance among the ordered values, and then we discuss how to learn both distances and orders during clustering in the next subsection.

To maintain an appropriate distance structure, a distance learning mechanism is derived based on a given order Φ and partition C. We introduce variables $\mathbf{W} = \{\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_d\}$ to describe the distance structures of each attribute. More specifically, each $\mathbf{w}_r = [w_{r,1}, w_{r,2}, ..., w_{r,v_r-1}]^\top$ is a (v_r-1)-dimensional vector representing the weights of inter-value distances of a_r as shown in Fig. 2. By considering $\mathbf{W}$ and Φ, the new distance Γ_{w} can be written as

$$\Gamma_{\mathrm{w}}(\mathbf{x}_i, C^l; \Phi) = \sum_{r=1}^{d} \sum_{m=1}^{v_r} \gamma_{\mathrm{w}}(x_{i,r}, o_{r,m}; \Phi_r) \cdot u^l_{r,m}, \tag{8}$$

which is the distance between $\mathbf{x}_i$ and C^l. Here, $u^l_{r,m}$ is the weight of the possible value $o_{r,m}$ to cluster C^l, which is defined as

$$u^l_{r,m} = \frac{\sigma(X^l_{r,m})}{\sigma(C^l)} \tag{9}$$

where $X^l_{r,m} = \{\mathbf{x_j} | x_{j,r} = o_{r,m}, \mathbf{x_j} \in C^l\}$ is a set of objects in C^l with their r-th values equal to $o_{r,m}$, which ranks m-th in Φ_r. Please note that once new Φ is obtained, subscripts of all the intra-attribute possible values in a value domain V_r are changed accordingly to reflect new orders in the corresponding Φ_r. In Eq. (8), $\gamma_{\mathrm{w}}(x_{i,r}, o_{r,m}; \Phi_r)$ is the distance between $x_{i,r}$ and $o_{r,m}$ according to the corresponding distance structure as shown in Fig. 2, and thus $\gamma_{\mathrm{w}}(x_{i,r}, o_{r,m}; \Phi_r)$ can be defined as

$$\gamma_{\mathrm{w}}(x_{i,r}, o_{r,m}; \Phi_r) = \begin{cases} \sum_{h=\min(\phi(x_{i,r}),m)}^{\max(\phi(x_{i,r}),m)-1} w_{r,h}, & \text{if } \phi(x_{i,r}) \neq m \\ 0, & \text{otherwise.} \end{cases} \tag{10}$$

For example, if $x_{i,r} = o_{r,2}$, i.e., $\phi(x_{i,r}) = 2$, then $\gamma_{\mathrm{w}}(x_{i,r}, o_{r,4}; \Phi_r)$ computes the distance between $o_{r,2}$ and $o_{r,4}$, which equals to $w_{r,2} + w_{r,3}$ according to the distance structure shown in Fig. 2.

As the whole distance structure can be described by $\mathbf{W}$, we update it according to the given data objects partition C by

$$w^{(\text{new})}_{r,m} = w^{(\text{old})}_{r,m} - \eta \cdot D_{r,m} \tag{11}$$

where η is a small learning rate, and $D_{r,m}$ is the overall intra-cluster distance contributed by $w_{r,m}$, which is computed by

$$D_{r,m} = w^{(\text{old})}_{r,m} \cdot \sum_{j=1}^{k} (u^j_{r,m} + u^j_{r,m+1}) \tag{12}$$

where the value of $(u^j_{r,m} + u^j_{r,m+1})$ reflects the probability that the distance described by $w_{r,m}$ being accumulated through Eq. (10) in computing the total intra-cluster distance of C^j, i.e., total difference among intra-cluster objects of C^j. It is intuitive that the overall probability $\sum_{j=1}^{k}(u^j_{r,m} + u^j_{r,m+1})$ multiplied by the distance $w_{r,m}$ (i.e., the distance between $o_{r,m}$ and $o_{r,m+1}$) in Eq. (12) computes the expectation of intra-cluster distance caused by $w_{r,m}$ in all the clusters. Therefore, $w_{r,m}$ with a larger contribution to the overall intra-cluster distance should be punished with greater force in Eq. (11) to ensure a steeper minimization of cluster objective, i.e., making the overall intra-cluster distance as small as possible. Moreover, after all the new weights of an attribute a_r are computed by Eq. (11), soft-max is adopted to normalize the updated weights by

$$w_{r,m} = \frac{w^{(\text{new})}_{r,m}}{\sum_{h=1}^{v_r-1} w^{(\text{new})}_{r,h}}. \tag{13}$$

The updating of $w_{r,m}$ can also be completed by using a more rigorously derived strategy in [22]. With the above weights updating strategies, we then discuss how to iteratively learn the weights with clustering.

D. Clustering Algorithm with Distance and Order Learning

The previous subsections illustrates how to reconstruct the distance structures of attributes (i.e., Φ and $\mathbf{W}$) given partition C. Our ultimate goal is to combine the order inference and distance learning processes with the clustering task to facilitate joint optimization. By combining the order adjustment in Section III-B and distance updating in Section III-C, the objective function in Eq. (4) can be rewritten as

$$E(\mathbf{Q}, \mathbf{W}, \Phi) = \sum_{i=1}^{n} \sum_{j=1}^{k} q_{i,j} \cdot \Gamma_{\mathrm{w}}(\mathbf{x}_i, C^j; \Phi). \tag{14}$$

The key process for minimizing Eq. (14) can be summarized into the following five steps: (1) Fix Φ and $\mathbf{W}$, compute $\mathbf{Q}$; (2) Fix Φ and $\mathbf{Q}$, update $\mathbf{W}$; (3) Iteratively implement (1) and (2) until $\mathbf{Q}$ remain unchanged; (4) Fix $\mathbf{Q}$ and $\mathbf{W}$, update Φ; (5) Iteratively implement (1)-(4) until Φ remain unchanged.

However, since the updating of Φ may result in new distance structures that prevent the learning from convergence, we replace Eq. (5) by

$$g^r_{i,j} = \sum_{l=1}^{k} \frac{|p^l_{r,i} - p^l_{r,j}| \cdot \sigma(C^l)}{n} \cdot |i-j|^\tau \tag{15}$$

Algorithm 2 CLORD: Clustering by Learning ORders and Distances

Input: X, k, η.
Output: $\mathbf{Q}$, $\mathbf{W}$, Φ.

Set $\Phi^{\tau} = \Phi$, $\tau = 0$, $Converge(\Phi) = False$;
while $Converge(\Phi) = False$ **do**
  Set $t = 0$, $Converge(\mathbf{Q}, \mathbf{W}) = False$; Initialize $\mathbf{W}^t$ by $w_{r,m} = \frac{1}{v_r - 1}$; Initialize $\mathbf{Q}^t$ by randomly selecting one entry from each row, and set the entry to 1;
  while $Converge(\mathbf{Q}, \mathbf{W}) = False$ **do**
    Fix Φ^{τ} and $\mathbf{W}^t$, obtain $\mathbf{Q}^{t+1}$ by Eq. (2);
    Fix Φ^{τ} and $\mathbf{Q}^{t+1}$, obtain $\mathbf{W}^{t+1}$ by Eqs. (11) and (13);
    if $\mathbf{Q}^{t+1}$=$\mathbf{Q}^t$ **then**
      $Converge(\mathbf{Q}, \mathbf{W}) = True$;
    end if
    $t = t + 1$;
  end while
  Fix $\mathbf{Q}^t$ and $\mathbf{W}^t$, obtain $\Phi^{\tau+1}$ by Algorithm 1;
  if $\Phi^{\tau+1}$=Φ^{τ} **then**
    $Converge(\Phi) = True$;
  end if
  $\tau = \tau + 1$;
end while

where τ is the number of learning iterations. The term $|i-j|^{\tau}$ gradually consolidate the previously learned orders, and thus weakens the instability caused updating of Φ.

The overall learning algorithm is summarized in **Algorithm 2**, and the complexity of the proposed method is provided in Theorem 1.

Theorem 1. *The overall time complexity of CLORD is* $O(EIndk)$ *at every iteration* τ.

Proof. Assuming there are n objects and d attributes in a dataset, the time complexity for obtaining $\Phi^{\tau+1}$ is $O(EndV)$, where E is the number of iterations for updating Φ, and $V = \max(v_1, v_2, ..., v_d)$ is adopted to simplify the analysis, as attributes may have different numbers of categories. Assuming I is the number of iterations to update $\mathbf{Q}^{t+1}$ and $\mathbf{W}^{t+1}$ when Φ is fixed, time complexity for obtaining $\mathbf{Q}^{t+1}$ is $O(EIndkV)$, and for obtaining $\mathbf{W}^{t+1}$ is $O(EIdkV^2)$, so the overall time complexity of CLORD is $O(EndV + EIndkV + EIdkV^2)$. Since $V << n$ and $V^2 < n$, the time complexity of CLORD can be simplified to $O(EIndk)$, which is similar to the state-of-the-art clustering methods, e.g., HD [29] and H2H [23]. □

IV. Experiments

The proposed CLORD is evaluated by comparing it with another 13 clustering methods on 10 real benchmark datasets. We introduce the experimental setup and then present the results of the designed experiments with observation analysis.

A. Experimental Setup

Comparative results are conducted from four perspectives: (1) compare CLORD with six existing methods with a significance test to statistically illustrate its superiority, (2) compare with five existing methods enhanced by the proposed core order learning mechanism to verify its scalability, (3) compare CLORD with its ablated versions to show the effectiveness of its components, and (4) compare CLORD with state-of-the-art method under different k values to demonstrate the necessity of order learning and the clustering flexibility brought by it. Moreover, changes in value ranks during clustering, execution time, cluster effect visualization, etc., are also provided to support the evaluation.

TABLE I
Statistics of 10 Datasets. n, d, and k^* stand for the numbers of data objects, attributes (ordinal+nominal), and "true" clusters used for all the experiments, respectively.

Datasets (Abbreviation)	n	d	k^*
Soybean Large (SY)	266	24+11	15
Balance scale (BS)	624	2+2	3
Fertility (FT)	100	4+3	2
Photo Evaluation (PE)	66	4+4	3
Shuttle Landing (SL)	15	5+1	2
Caesarian Section (CS)	80	3+2	2
Tic-Tac-Toe (TT)	958	9+0	2
Lense (LE)	999	4+0	5
Mammographic (MM)	824	4+0	2
Congressional Voting (VT)	434	16+0	2

The compared methods include the conventional clustering, i.e., k-means (KM), k-modes (KMD), clustering with object-cluster distance learning, i.e., OCIL [17], clustering based on the state-of-the-art distance metric UDM [16], and the most recent distance learning-based clustering approaches, i.e., HD [29] and H2H [23]. The above counterparts are chosen from different principle streams to form a more convincing performance comparison, and their hyper-parameters (if any) are set following the corresponding source papers. The learning rate η of CLORD is empirically set at 0.01. Five of the above methods are enhanced by our order learning for comparison. KMD is excluded as it treats nominal and ordinal attributes identically. Three ablated versions of CLORD are also compared, which will be introduced in the subsection "Ablation Study".

Two internal validity indices, i.e., ComPactneSs (CPS) and New Condorcet Criterion (NCC) [30], that are irrelevant to external labels and adopted distance metric have been chosen for fair evaluation, as we are doing unsupervised learning with different ks and the compared methods adopt various distance metrics that are incomparable during clustering. CPS quantifies the overall intra-cluster-object dissimilarity based on the value matching degree between two objects, and thus the lower the better. NCC simultaneously measures the intra-cluster similarity and inter-cluster dissimilarity, and thus the larger the better. Bonferroni-Dunn (BD) significance test [31] with Critical Difference (CD) interval is also adopted to provide statistical evidence for the superiority of CLORD.

Ten real datasets where PE from [22] and the remainder of nine real benchmark datasets from the UCI machine learning repository [32] are sorted out in Table I. Although the pro-

TABLE II

CPS PERFORMANCE (THE LOWER THE BETTER) COMPARISON. THE GREEN AND RED VALUES IN PARENTHESES REPRESENT THE IMPROVEMENT AND WEAKENING VALUES OBTAINED BY APPLYING OUR ORDER LEARNING MECHANISM TO THE CORRESPONDING METHOD, RESPECTIVELY.

Data	**KMD**	**KM** (Δ)	**OCIL** (Δ)	**UDM** (Δ)	**HD** (Δ)	**H2H** (Δ)	**CLORD**
SY	3.756	3.765 (+0.003)	3.823 (-0.017)	3.736 (+0.037)	3.686 (+0.038)	3.741 (-0.043)	**3.544**
BS	1.485	1.464 (+0.002)	1.457 (+0.000)	1.482 (-0.004)	1.454 (-0.005)	1.457 (-0.001)	**1.446**
FT	**1.665**	1.690 (-0.003)	1.690 (-0.002)	1.710 (-0.019)	1.687 (-0.016)	1.685 (-0.022)	1.672
PE	1.299	1.312 (+0.005)	1.311 (-0.001)	1.303 (+0.003)	1.311 (+0.003)	1.317 (-0.003)	**1.283**
SL	1.089	1.335 (-0.014)	1.318 (-0.040)	1.253 (-0.084)	1.133 (-0.053)	1.108 (-0.060)	**1.048**
CS	0.686	0.713 (-0.014)	0.722 (-0.011)	0.662 (-0.005)	0.667 (-0.028)	**0.623** (-0.005)	**0.623**
TT	2.748	2.743 (-0.014)	2.744 (-0.018)	2.781 (-0.010)	2.709 (-0.005)	2.777 (-0.007)	**2.700**
LE	1.298	1.333 (-0.002)	1.331 (-0.003)	1.341 (-0.010)	1.338 (-0.008)	1.315 (+0.005)	**1.294**
MM	0.766	0.714 (0.000)	0.714 (0.000)	0.716 (0.000)	0.737 (+0.033)	0.727 (-0.007)	**0.707**
VT	2.809	2.853 (-0.033)	2.935 (-0.085)	2.846 (-0.039)	3.161 (-0.269)	2.979 (-0.109)	**2.787**
Rank	3.80	4.95	4.85	4.80	4.20	4.30	1.10

TABLE III

NCC PERFORMANCE (THE HIGHER THE BETTER) COMPARISON. THE GREEN AND RED VALUES IN PARENTHESES REPRESENT THE IMPROVEMENT AND WEAKENING VALUES OBTAINED BY APPLYING OUR ORDER LEARNING MECHANISM TO THE CORRESPONDING METHOD, RESPECTIVELY.

Data	**KMD**	**KM** (Δ)	**OCIL** (Δ)	**UDM** (Δ)	**HD** (Δ)	**H2H** (Δ)	**CLORD**
SY	104.97	104.63 (-0.04)	104.93 (-0.37)	104.59 (-0.02)	105.87 (+0.42)	102.61 (+0.16)	**107.63**
BS	93.85	100.33 (-0.10)	100.00 (+0.11)	95.74 (+1.34)	100.09 (+0.35)	98.83 (-0.24)	**100.84**
FT	3.82	3.74 (+0.00)	3.73 (+0.00)	3.68 (+0.09)	3.76 (+0.05)	3.79 (+0.05)	**3.83**
PE	1.09	1.12 (-0.01)	1.11 (+0.00)	1.11 (-0.00)	1.10 (-0.00)	1.12 (-0.00)	**1.14**
SL	0.08	0.07 (+0.00)	0.08 (+0.00)	0.08 (+0.00)	0.08 (+0.00)	0.08 (+0.00)	**0.09**
CS	1.14	1.11 (+0.02)	1.09 (+0.03)	1.17 (+0.02)	1.17 (+0.05)	**1.24**(+0.01)	**1.24**
TT	438.90	442.71 (+2.92)	442.69 (+3.28)	428.35 (+0.06)	445.12 (+1.77)	432.37 (+2.79)	**446.44**
LE	279.98	287.05 (-0.24)	287.50 (+0.36)	275.16 (+4.07)	285.23 (+1.12)	287.92 (-1.71)	**289.83**
MM	172.46	179.76 (0.00)	179.76 (0.00)	179.62 (0.00)	176.19 (-5.04)	178.54 (+1.06)	**180.57**
VT	204.29	202.70 (+0.97)	199.38 (+3.00)	203.11 (+1.44)	190.51 (+10.21)	197.58 (+4.21)	**205.04**
Rank	4.60	4.15	4.65	5.20	4.30	4.10	1.00

posed method is discussed in terms of ordinal data, we use categorical data with mixed nominal and ordinal attributes to evaluate the proposed method in a more challenging scenario. Regarding data processing, since KMD does not have ordinal processing capabilities, the dataset is directly treated as nominal; KM and OCIL encode ordinal data into numerical data before clustering; UDM, HD, and H2H are originally proposed for mixed categorical data, so we follow the original settings. As for the CLORD method, we process nominal data using the same approach as KMD.

B. Clustering Performance Evaluation

Clustering performance evaluated by NCC and CPS is demonstrated in Tables II and III, respectively, with the result(s) of the best-performing method(s) marked in **boldface** on each dataset. The last rows in the two tables report the average ranks of the methods on all the data sets. Results in the brackets are the improvements achieved by enhancing the methods using our core order learning mechanism.

It can be observed that the proposed CLORD outperforms all the compared methods except for the CPS performance on FT dataset. CLORD outperforming all the state-of-the-art methods, i.e., UDM, HD, and H2H, further indicates its superiority in clustering categorical data.

According to the results in the brackets, we can make a statistic that our ordering mechanism successively improves the performance of the compared methods that take into account the value orders in 72 (i.e., the green results) out of 100 slots (i.e., results of 5 methods on 10 datasets evaluated by 2 validity indices). This indicates that the order learning mechanism can be easily extended to interactively learn the order of attribute values with the existing clustering methods, and effectively improve their clustering performance.

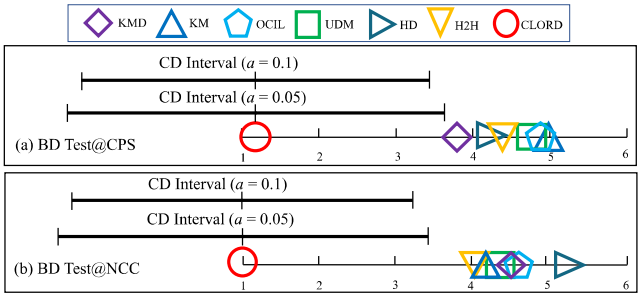


Fig. 3. Results of the two-tailed BD tests ($\alpha = 0.05$ and $\alpha = 0.1$) implemented on the "Rank" rows of Tables II and III, respectively.

C. Significance Study

To statistically illustrate the superiority of our method, we implement BD test on the "Rank" rows in Tables II and III under 95% and 90% confidence interval ($\alpha = 0.05$ and $\alpha = 0.1$). By computing the corresponding CD interval, the test results are visualized in Fig. 3. According to [31], the target method (i.e., CLORD) is believed to significantly outperform all the methods that appear out of its right-side CD interval.

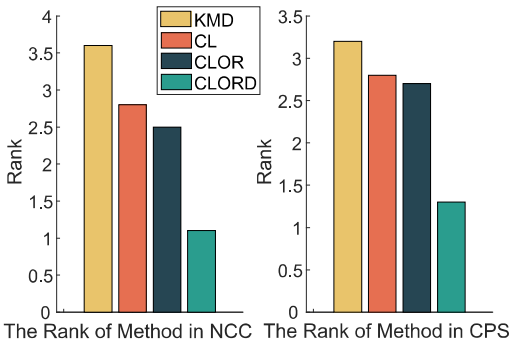


Fig. 4. Clustering performance of CLORD and its three ablated versions, i.e., CLOR, CL, and KMD.

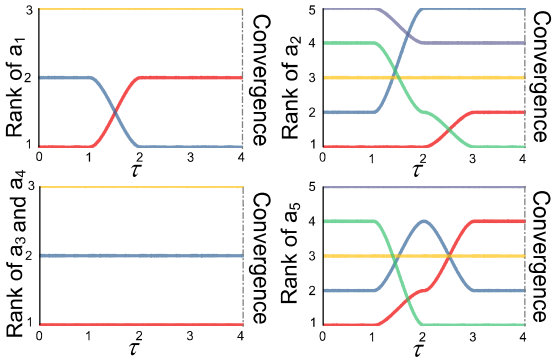


Fig. 5. Demonstration of the ordinal attribute ranks of SL dataset during the learning process of CLORD. Lines in different colors indicate the ranks of different possible values.

TABLE IV
PERFORMANCE OF H2H AND CLORD UNDER DIFFERENT kS.

Datesets		BS ($k^*=3$)		LE ($k^*=5$)	
Index	k	H2H	CLORD	H2H	CLORD
NCC	2	78.32	83.54↑	209.33	217.12↑
	3	98.83	100.84↑	253.48	258.60↑
	4	106.67	108.46↑	274.43	278.22↑
	5	112.09	112.82↑	287.92	289.83↑
	6	114.54	115.62↑	294.49	297.13↑
	7	116.92	117.81↑	300.73	302.48↑
	8	118.50	119.28↑	304.77	305.98↑
	9	119.58	120.27↑	307.18	308.97↑
	10	120.51	121.32↑	309.55	311.05↑
	11	120.88	121.79↑	311.07	312.66↑
	12	121.32	122.28↑	312.64	313.98↑

It can be seen that CLORD performs significantly better than all the counterparts in terms of both the two validity indices.

D. Ablation Study

The complete CLORD is compared with its three versions formed by successively removing the distance learning component (i.e., "D" of CLORD), the order learning component (i.e., "OR" of CLORD), and the component of distinguishing nominal and ordinal attributes. Accordingly, CLORD degrades to CLOR, CL, and KMD, respectively, where CLOR only learns the value orders without learning their distances, CL adopts the ranks to encode the ordinal attribute values into consecutive integers and adopts Hamming distance for the nominal attributes, and KMD indicates that when CL treats both ordinal and nominal attribute as nominal ones, it degrades to the conventional KMD. Performance of the above four versions of CLORD are evaluated by both CPS and NCC on all the datasets, and their average ranks are reported in Fig. 4.

It can be seen that the performance of CLORD, CLOR, CL, and KMD becomes worse in turn. The fact that CLOR performs worse than CLORD proves the rationality of distance learning; CLOR outperforming CL indicates the correctness of the learned orders; KMD performing worse than CL verifies the necessity of distinguishing ordinal and nominal attributes.

E. Performance under Different Numbers of Clusters

To illustrate the flexibility of CLORD in learning representations for different clustering tasks, we compare CLORD with the state-of-the-art H2H that does not learn the orders during clustering under different number of clusters, k, where $k = \{k', k'+1, k'+2, ..., k'+10\}$ with $k' = \max(k^*-5, 2)$. It can be seen from the results on BS and LE datasets in Table IV that CLORD always outperforms H2H. This indicates that CLORD is more flexible as it customizes both the orders and distances among attribute values for each given k. This property also makes CLORD a more promising clustering solution in real applications where k is usually subject to different analysis purposes and users.

F. Convergence Visualization

As the core of this work is the learning of orders of attribute values, we visualize the order changing of different attributes of SL datasets in Fig. 5 to provide an impression about the learning process. The horizontal and vertical axes represent the number of iterations τ and the ranks of ordinal attribute values, respectively. The dashed vertical lines indicate the convergence iteration of CLORD. We visualize the same ranks of values together (i.e., a_3, a_4 in the SL dataset), which remain unchanged during learning. It can be observed that CLORD converges very quickly, i.e., within 4 iterations on all the datasets. Moreover, the ranks of values fluctuate very little and the overall change is monotonous during the learning, which intuitively indicates the reasonableness of the proposed order learning mechanism.

V. CONCLUSION

In this paper, an intuitive but yet-to-be-concerned semantic order mismatch phenomenon that brings performance bottlenecks for categorical data clustering has been studied and elaborately addressed. To eliminate the mismatch in the clustering task without discarding relevant semantic information, we design a new learning paradigm to gradually tune the semantic order, which iteratively adjusts the previous order and learns distances among ordered values based on the current optimal

clustering result. It turns out that the proposed method is robust to different clustering tasks, i.e., either clustering different datasets or clustering the same dataset with different sought numbers of clusters k. Time complexity analysis and comprehensive experiments illustrate the promising performance of the proposed method.

ACKNOWLEDGEMENTS

REFERENCES

[1] Yiu-Ming Cheung. k-means: A new generalized k-means clustering algorithm. *Pattern Recognition Letters*, 24(15):2883–2893, 2003.

[2] Liang Bai, Xueqi Cheng, Jiye Liang, Huawei Shen, and Yike Guo. Fast density clustering strategies based on the k-means algorithm. *Pattern Recognition*, 71:375–386, 2017.

[3] Mingjie Zhao, Yiqun Zhang, Yuzhu Ji, and Yang Lu. Unsupervised concept drift detection via imbalanced cluster discriminator learning. In *Proceedings of the 6th Chinese Conference on Pattern Recognition and Computer Vision*, pages 31–43. Springer, 2023.

[4] Lang Zhao, Yiqun Zhang, Yuzhu Ji, An Zeng, Fangqing Gu, and Xiaopeng Luo. Heterogeneous drift learning: classification of mix-attribute data with concept drifts. In *Proceedings of the 9th International Conference on Data Science and Advanced Analytics*, pages 1–10. IEEE, 2022.

[5] Shenghong Cai, Yiqun Zhang, Xiaopeng Luo, Yiu-ming Cheung, Hong Jia, and Peng Liu. Robust categorical data clustering guided by multi-granular competitive learning. In *Proceedings of the 44th International Conference on Distributed Computing Systems*, pages 1–12. IEEE, 2024.

[6] Fangqi Nie, Pengcheng Yan, Yiqun Zhang, Fangqing Gu, Yang Lu, and Yue Zhang. Space2: dual space learning for categorical data clustering. In *Proceedings of the 19th International Conference on Computational Intelligence and Security*, pages 1–5. IEEE, 2023.

[7] Madhavi Alamuri, Bapi Raju Surampudi, and Atul Negi. A survey of distance/similarity measures for categorical data. In *Proceedings of the 2014 International Joint Conference on Neural Networks*, pages 1907–1914. IEEE, 2014.

[8] Yiqun Zhang and Yiu-Ming Cheung. Graph-based dissimilarity measurement for cluster analysis of any-type-attributed data. *IEEE Transactions on Neural Networks and Learning Systems*, 34(9):6530–6544, 2023.

[9] Richard G Wilkinson. Income distribution and life expectancy. *BMJ: British Medical Journal*, 304(6820):165, 1992.

[10] Dekang Lin. An information-theoretic definition of similarity. In *Proceedings of the 15th International Conference on Machine Learning*, page 296–304, San Francisco, CA, USA, 1998. Morgan Kaufmann Publishers Inc.

[11] Daniel Barbará, Yi Li, and Julia Couto. Coolcat: an entropy-based algorithm for categorical clustering. In *Proceedings of the 7th International Conference on Information and Knowledge Management*, pages 582–589, 2002.

[12] Si Quang Le and Tu Bao Ho. An association-based dissimilarity measure for categorical data. *Pattern Recognition Letters*, 26(16):2549–2557, 2005.

[13] Amir Ahmad and Lipika Dey. A method to compute distance between two categorical values of same attribute in unsupervised learning for categorical data set. *Pattern Recognition Letters*, 28(1):110–118, 2007.

[14] Dino Ienco, Ruggero G Pensa, and Rosa Meo. From context to distance: Learning dissimilarity for categorical data clustering. *ACM Transactions on Knowledge Discovery from Data*, 6(1):1–25, 2012.

[15] Yiqun Zhang, Yiu-Ming Cheung, and Kay Chen Tan. A unified entropy-based distance metric for ordinal-and-nominal-attribute data clustering. *IEEE Transactions on Neural Networks and Learning Systems*, 31(1):39–52, 2020.

[16] Yiqun Zhang and Yiu-Ming Cheung. A new distance metric exploiting heterogeneous interattribute relationship for ordinal-and-nominal-attribute data clustering. *IEEE Transactions on Cybernetics*, 52(2):758–771, 2022.

[17] Yiu-ming Cheung and Hong Jia. Categorical-and-numerical-attribute data clustering based on a unified similarity metric without knowing cluster number. *Pattern Recognition*, 46(8):2228–2238, 2013.

[18] Hong Jia, Yiu-ming Cheung, and Jiming Liu. A new distance metric for unsupervised learning of categorical data. *IEEE Transactions on Neural Networks and Learning Systems*, 27(5):1065–1079, 2015.

[19] Elaine Y Chan, Wai Ki Ching, Michael K Ng, and Joshua Z Huang. An optimization algorithm for clustering using weighted dissimilarity measures. *Pattern Recognition*, 37(5):943–952, 2004.

[20] Hong Jia and Yiu-Ming Cheung. Subspace clustering of categorical and numerical data with an unknown number of clusters. *IEEE Transactions on Neural Networks and Learning Systems*, 29(8):3308–3325, 2017.

[21] Chengzhang Zhu, Longbing Cao, and Jianping Yin. Unsupervised heterogeneous coupling learning for categorical representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(1):533–549, 2022.

[22] Yiqun Zhang and Yiu-ming Cheung. An ordinal data clustering algorithm with automated distance learning. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence*, volume 34, pages 6869–6876, 2020.

[23] Yiqun Zhang, Yiu-ming Cheung, and An Zeng. Het2hom: representation of heterogeneous attributes into homogeneous concept spaces for categorical-and-numerical-attribute data clustering. In *Proceedings of the 31st International Joint Conference on Artificial Intelligence*, pages 1–8, 2022.

[24] Lang Zhao, Yiqun Zhang, Xiaopeng Luo, Yue Zhang, Yiu-Ming Cheung, and Kangshun Li. Selecting heterogeneous features based on unified density-guided neighborhood relation for complex biomedical data analysis. In *Proceedings of the 2023 IEEE International Conference on Bioinformatics and Biomedicine*, pages 771–778. IEEE, 2023.

[25] Naoki Masuyama, Yusuke Nojima, Hisao Ishibuchi, and Zongying Liu. Adaptive resonance theory-based clustering for handling mixed data. In *2022 International Joint Conference on Neural Networks*, pages 1–8. IEEE, 2022.

[26] Joshua Zhexue Huang, Michael K Ng, Hongqiang Rong, and Zichen Li. Automated variable weighting in k-means type clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(5):657–668, 2005.

[27] Lifei Chen, Shengrui Wang, Kaijun Wang, and Jianping Zhu. Soft subspace clustering of categorical data with probabilistic distance. *Pattern Recognition*, 51:322–332, 2016.

[28] Jayanth Reddy Regatti, Aniket Anand Deshmukh, Eren Manavoglu, and Urun Dogan. Consensus clustering with unsupervised representation learning. In *2021 International Joint Conference on Neural Networks*, pages 1–9. IEEE, 2021.

[29] Yiqun Zhang and Yiu-ming Cheung. Learnable weighting of intra-attribute distances for categorical data clustering with nominal and ordinal attributes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(7):3560–3576, 2022.

[30] T.R. Santos and Luis E. Zárate. Categorical data clustering: What similarity measure to recommend? *Expert Systems with Applications*, 42(3):1247–1260, 2015.

[31] Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine learning research*, 7:1–30, 2006.

[32] Dheeru Dua, Casey Graff, et al. Uci machine learning repository. 2017.

Towards Unbiased Minimal Cluster Analysis of Categorical-and-Numerical Attribute Data

Yunfan Zhang[1] [redacted]

Abstract. Categorical and numerical attributes occur frequently in cluster analysis tasks. To bridge the information gap between the heterogeneous categorical and numerical attributes in cluster analysis, the existing approaches usually adopt prior assumptions to distance definition and cluster distribution, which unavoidably introduce bias to the clustering process. To address this issue, we propose to analyze mixed data comprising both categorical and numerical attributes by forming minimal clusters through neighborhood set theory. As the minimal clusters are the smallest cluster units that can be obtained without relying on prior assumptions, unbiased cluster analysis can be facilitated accordingly. To avoid information loss, distance and density metrics that are unified on both numerical and categorical attributes are also proposed and utilized to merge the minimal clusters hierarchically. It turns out that our proposed approach is highly interpretable, and is capable of accurately and robustly clustering data sets composed of any combination of numerical and categorical attributes. Extensive experimental evaluations demonstrate its efficacy.

Keywords: Cluster analysis · Categorical attribute · Neighborhood rough set · Mixed data · Unsupervised learning.

1 Introduction

Cluster analysis is a common data analytic technique to identify cluster patterns from data sets. In real clustering tasks, numerical attributes with quantitative values and categorical attributes [1] with qualitative values are very common, where we call the data set composed of both numerical and categorical attributes mixed data. However, as shown in Fig. 1, the distance space of mixed data cannot be well-defined like the Euclidean distance due to the qualitative categorical data values. Additionally, the possible values of categorical attributes are usually

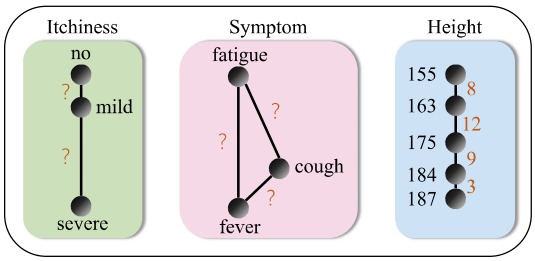


Fig. 1: Numerical attributes such as 'Height' can be effectively represented in Euclidean space, while quantifying dissimilarity between possible values within categorical attributes like 'Itchiness' and 'Symptoms' poses great challenges as the categorical values cannot directly participate in arithmetic operations.

divergent concepts in different domains with distinct implicit distance structures, which brings great challenges to the cluster analysis of mixed data.

Most existing attempts for mixed data clustering focus on the distance defining across heterogeneous attributes, and can be roughly divided into the following two streams: (i) k-ProtoType (KPT)-type methods: directly weight and combine different dissimilarity measures, e.g., Euclidean and Hamming metrics during clustering, and (ii) dedicated metric-based methods, which usually define a metric unified on the numerical and categorical attributes for distance measurement during clustering.

For KPT-type methods, the conventional KPT algorithm [11] combines Euclidean and Hamming distances [3] to cluster mixed data sets. A recent variant [17] improves the metric of categorical attributes by representing categorical values via inter-value and inter-attribute couplings, thus encoding relationships for better distance measurement. Context-based metric [13] considers attribute interdependence to form an informative categorical attribute metric. More advanced clustering methods like [16] measure the distance between possible values using Conditional Probability Distributions (CPDs) across attributes. However, these methods focus solely on proposing more advanced categorical attribute metrics and combine them with Euclidean distance for mixed data clustering, neglecting the heterogeneity of categorical and numerical attributes.

For metric-based methods, the work proposed in [7] quantifies inter-object-cluster similarity for numerical and categorical attributes within a unified probability framework, while an entropy-based approach [23] further considers the value order in categorical attributes and measures the dissimilarity degrees between different possible values from an information theory perspective. Nevertheless, these methods assume the independence of attributes, leading to information loss when applied to real-world data sets with interdependent attributes. Advanced distance definitions [14, 21] take into account attribute interdependence and preserve the corresponding information by reflecting distances based on more relevant attributes. However, they are not robust to various data sets as

their effectiveness highly relies on the consistency between inherent data characteristics and their assumptions, e.g., the existence of inter-value order, and inter-attribute dependence, etc.

In general, almost all the existing mixed data clustering approaches rely on certain prior knowledge or assumptions of data sets. Specifically, context-based [13] methods adopt the prior knowledge that the similarities of their possible values are reflected by the CPDs corresponding to the values obtained from other attributes, while the information theory-based methods like [23] measure dissimilarity between two possible values according to the degree of information chaos jointly demonstrated by them. Moreover, the number of true clusters is usually assumed to be known in advance. However, direct searching for oversized clusters may hinder the exploration of locally compact smaller-sized clusters. The above issues will inevitably lead to various clustering biases and thus influence clustering accuracy.

To this end, this paper proposes a universal clustering algorithm robust to various mixed data, addressing the challenges of considering heterogeneous attributes and lifting the restriction of prior knowledge. It groups data objects with distinct boundaries according to neighborhood set search, where only intra-cluster objects are expected to be included in a compact group (also called micro-cluster). Then the micro-clusters are merged to form larger "true" clusters (macro-cluster), and thus the proposed algorithm is called Mic2Mac. More specifically, a novel neighborhood relation is first proposed, forming rational and compact micro-clusters by comprehensively considering the distance and density of data objects. Subsequently, a hierarchical merging mechanism is designed to merge the current most similar micro-clusters into macro-clusters progressively. As the hierarchical merging is performed at the cluster level, the computation cost is thus not obviously increased. Extensive experiments, including comparative results, ablation studies, and visualization, affirm the superiority of Mic2Mac across various clustering methods on real benchmark data sets. The main contributions of this paper are three-fold:

1) A new clustering method is proposed based on neighborhood relationship to accurately form clusters of arbitrary shapes, tackling the cluster distribution bias of existing mixed data clustering methods.
2) An adaptive neighborhood relationship is defined based on both distance and density, leading to the generation of compact and non-overlapping micro-clusters, which has been proven to be universal and practical in the exploration of complex real-world data distributions.
3) Clustering process of Mic2Mac conforms to the inference process from deterministic micro-clusters to uncertain macro-clusters, providing interpretable cluster nesting relationships for multi-granular distribution analysis.

2 Related Work

As our proposed mixed data clustering approach is based on the data object partition technique, this section makes an overview of mixed data measures. It

focuses on mixed data clustering methods, and data object partition techniques including k-means-type partition techniques, and neighborhood rough sets.

2.1 Mixed Data Measures

Early mixed data clustering methods like k-prototypes [11], utilized one-hot encoding to transform categorical attribute values [2] into binary vectors. However, the Hamming distance has obvious limitations in discerning differences between various value pairs. Consequently, numerous advanced techniques have emerged to efficiently address the heterogeneous attribute data, including similarity-based and representation-based approaches.

For similarity-based measures, such as context-based distance measures [13], they evaluate the distance between related attribute CPDs to highlight their dissimilarity and identify attributes with weaker dependencies. Nonetheless, these methods do not fully account for the heterogeneity of the complex categorical attributes. Subsequently, the information theory-based metric [23] measures the distances for categorical attributes by incorporating attribute weighting. Most recently, a distance learning-based approach [19] has been proposed to learn the ordinal structure of the qualitative attributes and then cluster them, while AMPHM [24] is proposed to cluster mixed data based on the rough set theory.

For representation-based measures, an interpretable representation method [16] encodes original data and further performs k-means clustering and PCA for more accurate representation. However, it is designed for categorical data only. Recently, a deep learning clustering method [5] transforms both numerical and categorical attribute values into a unified space to enable more appropriate clustering. Most recently, an approach [25] constructs minimal spanning trees for possible values to tackle qualitative-attribute clustering tasks. Moreover, the competitive theory has been utilized to handle the qualitative categorical data [4] and clustering in a federated scenario [26]. Most existing methods for clustering mixed data typically have one or both following restrictions: 1) they are tailored to data sets with one specific attribute type, and 2) they often rely on prior knowledge or assumptions.

2.2 Data Object Partition Techniques

The early k-means-type approach [12] was widely used for partitioning numerical and categorical attributes data into k clusters, while it treats all categorical variables equally during the clustering process. Recently, the representative attribute weighting partition methods w-k-means [10] was proposed for reasonably selecting variables, thereby partitioning mixed data. Nevertheless, it unreasonably assigns identical distances to different pairs of adjacent categories that may have intrinsically unequal distances, thus showing unsatisfactory partition results. Most recently, The clustering approach in [6] is proposed for incomplete data, but designed for numerical data only.

Neighborhood rough set (also called neighborhood set interchangeably for simplicity) is commonly used to partition categorical or mixed data sets. Specifically, it lets each object $\mathbf{x}_i$ find a micro-cluster based on the neighborhood set, consisting of objects that are closer to $\mathbf{x}_i$. $D(\mathbf{x}_i, \mathbf{x}_j)$ represents the distance between $\mathbf{x}_i$ and $\mathbf{x}_j$. The common neighborhood relations are the k-nearest

$$M^k(\mathbf{x}_i) = \{\mathbf{x}_j | D_k(\mathbf{x}_i, \mathbf{x}_j) < D(\mathbf{x}_i, \mathbf{x}_g)\}, \tag{1}$$

and the δ-radius

$$M^\delta(\mathbf{x}_i) = \{\mathbf{x}_j | D(\mathbf{x}_i, \mathbf{x}_j) \leq \delta\}, \tag{2}$$

where $j, g \in \{1,2,...,n\}$, $g \neq j$, and the k in Eq. (1) represents the first k objects with the closest distance to $\mathbf{x}_i$. For simplification, we employ $M(\mathbf{x}_i)$ to denote the general neighborhood relation.

3 Proposed Method

In this section, we begin by formulating the problem in Section 3.1. Then, we present the micro partition based on the neighborhood set and the mixed data distance metric in Section 3.2. Finally, the hierarchical merging mechanism, and the whole clustering algorithm Mic2Mac are proposed in Section 3.3.

3.1 Problem Formulation

Given a mixed data set $S = \{\mathbf{x}_1, \mathbf{x}_2,..., \mathbf{x}_n\}$ comprising n data objects, each data object $\mathbf{x}_i = [x_i^1, x_i^2,... ,x_i^d]^\top$ is a d-dimensional vector with values from the d attributes, which can also be denoted as a set $A = \{a^1, a^2,..., a^d\}$. The possible value set $V = \{V^1, V^2,..., V^d\}$ stores the value domains corresponding to each attribute. The goal of clustering is to assign the n objects to k suitable clusters $C = \{C_1, C_2,..., C_k\}$, where C_l is the set of data objects in the l-th cluster, with $S = \bigcup_{l=1}^{k} C_l$. To represent each cluster, a representative objects set $R = \{\mathbf{r}_1, \mathbf{r}_1,..., \mathbf{r}_k\}$ is maintained during clustering, and each representative object $\mathbf{r}_l$ of R is a data object selected from S. A common way is to use an $n \times k$ matrix $\mathbf{Q}$, indicating which cluster is an object assigned to. The (i, l)-th entry $q_{i,l}$ of $\mathbf{Q}$ is denoted as

$$q_{i,l} = \begin{cases} 1, \text{ if } & l = arg\min_g D(\mathbf{x}_i, \mathbf{r}_g), \\ 0, \text{ if } & l \neq g. \end{cases} \tag{3}$$

According to Eq. (3), we have

$$\sum_{l=1}^{k} q_{i,l} = 1, \quad 1 \leq i \leq n, \tag{4}$$

and $q_{i,l} \in \{0, 1\}$. To appropriately cluster mixed data sets, we first need inter-object distances to be prepared where a common form can be

$$D(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{\sum_{a^r \in A} D^r(x_i^r, x_j^r)^2}. \tag{5}$$

In Eq. (5), $x_i^r \in V^r$ represents the value of $\mathbf{x}_i$ on a^r, while $D^r(x_i^r, x_j^r)$ quantifies the distance between $\mathbf{x}_i$ and $\mathbf{x}_j$ w.r.t. a^r. In the following subsection, we present how to define $D^r(x_i^r, x_j^r)$ to form neighborhood sets.

3.2 Micro Partition based on Neighborhood Set

To unify the distance metric on heterogeneous attributes, we use transformation cost that quantifies the effort required to transform one Conditional Probability Distribution (CPD) into another. We begin by defining the CPD and establishing the distance between possible values of a categorical attribute to explain the principles of transformation cost quantification more clearly. Subsequently, we illustrate how this approach unifies both categorical and numerical scenarios. Finally, we derive the object-level distance and propose the micro partition based on the neighborhood set. Given a possible value v_h^r from attribute a^r, the CPD of a^t with v^t possible values $V^t = \{v_1^t, v_2^t, ..., v_{\mathrm{v}^t}^t\}$ is computed accordingly

$$\Psi_h^{rt} = [p(v_1^t|v_h^r), p(v_2^t|v_h^r), ..., p(v_{\mathrm{v}^t}^t|v_h^r)]^\top, \tag{6}$$

where $p(v_o^t|v_h^r)$ is the conditional probability of v_o^t as given v_h^r. We represent the CPD as Ψ_h^{rt} where the superscript rt signifies that this CPD characterizes the h-th possible value of a^r concerning the values of a^t. The distinction between two CPDs, such as Ψ_h^{rt} and Ψ_o^{rt}, captures the dissimilarity between v_h^r and v_o^r, according to the possible values V^t.

To quantitatively measure this dissimilarity between the CPDs describing two possible values of a categorical attribute, we employ the Earth Mover's Distance (EMD) [20], which was designed to calculate the transformation costs between two histogram descriptors. Thus, the dissimilarity between two possible values v_h^r and v_o^r, reflected by a^t can be calculated using EMD by

$$D^{rt}(v_h^r, v_o^r) = \Gamma(\Psi_h^{rt} - \Psi_o^{rt}, \boldsymbol{O}) \cdot \mathbf{I}, \tag{7}$$

where $\Gamma(\cdot, \cdot)$ compares each pair of corresponding bits of two vectors and retains the maximum value, while $\boldsymbol{O}$ and $\mathbf{I}$ represent a v^t-dimensional vector with all values equal to 0 and 1, respectively.

Different attributes a^ts can have varying contributions to the distance $D^{rt}(v_h^r, v_o^r)$ due to variations in inter-attribute dependence. The overall $D^{rt}(v_h^r, v_o^r)$ reflected by its respective weight w^{rt} is computed by

$$D^r(v_h^r, v_o^r) = \sum_{a^t \in A} D^{rt}(v_h^r, v_o^r) \cdot w^{rt}. \tag{8}$$

The Eq. (7) is further extended to quantify the inter-attribute dependence as the weights w^{rt}, which can be expressed as

$$w^{rt} = \frac{\sum_{h=1}^{\mathrm{v}^r-1}\sum_{o=h+1}^{\mathrm{v}^r} D^r(v_h^r, v_o^r)}{\mathrm{v}^r(\mathrm{v}^r-1)/2}, \tag{9}$$

where v^r represents the number of possible values contained within a^r. More specifically, w^{rt} measures the average transformation cost of the $\mathrm{v}^r(\mathrm{v}^r-1)/2$ pairs of possible values of attribute a^r reflected by a^t. According to Eqs. (7)-(9), the heterogeneous attributes are uniformly quantified as the transformation cost.

According to the work proposed in [22], the possible values of a categorical attribute are considered as concepts, so that the above process essentially quantifies the average inter-concept distances of a^r as influenced by a^t. To illustrate the principle of Eq. (8), we examine an extreme scenario. Assuming attributes a^r and a^t are identical, they will exhibit perfect interdependence, and thus their $D^{rt}(v_h^r, v_o^r)$ always reaches the maximal value, i.e., "1", for any combinations of h and o with $h \neq o$, according to Eq. (7). Consequently, w^{rt} also reaches the maximal value of "1", representing 100% dependence of two attributes. By applying Eqs. (7) - (9), we can obtain the distance between data objects $\mathbf{x}_i$ and $\mathbf{x}_j$.

The defined dissimilarity measure applies to both categorical attributes and numerical attributes, as Eq. (8) provides a uniform treatment of heterogeneous attributes. Then we prove that our measure is a distance metric.

Theorem 1. *$D(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is a distance metric.*

Proof. As Eq. (7) satisfies the properties of a metric, it follows naturally Eq. (8), which is derived from Eq. (7), is also a metric. Moreover, the calculation of Eq. (5) involves finite arithmetic processes according to Eq. (8), guaranteeing that $D(\mathbf{x}_i, \mathbf{x}_j)$ adheres to all essential metric properties for any $i, j, h \in \{1, 2, ..., n\}$, which are listed as follows:
(1)$D(\mathbf{x}_i, \mathbf{x}_j) \geq 0$; $D(\mathbf{x}_i, \mathbf{x}_j) = 0$ iff $\mathbf{x}_i = \mathbf{x}_j$;
(2)$D(\mathbf{x}_i, \mathbf{x}_j) = D(\mathbf{x}_j, \mathbf{x}_i)$;
(3)$D(\mathbf{x}_i, \mathbf{x}_j) \leq D(\mathbf{x}_i, \mathbf{x}_h) + D(\mathbf{x}_h, \mathbf{x}_j)$. □

The conventional neighborhood sets $M^k(\mathbf{x}_i)$ and $M^\delta(\mathbf{x}_i)$ (i.e., Eqs. (1) and (2)) generate n neighborhood sets, which may partially overlap with surrounded ones, causing laborious computation with a large n. Additionally, these neighborhood relations may group dissimilar objects in the uneven distribution of data objects. To better partition objects and reduce computational costs, we have developed a new approach called micro partition based on the neighborhood set, which considers both distance and density. This approach creates non-overlapping neighborhood sets by selecting representative objects and grouping their corresponding neighbors based on *merging interval*.

Definition 1. *Merging interval: Given an object $\boldsymbol{x}_i$ with a density ρ_i, the merging interval ϕ_i signifies the minimum distance between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ with a higher corresponding density ρ_j, which can be expressed as:*

$$\phi_i = \min D(\mathbf{x}_i, \mathbf{x}_j), \quad s.t.\ \rho_i < \rho_j \ and\ \mathbf{x}_j \in S \backslash \mathbf{x}_i, \tag{10}$$

where $S\backslash\mathbf{x}_i$ is the data set that excludes $\mathbf{x}_i$, while ρ_i and ρ_j denote the densities of $\mathbf{x}_i$ and $\mathbf{x}_j$, respectively. Furthermore, for the object with the maximum density, its merging interval is defined as $\max D(\mathbf{x}_i, \mathbf{x}_j)$, where $\mathbf{x}_j \in S$.

In Definition 1, the density ρ_i can be computed as

$$\rho_i = \frac{D(\mathbf{x}_i, \mathbf{x}_{\langle i,q_i \rangle})}{q_i}. \tag{11}$$

Eq. (11) computes the distance corresponding to the ranking q_i of the adjacent object $\mathbf{x}_{\langle i,q_i \rangle}$, which can be seen as the density of $\mathbf{x}_i$. Assuming the $\mathbf{x}_i$ is a center point, $\mathbf{x}_{\langle i,q_i \rangle}$ is the q_i-th closest object to $\mathbf{x}_i$ in n objects. Specifically, we initially create the neighbor set $AM_i = \{\mathbf{x}_{\langle i,0 \rangle}, \mathbf{x}_{\langle i,1 \rangle}, \mathbf{x}_{\langle i,2 \rangle}, ..., \mathbf{x}_{\langle i,n-1 \rangle}\}$ in *ascending* sequence relative to $\mathbf{x}_i$, where $\mathbf{x}_{\langle i,0 \rangle} \equiv \mathbf{x}_i$ and $AM_i(y) = \mathbf{x}_{\langle i,y \rangle}$. Afterwards, when we iterate through AM_i from small to large, we choose the object $\mathbf{x}_{\langle i,g \rangle}$ that first satisfies the condition $D(\mathbf{x}_i, \mathbf{x}_{\langle i,g \rangle})/g < D(\mathbf{x}_{\langle i,g-1 \rangle})/(g-1)$, which confirms the value of q_i as $q_i = g - 2$. The density calculation effectively selects neighboring objects, ensuring that objects beyond a noticeable interval boundary are not included in the neighborhood set corresponding to $\mathbf{x}_i$. Hence, it will partition objects into compact clusters, which contain the most similar objects.

To select the most suitable representative object for a micro-cluster, we prioritize objects with higher density than their neighbors and positioning far from other representative objects. According to Definition 1, objects with greater merging intervals are considered more suitable to be the representative objects. Thus, we rank data objects based on their merging intervals in *descending* sequence and form micro-clusters based on neighborhood set by

$$M^{\phi}(\mathbf{x}_i) = \left\{ \bigcup_{j=1}^{q_i} AM_i(j) \right\} \setminus \left\{ \bigcup_{\phi_p > \phi_i} M^{\phi}(\mathbf{x}_p) \right\}, \tag{12}$$

where q_i is the q_i-th closeness to $\mathbf{x}_i$ among all the n objects, as metioned in Eq. (11), while the objects $\mathbf{x}_p$ with larger merging intervals than the $\mathbf{x}_i$ will be excluded from $M^{\phi}(\mathbf{x}_i)$ corresponding to $\mathbf{x}_i$. The process of forming micro-clusters will continue until all objects are contained by these micro-clusters. All the representative objects in each micro-cluster are stored in the micro representative objects set $MR = \{\mathbf{b}_1, \mathbf{b}_1, ..., \mathbf{b}_m\}$, where m is the number of representative objects.

To illustrate our calculation and merging processes more clearly, Fig. 2 provides a toy example shown in processes 1-4. The proposed micro partition based on the neighborhood set is outlined in Algorithm 1. The mechanism for merging $M^{\phi}(\mathbf{x}_i)$ is crucial and will be discussed in the next subsection.

3.3 Merge Micro-Clusters into Macro-Clusters

Based on our proposed micro partition, a hierarchical merging mechanism is presented to merge micro-clusters.

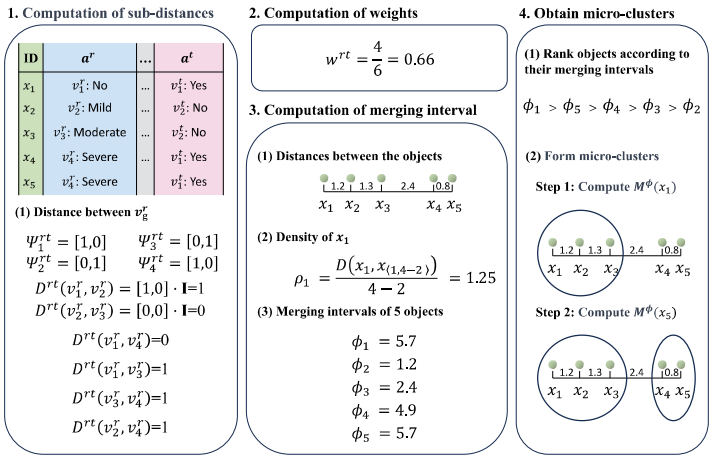


Fig. 2: A toy example illustrates the calculation processes. In processes 1 and 2, we compute the sub-distances in attribute a^r. Then, we compute the contribution of a^t to a^r. In processes 3 and 4, we confirm q_i, where $AM_1 = \{\mathbf{x}_{\langle 1,0 \rangle}, \mathbf{x}_{\langle 1,1 \rangle}, \mathbf{x}_{\langle 1,2 \rangle}, \mathbf{x}_{\langle 1,3 \rangle}, \mathbf{x}_{\langle 1,4 \rangle}\}$. After obtaining the merging intervals corresponding to each object, we then merge data objects into micro-clusters according to the descending order of the merging intervals.

Algorithm 1 MPNS: Micro Partition based on Neighborhood Set

Input: S, D.
Output: $M^{\phi}(\mathbf{x}_i)$, MR.

for $i = 1$ to n **do**
  Update the density ρ_i of $\mathbf{x}_i$ based on Eq. (11);
end for
for $i = 1$ to n **do**
  Update the merging interval ϕ_i of $\mathbf{x}_i$ based on Eq. (10);
end for
for $i = 1$ to n **do**
  if $\rho_i > 0$ **then**
    Select $\mathbf{x}_i$ as representative object $\mathbf{b}_i$ to MR;
  end if
  Update $M^{\phi}(\mathbf{x}_i)$ based on Eq. (12);
end for

Given the data set S and the number of clusters k, we iteratively compute micro-clusters $M^{\phi}(\mathbf{x}_i)$ and update data set S at each layer in the following two steps: 1) fix S, compute $M^{\phi}(\mathbf{x}_i)$ and micro representative object set MR by Algorithm 1 according to dissimilarity matrices D, and 2) fix MR, update S based on MR. Specifically, the hierarchical merging mechanism utilizes MR from the previous layer as the new local data set in the next layer. This process enables multiple partitioning and merging of objects while preserving the local

Algorithm 2 Mic2Mac: Merge Micro-Clusters into Macro-Clusters

Input: S, k.
Output: $\mathbf{Q}$.

```
Initialize the iteration counter by τ = 0; Set each object as a micro-cluster;
while |MR^(τ)| > k do
    Update D based on Eq. (5);
    Update M^(φ,(τ))(x_i) and MR^(τ) by Algorithm 1;
    Update S^(τ+1) by S^(τ+1) = MR^(τ);
    Update the iteration counter by τ = τ + 1;
end while
Update R = MR^(τ);
Compute Q^(τ) according to Eq. (3).
```

micro-clusters. These two steps iterate until $m = k$, where m is the number of representative objects. The overall Mic2Mac clustering algorithm is outlined in Algorithm 2.

Theorem 2. *The time complexity of* Mic2Mac *is* $O(d^2n + n\log n)$ *for each iteration.*

Proof. In the worst-case scenario, all attributes are categorical, and V is equal to the maximum number of possible values across all the categorical attributes. To analyze the overall complexity, we compute the complexity of $D^{(\tau)}$, $M^{\phi,(\tau)}$, and hierarchical merging once, respectively.

To compute the dissimilarity matrices $D^{(\tau)}$, we need to derive $d \times d$ pairs of CPDs by scanning n data objects in data set S. This results in a $O(d^2n)$ complexity. For computing the distances between a pair of intra-attribute possible values, it takes $O(V)$ complexity for every attribute. Thus, obtaining $D^{(\tau)}$ incurs a complexity of $O(nd^2 + V)$.

Given $D^{(\tau)}$ obtained from Algorithm 2, to compute $M^{\phi,(\tau)}$, we need to sort an $n \times n$ matrix, taking $O(n + n\log n)$ complexity. Subsequently, we sort n merging intervals in $O(n\log n)$ complexity. Therefore, computing $M^{\phi,(\tau)}$ takes $O(n + 2n\log n)$.

To implement hierarchical merging, we need to update S in each iteration according to the micro representative objects set MR, which takes $O(n)$.

Therefore, the overall complexity of Mic2Mac at a given iteration τ can be simplified to $O(d^2n + n\log n)$. □

4 Experiments

4.1 Experimental Settings

This section presents three types of experiments to comprehensively evaluate the clustering performance of our proposed Mic2Mac: (1) Clustering performance evaluation, (2) Ablation study, and (3) Visualization of cluster discrimination ability. Counterparts, validity indices, and data sets are introduced below.

Table 1: Summary of nine utilized data sets. The columns "Categorical", "Numerical", "Objects", and "Clusters" are the numbers of categorical attributes, numerical attributes, data objects, and clusters, respectively.

No.	Data Set	Abbrev.	Categorical	Numerical	Objects	Clusters
1	Dermatology	Derm	33	1	366	6
2	Autism-Adolescent	Autism	2	7	104	2
3	Common Toad	Toad	12	2	189	2
4	Hayes-Roth	Hayes	4	0	132	3
5	Breast Cancer	Cancer	9	0	286	2
6	Lymphography	Lym	18	0	148	4
7	Congressional Voting	Vote	16	0	435	2
8	Employee Selection	Employee	4	0	488	9
9	Social Workers	Workers	10	0	1000	4

Ten counterparts are compared, including Jia's Distance Metric (JDM) [14], Coupled Similarity Metric (CSM) [15], Entropy-based Distance Metric (EDM) [23], and Zhang's Distance Metric (ZDM) [21] incorporated with the conventional k-modes (KMD) [12] and k-prototypes (KPT) [11] approaches based on the attribute composition of data sets. Cheung's Iterative Learning (CIL) [7], designed for data sets with numerical and categorical attributes, is also selected. JDM, CSM, EDM, and ZDM represent state-of-the-art methods. Additionally, three conventional clustering algorithms, namely Attribute Weighting k-means (WKM) clustering algorithm [10], the original KMD, and KPT adopting Hamming and Euclidean distance metrics, are also included in the comparison. Furthermore, two variations of Mic2Mac, named Mic2-Mac$^{\mathrm{I}}$ and Mic2Mac$^{\mathrm{II}}$, are introduced for ablation studies, and additional details about these two Mic2Mac variants are provided in section 4.3.

Two validity indices have been chosen for comprehensively verifying the clustering performance, including CA [9] with a value range of [0, 1], and ARI [8] with a value range of [-1, 1]. A higher value for both these indices indicates better clustering performance.

Nine real-world data sets from various domains, including medicine, biology, sociology, etc., have been selected, which are shown in Table 1. Data sets 1-7 are public data sets collected from the UCI machine learning library[1]. Data sets 8 and 9 are obtained from the Weka website[2]. All data sets are pre-processed by removing objects with missing values.

4.2 Clustering Performance Evaluation

The clustering performance is reported in Tables 2 and 3, which are accessed by CA and ARI, respectively. The average ranks of the CA and ARI performances across all data sets for the compared methods are presented in Table 4, based on the results in Tables 2 and 3.

The key observations are as follows: (1) Mic2Mac consistently performs the best on most data sets in terms of CA index. (2) On certain data sets, such

Table 2: Clustering performance evaluated by CA, where the best results are highlighted in **bold** and the second-best results are underlined.

Methods	Derm	Autism	Toad	Hayes	Cancer	Lym	Vote	Employee	Workers
KMD	0.554±0.10	0.545±0.11	0.548±0.03	0.364±0.01	0.519±0.02	0.453±0.04	0.864±0.00	0.367±0.03	0.392±0.03
KPT	0.554±0.10	0.530±0.03	0.530±0.02	0.364±0.01	0.519±0.02	0.453±0.04	0.864±0.00	0.367±0.03	0.392±0.03
WKM	0.623±0.09	0.525±0.02	0.523±0.03	0.408±0.05	0.584±0.09	0.439±0.05	0.857±0.07	0.368±0.03	0.375±0.03
CIL	0.675±0.10	0.519±0.03	0.506±0.00	0.376±0.04	0.541±0.06	0.500±0.04	**0.881±0.00**	0.384±0.04	0.373±0.03
JDM	0.665±0.10	0.579±0.05	0.522±0.02	0.375±0.02	0.582±0.10	0.473±0.04	0.868±0.00	0.351±0.03	0.334±0.03
CSM	0.602±0.14	0.524±0.03	0.526±0.02	0.405±0.04	0.528±0.04	0.419±0.05	0.865±0.01	**0.402±0.04**	0.331±0.03
EDM	0.587±0.10	0.558±0.03	0.537±0.03	0.407±0.03	0.530±0.02	0.452±0.04	0.832±0.10	0.366±0.02	0.332±0.01
ZDM	0.685±0.11	0.558±0.02	**0.578±0.02**	0.404±0.03	0.569±0.19	0.470±0.04	0.872±0.00	0.368±0.03	0.374±0.03
Mic2Mac	**0.768±0.00**	**0.596±0.00**	0.545±0.00	**0.417±0.00**	**0.766±0.00**	**0.561±0.00**	0.874±0.00	0.393±0.00	**0.435±0.00**

Table 3: Clustering performance evaluated by ARI, where the best results are highlighted in **bold** and the second-best results are underlined.

Methods	Derm	Autism	Toad	Hayes	Cancer	Lym	Vote	Employee	Workers
KMD	0.396±0.15	-0.003±0.01	-0.002±0.02	-0.012±0.00	-0.004±0.00	0.113±0.04	0.530±0.00	0.162±0.02	0.057±0.02
KPT	0.422±0.12	-0.003±0.01	-0.008±0.01	-0.012±0.00	-0.004±0.00	0.113±0.04	0.530±0.00	0.162±0.02	0.057±0.02
WKM	0.509±0.09	-0.006±0.01	-0.008±0.01	0.007±0.02	0.040±0.07	0.085±0.04	0.527±0.11	0.172±0.03	0.046±0.02
CIL	0.606±0.10	-0.007±0.01	-0.021±0.00	-0.004±0.02	0.011±0.04	**0.182±0.05**	**0.579±0.00**	0.193±0.02	0.052±0.02
JDM	0.614±0.13	0.018±0.03	-0.014±0.01	-0.006±0.01	0.041±0.07	0.123±0.04	0.541±0.01	0.167±0.02	0.052±0.01
CSM	0.518±0.17	-0.009±0.01	-0.008±0.01	0.008±0.02	0.003±0.02	0.089±0.04	0.532±0.03	**0.212±0.03**	0.051±0.02
EDM	0.439±0.12	0.006±0.01	0.002±0.01	0.008±0.02	0.007±0.01	0.089±0.03	0.478±0.17	0.163±0.04	0.059±0.01
ZDM	0.627±0.15	-0.015±0.01	**0.013±0.02**	0.007±0.02	0.062±0.02	0.132±0.05	0.553±0.01	0.211±0.02	0.076±0.01
Mic2Mac	**0.678±0.00**	**0.019±0.00**	-0.001±0.00	**0.009±0.00**	**0.109±0.00**	0.129±0.00	0.557±0.00	0.173±0.00	**0.085±0.00**

Table 4: Ave. Rank of CA and ARI rows report the average performance ranks, where the best results are highlighted in **bold**, while the second-best results are underline.

Ave. Rank	KMD	KPT	WKM	CIL	JDM	CSM	EDM	ZDM	Mic2Mac
Ave. Rank @ CA	5.944	6.389	5.278	4.889	5.222	6.111	6.167	3.556	**1.444**
Ave. Rank @ ARI	6.611	6.722	6.389	4.722	4.833	5.556	5.222	3.056	**1.889**

as Toad, Vote, and Employee, Mic2Mac does not achieve the best result, but the performance gaps between Mic2Mac and the best-performing counterparts are tiny, also highlighting the superiority of Mic2Mac. (3) While Mic2Mac does not yield the best results in terms of the ARI on some data sets, e.g., Lym and Employee, it consistently performs the best and the second-best on most data sets, which still verifies its effectiveness. Intuitively, if a data set contains only numerical attributes, the performance of Mic2Mac downgrades to traditional k-means. The more categorical attributes a data set contains, the better the Mic2Mac can perform. Meanwhile, Mic2Mac also performs well on mixed data.

4.3 Ablation Study

In ablation studies, we focus on the clustering performance assessed by the ARI. Firstly, to assess the effectiveness of the dissimilarity metric proposed for heterogeneous attributes, we restrict Mic2Mac to utilize the combination of Hamming distance and Euclidean distance to tackle mixed data, forming Mic2Mac$^{\text{I}}$. Secondly, to evaluate the effectiveness of our proposed hierarchical merging mechanism, we compare Mic2Mac and Mic2Mac$^{\text{I}}$ with their variation Mic2Mac$^{\text{II}}$, which

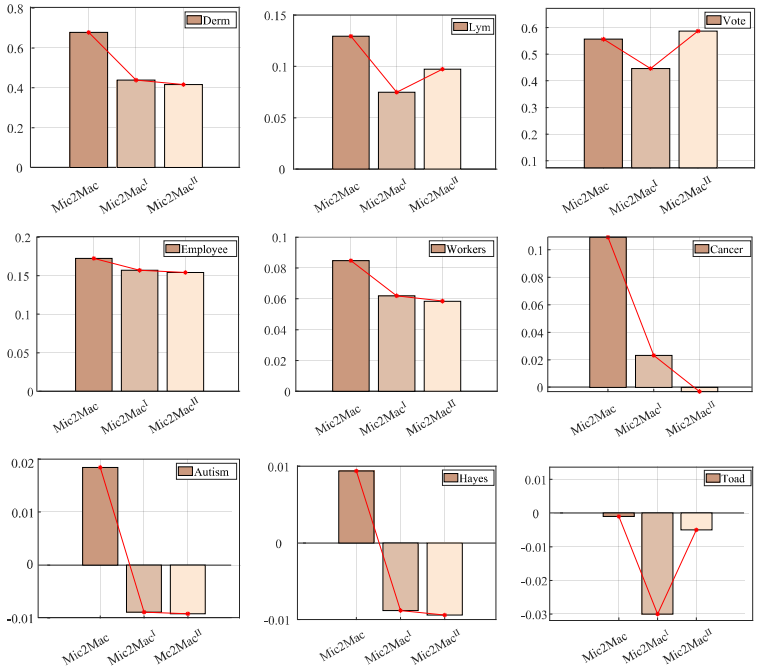
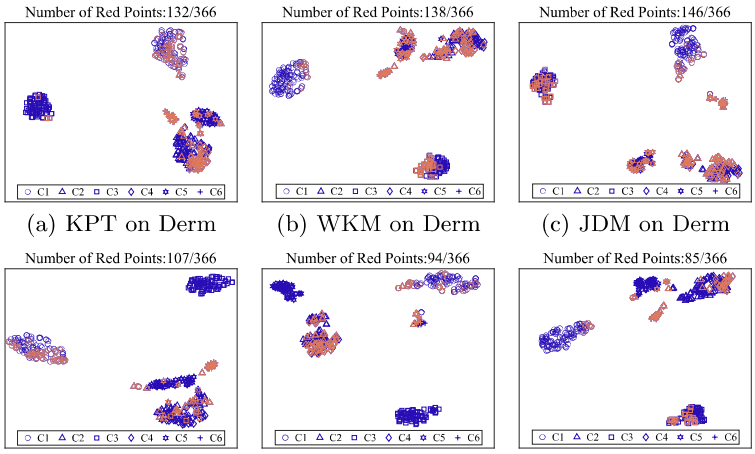


Fig. 3: Comparison of clustering performance among Mic2Mac, Mic2Mac$^{\text{I}}$, and Mic2Mac$^{\text{II}}$ on all the 9 data sets. A better measure has a higher value. The Ave. Rank @ ARI of Mic2Mac, Mic2Mac$^{\text{I}}$, and Mic2Mac$^{\text{II}}$ are 1.111, 2.333, and 2.556, respectively.

incorporates the partitioning strategy of KPT by partitioning the representative objects after the first formation of the micro-clusters. The clustering performance and the average rank of Mic2Mac with its two variations are illustrated in Fig. 3.

The overall result reveals that Mic2Mac consistently outperforms its two variations, demonstrating the effectiveness of Mic2Mac. Specifically, Mic2Mac surpasses Mic2Mac$^{\text{I}}$ on nine data sets, indicating that Mic2Mac can effectively measure the original heterogeneous attribute data information. Furthermore, Mic2Mac outperforms Mic2Mac$^{\text{II}}$ on eight data sets, and Mic2Mac$^{\text{I}}$ performs better than Mic2Mac$^{\text{II}}$ on six data sets. This emphasizes the effectiveness of the proposed merging mechanism. The reason why Mic2Mac$^{\text{I}}$ perform worse than Mic2Mac$^{\text{II}}$ on certain data sets (i.e. Toad, Lym, and Vote) would be that Mic2Mac$^{\text{I}}$ employs the simplest Euclidean and Hamming distance measures, which makes it hard to handle the complex issues in real-world data distribution, e.g., overlapping, and coupling categorical attributes.



(a) KPT on Derm (b) WKM on Derm (c) JDM on Derm

(d) CSM on Derm (e) ZDM on Derm (f) Mic2Mac on Derm

Fig. 4: t-SNE visualization of the Derm data sets, where data points marked in "C1" to "C6" indicate their "true" cluster labels, while objects marked in red indicate they were incorrectly clustered.

4.4 Visualization

In Fig. 4, t-SNE [18] is employed to showcase the cluster discrimination ability of Mic2Mac. The Derm data set is first clustered using KPT, WKM, JDM, CSM, ZDM, and Mic2Mac. Subsequently, the data is encoded according to the distance matrix of objects created by the distance metrics of the corresponding approaches, respectively. These distance matrices are treated as the representations of the data and are then projected into two-dimensional space using t-SNE for visualization. Data points are marked with different markers to indicate their "true" cluster labels. The red markers are utilized to indicate the objects that are incorrectly clustered. Intuitively, fewer red markers indicate a more accurate clustering performance and a more separable distribution of different markers indicates a more powerful cluster discrimination ability.

The visualization in Fig. 4 clearly shows that Mic2Mac exhibits fewer red markers and a more separable distribution of different markers, signifying its stronger cluster discrimination ability than the conventional and state-of-the-art methods.

5 Concluding Remarks

In this paper, a novel approach called Mic2Mac has been proposed for mixed data clustering, which simultaneously tackles two challenges inherent in clustering real-world mixed data sets, i.e., the information gap of heterogeneous attributes and the bias brought by prior knowledge. To address these challenges, we have

proposed: (1) A heterogeneous attribute metric for preserving and leveraging original data information; (2) A micro partition approach based on neighborhood set theory for forming unbiased micro-clusters; and (3) A merging mechanism for hierarchically merging micro-clusters into sought number of clusters. The superiority of Mic2Mac is evidenced through extensive experiments. Moreover, the clustering process of Mic2Mac is highly interpretable due to the nested relationship among multi-granular clusters extracted during the merging phase.

In the future, this research will be extended to address more challenging clustering analysis tasks, e.g., federated mixed data clustering, and exploring cluster patterns for unstructured multi-modal data. Moreover, the potential of the dendrogram formed by merging the micro-clusters will also be explored for understanding complex data sets.

Acknowledgements

References

1. Agresti, A.: Categorical Data Analysis. Wiley Series in Probability and Statistics, Wiley (2002)
2. Alamuri, M., Surampudi, B.R., Negi, A.: A survey of distance/similarity measures for categorical data. In: The 24th International Joint Conference on Neural Networks. pp. 1907–1914. IEEE (2014)
3. Arabie, P., Baier, N.D., Critchley, C.F., Keynes, M.: Studies in classification, data analysis, and knowledge organization. Springer (2006)
4. Cai, S., Zhang, Y., Luo, X., Cheung, Y.m., Jia, H., Liu, P.: Robust categorical data clustering guided by multi-granular competitive learning. In: The IEEE 44th International Conference on Distributed Computing Systems. pp. 288–299 (2024)
5. Chen, J., Ji, Y., Zou, R., Zhang, Y., Cheung, Y.m.: QGRL: Quaternion graph representation learning for heterogeneous feature data clustering. In: The 30th SIGKDD Conference on Knowledge Discovery and Data Mining. pp. 1–10 (2024)
6. Cheng, M., You, X.: Leachable component clustering. In: The 26th International Conference on Pattern Recognition. pp. 1858–1864 (2022)
7. Cheung, Y.m., Jia, H.: Categorical-and-numerical-attribute data clustering based on a unified similarity metric without knowing cluster number. Pattern Recognition **46**(8), 2228–2238 (2013)
8. Gates, A.J., Ahn, Y.Y.: The impact of random models on clustering similarity. The Journal of Machine Learning Research **18**, 3049–3076 (2017)
9. He, X., Cai, D., Niyogi, P.: Laplacian Score for Feature Selection. In: The 17th Advances in Neural Information Processing Systems. pp. 507–514 (2005)

10. Huang, J., Ng, M., Hongqiang Rong, Zichen Li: Automated variable weighting in k-means type clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence **27**(5), 657–668 (2005)
11. Huang, Z.: Clustering large data sets with mixed numeric and categorical values. In: The 1st Pacific-Asia Conference on Knowledge Discovery and Data Mining. pp. 21–34 (1997)
12. Huang, Z.: Extensions to the k-means algorithm for clustering large data sets with categorical values. Data Mining and Knowledge Discovery **2**(3), 283–304 (1998)
13. Ienco, D., Pensa, R.G., Meo, R.: From Context to Distance. ACM Transactions on Knowledge Discovery from Data **6**(1), 1–25 (2012)
14. Jia, H., Cheung, Y.m., Liu, J.: A New Distance Metric for Unsupervised Learning of Categorical Data. IEEE Transactions on Neural Networks and Learning Systems **27**(5), 1065–1079 (2016)
15. Jian, S., Cao, L., Lu, K., Gao, H.: Unsupervised Coupled Metric Similarity for Non-IID Categorical Data. IEEE Transactions on Knowledge and Data Engineering **30**(9), 1810–1823 (2018)
16. Jian, S., Pang, G., Cao, L., Lu, K., Gao, H.: CURE: Flexible Categorical Data Representation by Hierarchical Coupling Learning. IEEE Transactions on Knowledge and Data Engineering **31**(5), 853–866 (2019)
17. Qian, Y., Li, F., Liang, J., Liu, B., Dang, C.: Space Structure and Clustering of Categorical Data. IEEE Transactions on Neural Networks and Learning Systems **27**(10), 2047–2059 (2016)
18. Van Der Maaten, L., Hinton, G.: Visualizing Data using t-SNE. Journal of Machine Learning Research **9**(86), 2579–2605 (2008)
19. Wang, P., Zhang, Y., Zhang, Y., Lu, Y., Li, M., Cheung, Y.m.: Clustering by learning the ordinal relationships of qualitative attribute values. In: The 34th International Joint Conference on Neural Networks. pp. 1–8 (2024)
20. Xu, J., Lei, B., Gu, Y., Winslett, M., Yu, G., Zhang, Z.: Efficient Similarity Join Based on Earth Mover's Distance Using MapReduce. IEEE Transactions on Knowledge and Data Engineering **27**(8), 2148–2162 (2015)
21. Zhang, Y., Cheung, Y.m.: A New Distance Metric Exploiting Heterogeneous Inter-attribute Relationship for Ordinal-and-Nominal-Attribute Data Clustering. IEEE Transactions on Cybernetics **52**(2), 758–771 (2022)
22. Zhang, Y., Cheung, Y.m.: Graph-Based Dissimilarity Measurement for Cluster Analysis of Any-Type-Attributed Data. IEEE Transactions on Neural Networks and Learning Systems **34**(9), 6530–6544 (2023)
23. Zhang, Y., Cheung, Y.m., Tan, K.C.: A Unified Entropy-Based Distance Metric for Ordinal-and-Nominal-Attribute Data Clustering. IEEE Transactions on Neural Networks and Learning Systems **31**(1), 39–52 (2020)
24. Zhang, Y., Zou, R., Zhang, Y., Zhang, Y., Cheung, Y.m., Li, K.: Adaptive micro partition and hierarchical merging for accurate mixed data clustering. Complex & Intelligent Systems pp. 1–13 (2024)
25. Zhao, M., Feng, S., Zhang, Y., Li, M., Lu, Y., Cheung, Y.m.: Learning order forest for qualitative-attribute data clustering. In: The 27th European Conference on Artificial Intelligence. pp. 1–8 (2024)
26. Zou, R., Zhang, Y., Zhang, Y., Lu, Y., Li, M., Cheung, Y.m.: Federated clustering with unknown number of clusters. In: The 6th International Conference on Data-driven Optimization of Complex Systems. pp. 1–6 (2024)

MACL: Metric and Attribute Space Co-Learning for Qualitative Data Clustering

[redacted] Xinxi Chen[1], [redacted]

Abstract. Cluster analysis of unlabeled categorical data is crucial in many practical applications, such as medical data analysis, financial risk warnings, recommendation systems, etc. Compared with numerical data in explicit distance space, the adoption of metrics is often critical to the success of cluster analysis on categorical data, where qualitative values do not initially have well-defined similarities. However, categorical data metrics are often defined based on certain prior knowledge with limitations, and a particular metric usually cannot reasonably serve the clustering on different datasets. Furthermore, without a well-established metric space, advanced downstream processing probably cannot function properly to enhance clustering performance. This paper, therefore, first proposes to learn a fusion of metrics that complement each other and then learns to adapt the fusion to clustering tasks for a more appropriate exploration of clusters. Experiments on real public datasets from various domains illustrate the superiority and stability of the proposed method in categorical data clustering.

Keywords: Qualitative Data Clustering · Metric Space Learning · Attributes Weighting · Ensemble Learning

1 Introduction

Categorical data attributes are made up of qualitative values that are not suited for mathematical operations on numerical data and do not have a well-defined metric space. It is common to analyze the categorical data in typical data analysis domains such as medical, financial, and social behavior data analysis [2,8,21]. However, exploring object groups for the qualitative-valued categorical data in an unsupervised way is a challenging task. Thus, the clustering of categorical data keeps absorbing attention in the literature [6,13,25], and the existing attempts can be roughly divided into two streams: (1) directly define a metric/measure for similarity-based clustering[12,17], and (2) encode categorical values into numerical ones and perform numerical data clustering[22,34].

For the metric/measure defining stream, Hamming Distance is probably the most common metric adopted by the k-modes algorithm proposed for categorical data clustering. It cannot represent the similarity of data objects finely since it treats the dissimilarity between different values identically. Later, more advanced metrics [14,28,30,31] for defining inter-value distances based on attribute context are proposed. These measures perform much better as more statistical information is considered. Some other approaches [3,20] also adopt entropy to quantify object-cluster affiliations to search for better clustering results of categorical data.

For the encoding stream, the conventional one-hot encoding converts categorical attribute values into integers, and then represents the integer values into binary vectors. Since such encoding is equivalent to assigning identical distance "1" to any pair of unequal values, the intrinsic structure of categorical data is overlooked. To address this issue, approaches encoding categorical values based on the inter- and/or intra-attribute statistic are presented in the literature [15,23,39], which can obtain more informative representations of data. It is worth noting that the object-level encoding approach [23] reconstructs the whole dataset using the inter-object similarities to achieve better performance.

Most of the above solutions actually concentrate on the common ultimate goal, i.e., providing a more appropriate distance metric space for cluster learning. Recently, in this direction, more advanced similarity metrics [27,29,33] and encoding approaches [32,38] have been proposed, which not only informatively define the distance space, but also tune the defined distance space w.r.t. the clustering task. As a result, they can provide a more proper space as the basis for categorical data clustering. However, their space defining is still based on certain prior knowledge or hypothesis, which cannot precisely reflect the true structure of distance space of categorical data with attributes described by the vague qualitative values, and thus degenerates the effectiveness of downstream processing [11].

This paper, therefore, proposes a categorical data clustering paradigm, which first learns to fuse multiple metrics to complement the embedded information, and then performs subspace learning on the represented attributes to leverage accurate clustering. As a distance Metric space and an Attribute subspace, are Collaboratively Learned, the proposed method is named **MACL**. Since dual space learning and cluster learning are connected to facilitate information passing, MACL is flexible in adapting distances to the clustering tasks w.r.t. various datasets. As a result, MACL features superior clustering accuracy. Moreover, the suitability issue of categorical data metrics w.r.t. clustering is initially revealed and addressed by this paper. Comprehensive experimental evaluations illustrate the promising performance of the proposed method. Two main contributions are summarized below:

1. Dual space learning paradigm is proposed for solving categorical data clustering problems. It first learns metric space to provide a rational learning foundation, and then learns to obtain a subspace for more flexible detection of clusters.

Table 1. Explanation of symbols

Symbol	Explanation
X	Whole dataset with n objects
k	Number of clusters
$\mathbf{x}_j$	jth data object
x_{jh}	hth attribute value of jth data object
$\mathbf{U}$	Membership matrix
u_{ij}	Membership indicator of $\mathbf{x}_j$ to c_i
c_i	ith cluster
c_{ih}	hth attribute value of ith cluster's mode
a_h	hth attribute
v_h	Number of possible values of hth attribute
o_{lh}	lth possible value of hth attribute
w_s	sth weight of metric in combined metrics
w_{ih}	Weight of attribute a_h to cluster c_i
m	Number of combined metrics
d	Number of attributes of dataset
ς	A threshold for convergence determination
$\Xi(\cdot)$	Number of data objects
T	Maximum iteration
W^M	Weights of combined metrics
$\mathbf{W}^C$	A $k \times d$ matrix with weights of d attributes corresponding to k clusters
$\tau(\cdot)$	New distance metric
$\tau^s(\cdot)$	sth basic distance metric

2. A new metric space learning strategy is dedicatedly designed to leverage multiple metrics to complement each other, and it is revealed that diversified metrics are more comprehensive in approaching the optimal metric space.

2 Related Work

This section overviews the related similarity measures, encoding strategies, and representation learning methods.

2.1 Similarity Measures

Similarity measures, e.g., Entropy-based ones [18,20,31], use information entropy to quantify object-cluster associations. Later, approaches [1,16] evaluate the distance between two values as the differences in their associated conditional probability distributions reflected by the other attributes. In [14], a group of highly

associated attributes is formed as context for enhancing distance measurement. Most recently, the measures [28,30] are proposed to consider both the intra- and inter-cluster information simultaneously to form a more comprehensive metric.

2.2 Encoding Strategies

One-hot encoding is a conventional method that gives any pair of unequal values identical distances "1", and thus unavoidably ignores the statistical information of categories. Therefore, the method [23] has been proposed to encode values based on the interaction of attribute values. Later, the measure [15] considers coupling relationships between intra-attributes and inter-attributes for encoding to obtain a more informative representation.

2.3 Representation Learning Methods

Representation learning methods include the conventional approaches that learn the importance of a whole attribute. Advanced methods [29,33] first informatively represent categorical values, then learn the represented inter-value distances and attribute importance. However, they are still based on hand-crafted encoding and measures. Most recently, nested cluster distribution learning approaches [6,7,26,36], ordinal attribute value relationship mining methods [26,27,37], and distributed clustering algorithms [4,19,35,40] have been proposed. Since they focus on addressing the other complexity in the clustering field, they are relatively incompetent in solving our focused metric space and attribute subspace learning problems.

3 Proposed Method

We present a learning paradigm to iteratively optimize the combined metric space w.r.t. clustering task. Table 1 lists the symbols that appear frequently in this work. Specific definitions of the symbols will also be provided where they first appear.

3.1 Metric Space Learning

The learning framework is to learn the combinations of metrics that have an impact on clustering. We can construct a new metric $\tau < w_1, w_2, ..., w_m >$ by combining the existing m different metrics through their corresponding weights w_1, w_2, ..., w_m. When the new metric is determined by the weights, we use $\tau(\mathbf{x}_j, c_i)$ to indicate the similarity between an object $\mathbf{x}_j$ and a cluster c_i in a new metric space, where $i \in \{1, 2, ..., k\}$ and $j \in \{1, 2, ..., n\}$. When given a partition, it can be expressed by $\mathbf{U}$, where u_{ij} is its (i, j)-th entry indicating the affiliation of object $\mathbf{x}_j$ to cluster c_i. For a u_{ij} satisfying

$$i = \arg\min_{y} \tau(\mathbf{x}_j, c_y), \tag{1}$$

Algorithm 1 Distance Metric Space Learning Algorithm

Input: X, k, T, m, ς.
Output: $\mathbf{U}$, W^M.

1: Set the number of iterations $t = 0$, $Converged = False$, initialize $\mathbf{U}$ randomly, and initialize the W^M by setting each of its weights as $w_s = 1/m$.
2: **while** $t \leq T$ and $Converged = False$ **do**
3: Fix W^M, update $\mathbf{U}^{'}$ by k-modes.
4: Fix $\mathbf{U}^{'}$, update W^M by Eqs. (4) and (5).
5: **if** $|O - O^{'}|/O \leq \varsigma$ **then**
6: $Converged = True$.
7: **end if**
8: **end while**

$$\text{s.t. } y \in \{1, 2, ..., k\},$$

we have $u_{ij} = 1$, otherwise, $u_{ij} = 0$.

To learn the importance of basic metrics, we solve the clustering problem by learning the weights of the combined metrics. The weights of the combined metrics are denoted as $W^M = \{w_s\}_{s=1}^m$, where a weight w_s satisfies $0 \leq w_s \leq 1$, and m corresponds to the number of combined metrics. The purpose of our metric learning is to minimize the value of the objective function O:

$$O = \sum_{i=1}^{k} \sum_{j=1}^{n} u_{ij} \cdot \tau(\mathbf{x}_j, c_i), \tag{2}$$

where the new metric $\tau(\mathbf{x}_j, c_i)$ can be defined based on the basic distance metrics $\tau^s(\mathbf{x}_j, c_i)$ as

$$\tau(\mathbf{x}_j, c_i) = \sum_{s=1}^{m} \left(w_s \cdot \tau^s(\mathbf{x}_j, c_i)\right). \tag{3}$$

Note that in this paper, the conventional CBDM [14] and two advanced metrics, i.e., UDM [30] and HDM [29], are chosen to be the basic distance metrics $\tau^s(\cdot, \cdot)$.

So far, the problem becomes how to obtain appropriate W^M, which can be updated in the Monte Carlo style by the following three steps: (1) Fix W^M, update the $\mathbf{U}$ until the objective function O is minimized; (2) Fix $\mathbf{U}$, update W^M with a gradient descent in accordance with the contribution of each metric to the current O; (3) Iteratively implement steps (1) and (2) until O no longer decreases or changes. The weights are computed by:

$$w_s = w_s - \frac{\tau^s(\mathbf{x}_j, c_i)}{O} \cdot \alpha, \tag{4}$$

and then processed by soft-max as:

$$w_s = \frac{w_s}{\sum_{y=1}^{m} w_y}, \tag{5}$$

Algorithm 2 MACL Learning Algorithm

Input: X, T, k, ς.
Output: $\mathbf{U}$.

1: Set the number of iterations $t = 0$, $Converged = False$, initialize $\mathbf{U}$ randomly, and initialize the $\mathbf{W}^C$ by setting each of its weights as $w_{ih} = 1/(k \times d)$.
2: **while** $t \leq T$ and $Converged = False$ **do**
3: Fix $\mathbf{W}^C$, Obtain W^M and $\mathbf{U}$ by Algorithm 1;
4: Fix $\mathbf{U}$ and W^M, update $\mathbf{W}^C$ by Eqs. (6), (7) and (8).
5: **if** $|O - O^{'}|/O \leq \varsigma$ **then**
6: $Converged = True$.
7: **end if**
8: **end while**

which is in the interval $[0, 1]$ and all the weights satisfies $\sum_{s=1}^{m} w_s = 1$. α is a hyper-parameter that is used to control the rate of gradient descent. It is intuitive that w_s is heuristically updated according to how much it contributes to the minimization of O.

The overall procedure is summarized in Algorithm 1. Note that it is possible to combine any basic distance metric based on the above strategy to form a new metric for clustering analysis. But in practice, according to the diversity principle, we suggest selecting metrics proposed based on different principles as the basic distance metrics for combination.

3.2 MACL: Metric and Attribute Space Co-Learning

Based on the metric space learning, we further extend it into a dual space learning algorithm. The basic idea is to refine the attribute weights based on the learned combined metric, which measures the contribution of each attribute in minimizing O. $\mathbf{W}^C$ is a $k \times d$ matrix with d attribute weights corresponding to k clusters. For example, w_{ih} is the (i, h)-th entry of $\mathbf{W}^C$, which is the weight of attribute a_h to cluster c_i. To quantify the contribution of each attribute, we start by explaining cluster discrimination C^{out} and cluster compactness C^{in}.

The Hellinger distance [5] quantifies the difference between two probability distributions, and is adopted to quantify the overall discrimination ability of an attribute a_h in discriminating a cluster c_i from the combination of all the other clusters $X \backslash c_i$:

$$C_{ih}^{out} = \sqrt{\sum_{l=1}^{v_h} \left(\frac{\Xi(c_i | c_{ih} = o_{lh})}{\Xi(c_i)} - \frac{\Xi(c_i | c_{ih} \neq o_{lh})}{\Xi(X \backslash c_i)} \right)^2}, \tag{6}$$

where o_{lh} is the l-th possible value in attribute a_h and $\Xi(c_i | c_{ih} = o_{lh})$ represents the number of objects with their l-th values equal to o_{lh} in cluster c_i. Similarly, $\Xi(c_i | c_{ih} \neq o_{lh})$ means the number of objects with their l-th values not equal to

o_{lh} in cluster c_i. Cluster compactness is defined as

$$C_{ih}^{in} = \sqrt{\sum_{l=1}^{v_h} \left(\frac{\Xi(c_i | c_{ih} = o_{lh})}{\Xi(c_i)} \right)^2}. \tag{7}$$

When both C_{ih}^{out} and C_{ih}^{in} reach a larger value, it can be indicated that attribute a_h contributes more in forming c_i, as a_h helps to discriminate c_i from the other clusters and gather similar objects into c_i. Accordingly, weight w_{ih} indicating the importance of a_h in forming c_i is defined as

$$w_{ih} = \frac{C_{ih}^{in} \cdot C_{ih}^{out}}{\sum_{y=1}^{d} C_{iy}^{in} \cdot C_{iy}^{out}}, \tag{8}$$

and all the weights in $\mathbf{W}^C$ are updated accordingly by Eq. (8). So far, the objective function O in Eq. (2) can be rewritten with considering w_{ih} as:

$$O(U, W^M, \mathbf{W}^C) = \sum_{i=1}^{k} \sum_{j=1}^{n} u_{ij} \sum_{s=1}^{m} \left(w_s \cdot \sum_{h=1}^{d} \tau^s \left(x_{jh}, c_{ih} \right) \cdot w_{ih} \right), \tag{9}$$

where $\tau^s \left(x_{jh}, c_{ih} \right)$ is the similarity measure between the object x_j and cluster c_i w.r.t. h-th attribute in the s-th combined basic distance metric. Algorithm 2 summarizes the optimization process of the objective function in Eq. (9).

4 Experiments

The proposed MACL is compared with four different clustering approaches on six real benchmark datasets.

Table 2. Statistics of six datasets.

Datasets	**Abbr.**	**#Objects** (n)	**#Attributes** (d)	**#True Clusters** (k^*)
Balance Scale	BS	624	4	3
Soybean Large	SL	266	35	15
Mammographic	MM	824	4	2
Hayes Roth	HR	132	4	3
Lymphography	LG	148	18	4
Lenses	LS	999	4	5

4.1 Experimental Setup

We use six real datasets from various fields obtained through the UCI machine learning repository [9]. Statistics of the datasets are sorted out in Table 2. Data

Table 3. ARI and NMI performance of different clustering methods. The best and second-best base-performing methods are emphasized in **bold** and underline,respectively.

Method	Index	BS	SL	MM	HR	LG	LS	Avg. Rank
KMD	ARI	0.112±0.007	0.393±0.014	0.407±0.000	0.015±0.003	0.167±0.015	0.046±0.005	4.33
	NMI	0.100±0.008	0.652±0.006	0.333±0.000	0.034±0.012	0.192±0.002	0.065±0.007	4.33
CBDM	ARI	**0.153±0.001**	0.426±0.020	**0.428±0.000**	0.036±0.031	0.196±0.044	0.051±0.008	2.17
	NMI	**0.242±0.002**	0.692±0.015	**0.341±0.000**	0.052±0.044	0.234±0.026	0.083±0.004	2.42
UDM	ARI	0.092±0.007	0.402±0.008	0.425±0.000	0.024±0.007	0.203±0.031	0.094±0.009	3.33
	NMI	0.083±0.021	0.677±0.007	0.338±0.000	0.042±0.006	0.243±0.008	0.131±0.020	3.33
HDM	ARI	0.137±0.029	0.420±0.022	0.407±0.000	0.013±0.002	0.206±0.036	0.039±0.004	3.83
	NMI	0.143±0.035	0.696±0.025	0.326±0.000	0.022±0.003	0.253±0.020	0.060±0.004	3.67
MACL (ours)	ARI	0.150±0.004	**0.447±0.011**	**0.428±0.000**	**0.059±0.019**	**0.222±0.022**	**0.100±0.031**	1.17
	NMI	0.180±0.032	**0.732±0.025**	**0.341±0.000**	**0.060±0.017**	**0.261±0.009**	**0.150±0.036**	1.25

objects with missing values are entirely discarded. Four counterparts are the traditional k-modes (KMD) and conventional CBDM [14], and the state-of-the-art UDM [30] and HDM [29]. CBDM, UDM, and HDM are also selected as three basic metrics for combination in our method. Two validity indices, i.e., Adjusted Rand Index (ARI) [24] and Normalized Mutual Information (NMI) [10] are utilized for clustering performance evaluation. ARI is in the value range $[-1, 1]$, and NMI is in the value range $[0, 1]$. They both indicate a better performance with a higher value.All the hyperparameters of compared methods (if any) are set according to the original papers.

4.2 Clustering Performance Evaluation

We randomly initialize the k modes and implement clustering ten times for all compared clustering methods, and the mean performance of the top-3 results of each method is reported. Table 3 compare the ARI and NMI clustering performance, respectively. The best and the second-best base-performing methods are marked by boldface and underline, respectively. The average rank of each method on all the datasets is reported in the bottom row of the tables. It can be observed that our MACL always performs the best or second-best on all the datasets because it combines context-based CBDM, entropy-based UDM, and the probability-based HDM to let them complement in forming the new distance metric space. In short, MACL is very stable and competent in categorical data clustering.

4.3 Experimental Results Visualization

We visualize the average rank of all the compared methods and the convergence curve of MACL in Fig 1. It can be seen from the left part of Fig 1 that MACL ranks the highest in terms of ARI and NMI, which intuitively reflects its superiority. It can be observed from the right part of Fig 1 that MACL converges very quickly, with at most five iterations for the convergence of k-modes (i.e., line 3 of Algorithm 1), two iterations for the convergence of metric space learning

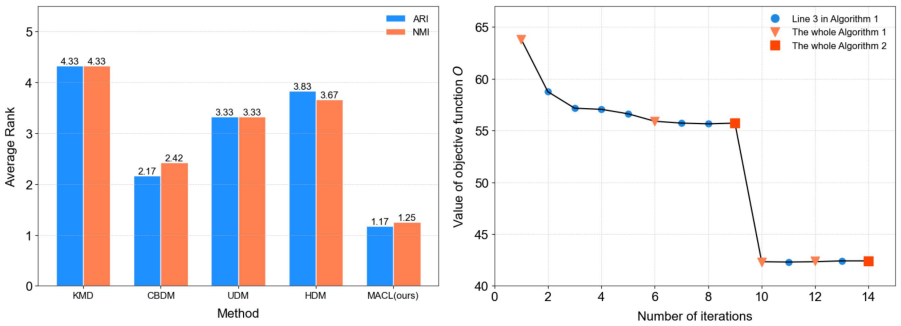


Fig. 1. Average ARI and NMI ranks (left) and convergence curve of MACL on Soybean Large (right). Blue dots, orange triangles, and red squares indicate the execution of line 3 in Algorithm 1, the whole Algorithm 1, and the whole Algorithm 2, respectively.

(i.e., the whole Algorithm 1), and two iterations for the convergence of subspace learning (i.e., the whole Algorithm 2). In general, MACL performs the best and is efficient in general.

4.4 Parameter Analysis

Parameter α has a significant role in influencing the performance of the proposed algorithm, which is tested within the interval [0, 0.1] using a step size of 0.01. The mean ARI and NMI of the proposed approach are computed on all the datasets, for each value of α. Due to space limitation, this part of results are omitted from the paper, but some discussions are provided below. Note that $\alpha = 0.06$ is observed to be a suitable choice for majority datasets. Besides, as it is difficult to make the algorithm converge by solely assessing reductions in the objective function, convergence threshold, i.e., ς, is adopted, and its value should be set at $\varsigma = 0.001$ to achieve a generally satisfactory convergence performance according to the experiments. When the change in the objective function falls below the value of ς, the algorithm is considered to have converged.

5 Conclusion

In this paper, we investigate the vital problem of categorical data clustering, i.e., how to cluster categorical data objects in an appropriate distance metric space. It is intuitive that a single metric defined based on certain prior knowledge may not be suitable for different clustering tasks, which results in a performance bottleneck. We propose to incorporate multiple distance metrics through learning to let them complement each other. The newly formed reasonable metric space thus becomes the basis for clustering categorical data in subspace. As the metric space and subspace are jointly learned with the clustering task, the proposed

MACL achieves competitive clustering accuracy and is robust to different clustering tasks, i.e., clustering datasets in various domains, with different numbers of sought clusters k.

References

1. Ahmad, A., Dey, L.: A method to compute distance between two categorical values of same attribute in unsupervised learning for categorical data set. Pattern Recognition Letters **28**(1), 110–118 (2007)
2. Ahmad, A., Khan, S.S.: Survey of state-of-the-art mixed data clustering algorithms. Ieee Access **7**, 31883–31902 (2019)
3. Barbará, D., Li, Y., Couto, J.: Coolcat: an entropy-based algorithm for categorical clustering. In: Proceedings of the 7th International Conference on Information and Knowledge Management. pp. 582–589 (2002)
4. Bendechache, M., Kechadi, M.T.: Distributed clustering algorithm for spatial data mining. In: 2015 2nd IEEE International Conference on Spatial Data Mining and Geographical Knowledge Services (ICSDM). pp. 60–65. IEEE (2015)
5. Bhattacharyya, A.: On a measure of divergence between two statistical populations defined by their probability distribution. Bulletin of the Calcutta Mathematical Society **35**, 99–110 (1943)
6. Cai, S., Zhang, Y., Luo, X., Cheung, Y.M., Jia, H., Liu, P.: Robust categorical data clustering guided by multi-granular competitive learning. In: 2024 IEEE 44th International Conference on Distributed Computing Systems (ICDCS). pp. 288–299. IEEE (2024)
7. Chen, J., Ji, Y., Zou, R., Zhang, Y., Cheung, Y.m.: Qgrl: quaternion graph representation learning for heterogeneous feature data clustering. In: Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. pp. 297–306 (2024)
8. Chu, K., Zhang, M., Xun, Y., Zhang, J.: A hybrid similarity measure-based clustering approach for mixed attribute data. International Journal of Machine Learning and Cybernetics **15**(4), 1295–1311 (2024)
9. Dua, D., Graff, C., et al.: Uci machine learning repository (2017), https://archive.ics.uci.edu/ml/index.php
10. Estévez, P.A., Tesmer, M., Perez, C.A., Zurada, J.M.: Normalized mutual information feature selection. IEEE Transactions on Neural Networks **20**(2), 189–201 (2009)
11. Feng, S., Zhao, M., Huang, Z., Ji, Y., Zhang, Y., Cheung, Y.M.: Robust qualitative data clustering via learnable multi-metric space fusion. In: ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 1–5. IEEE (2025)
12. Gibson, D., Kleinberg, J., Raghavan, P.: Clustering categorical data: An approach based on dynamical systems. The VLDB Journal **8**, 222–236 (2000)
13. Hu, L., Jiang, M., Liu, X., He, Z.: Significance-based decision tree for interpretable categorical data clustering. Information Sciences **690**, 121588 (2025)
14. Ienco, D., Pensa, R.G., Meo, R.: From context to distance: Learning dissimilarity for categorical data clustering. ACM Transactions on Knowledge Discovery from Data **6**(1), 1–25 (2012)
15. Jian, S., Cao, L., Pang, G., Lu, K., Gao, H.: Embedding-based representation of categorical data by hierarchical value coupling learning. In: Proceedings of the 26th

International Joint Conference on Artificial Intelligence. p. 1937–1943. IJCAI'17, AAAI Press (2017)
16. Le, S.Q., Ho, T.B.: An association-based dissimilarity measure for categorical data. Pattern Recognition Letters **26**(16), 2549–2557 (2005)
17. Lee, Y., Park, C., Kang, S.: Deep embedded clustering framework for mixed data. IEEE Access **11**, 33–40 (2022)
18. Li, T., Ma, S., Ogihara, M.: Entropy-based criterion in categorical clustering. In: Proceedings of the twenty-first international conference on Machine learning. p. 68 (2004)
19. Liang, T., Wu, X., Xu, J., Feng, Q.: The distributed algorithms for the lower-bounded k-center clustering in metric space. Theoretical Computer Science **1027**, 114975 (2025)
20. Lin, D.: An information-theoretic definition of similarity. In: Proceedings of the 15th International Conference on Machine Learning. p. 296–304. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1998)
21. Luo, X., Zhang, Y., Ji, Y., Liu, P., Xiao, T.: Efficient topology-driven clustering for imbalanced streaming biomedical data analysis. In: 2024 IEEE International Conference on Bioinformatics and Biomedicine (BIBM). pp. 2262–2267. IEEE (2024)
22. Nguyen, T.H.T., Dinh, D.T., Sriboonchitta, S., Huynh, V.N.: A method for k-means-like clustering of categorical data. Journal of Ambient Intelligence and Humanized Computing **14**(11), 15011–15021 (2023)
23. Qian, Y., Li, F., Liang, J., Liu, B., Dang, C.: Space structure and clustering of categorical data. IEEE Transactions on Neural Networks and Learning Systems **27**(10), 2047–2059 (2015)
24. Santos, J.M., Embrechts, M.: On the use of the adjusted rand index as a metric for evaluating supervised classification. In: Proceedings of the International Conference on Artificial Neural Networks. pp. 175–184. Springer (2009)
25. Santos-Mangudo, C., Heras, A.J.: A fair-multicluster approach to clustering of categorical data. Central European Journal of Operations Research **31**(2), 583–604 (2023)
26. Wang, P., Zhang, Y., Zhang, Y., Lu, Y., Li, M., Cheung, Y.m.: Clustering by learning the ordinal relationships of qualitative attribute values. In: 2024 International Joint Conference on Neural Networks (IJCNN). pp. 1–8. IEEE (2024)
27. Zhang, Y., Cheung, Y.M.: An ordinal data clustering algorithm with automated distance learning. In: Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI). vol. 34, pp. 6869–6876 (2020)
28. Zhang, Y., Cheung, Y.M.: Graph-based dissimilarity measurement for cluster analysis of any-type-attributed data. IEEE Transactions on Neural Networks and Learning Systems (2022)
29. Zhang, Y., Cheung, Y.m.: Learnable weighting of intra-attribute distances for categorical data clustering with nominal and ordinal attributes. IEEE Transactions on Pattern Analysis and Machine Intelligence **44**(7), 3560–3576 (2022)
30. Zhang, Y., Cheung, Y.M.: A new distance metric exploiting heterogeneous interattribute relationship for ordinal-and-nominal-attribute data clustering. IEEE Transactions on Cybernetics **52**(2), 758–771 (2022). https://doi.org/10.1109/TCYB.2020.2983073
31. Zhang, Y., Cheung, Y.M., Tan, K.C.: A unified entropy-based distance metric for ordinal-and-nominal-attribute data clustering. IEEE Transactions on Neural Networks and Learning Systems **31**(1), 39–52 (2020). https://doi.org/10.1109/TNNLS.2019.2899381

32. Zhang, Y., Cheung, Y.m., Zeng, A.: Het2hom: representation of heterogeneous attributes into homogeneous concept spaces for categorical-and-numerical-attribute data clustering. In: Proceedings of the 31st International Joint Conference on Artificial Intelligence. pp. 1–8 (2022)
33. Zhang, Y., Zhao, M., Chen, Y., Lu, Y., Cheung, Y.m.: Learning unified distance metric for heterogeneous attribute data clustering. Expert Systems with Applications p. 126738 (2025)
34. Zhang, Y., Luo, X., Chen, Q., Zou, R., Zhang, Y., Cheung, Y.m.: Towards unbiased minimal cluster analysis of categorical-and-numerical attribute data. In: International Conference on Pattern Recognition. pp. 254–269. Springer (2024)
35. Zhang, Y., Zhang, Y., Lu, Y., Li, M., Chen, X., Cheung, Y.m.: Asynchronous federated clustering with unknown number of clusters. arXiv preprint arXiv:2412.20341 (2024)
36. Zhang, Y., Zou, R., Zhang, Y., Zhang, Y., Cheung, Y.m., Li, K.: Adaptive micro partition and hierarchical merging for accurate mixed data clustering. Complex & Intelligent Systems **11**(1), 1–14 (2025)
37. Zhao, M., Feng, S., Zhang, Y., Li, M., Lu, Y., Cheung, Y.M.: Learning order forest for qualitative-attribute data clustering. In: ECAI 2024, pp. 1943–1950. IOS Press (2024)
38. Zhu, C., Cao, L., Yin, J.: Unsupervised heterogeneous coupling learning for categorical representation. IEEE Transactions on Pattern Analysis and Machine Intelligence **44**(1), 533–549 (2022). https://doi.org/10.1109/TPAMI.2020.3010953
39. Zhu, C., Zhang, Q., Cao, L., Abrahamyan, A.: Mix2vec: Unsupervised mixed data representation. In: Proceedings of the 7th International Conference on Data Science and Advanced Analytics. pp. 118–127 (2020). https://doi.org/10.1109/DSAA49011.2020.00024
40. Zou, R., Zhang, Y., Zhang, Y., Lu, Y., Li, M., Cheung, Y.m.: Federated clustering with unknown number of clusters. In: 2024 6th International Conference on Data-driven Optimization of Complex Systems (DOCS). pp. 671–677. IEEE (2024)

2024 6th International Conference on Data-driven Optimization of Complex Systems (DOCS)

Federated Clustering with Unknown Number of Clusters

[redacted] Yunfan Zhang[2,b], [redacted]

2024 6th International Conference on Data-driven Optimization of Complex Systems (DOCS) | 979-8-3503-7784-2/24/$31.00 ©2024 IEEE | DOI: 10.1109/DOCS63458.2024.10704350

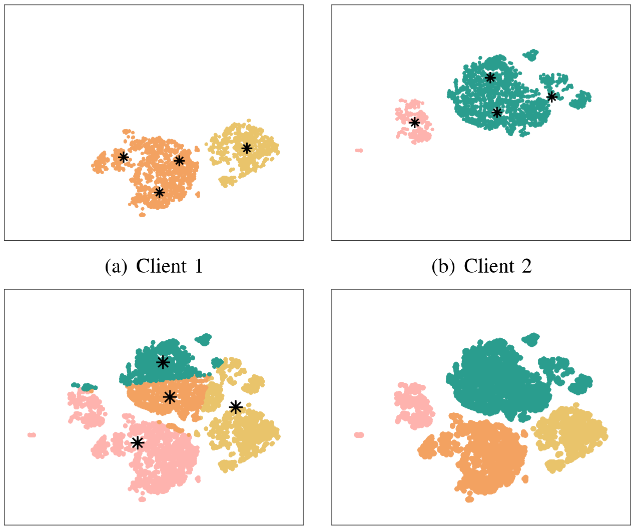
***Abstract*—Federated clustering is crucial to mining knowledge from unlabeled data distributed to multiple clients while preserving privacy. As there is no explicit learning supervision, clustering is considered a challenging federated learning task. Most existing works assume that the 'true' cluster number k^* is given to each client and server, which is far from a real federated learning scenario. Without the guidance of k^*, federated clustering becomes more challenging, rendering most existing solutions infeasible. We therefore propose a Federated Competitive and Cooperative Learning mechanism (FedCCL) to explore and fuse heterogeneous cluster distributions from clients automatically, and eventually form a global cluster partition, without requiring the cluster number to be given. We let the clients download seed points to explore their local distributions, which are then uploaded to the server for fusion. Different clients are allowed to compete on a single seed to form a consensus, while close seeds cooperate to represent a cluster. By iteratively homogenizing the cooperated seeds, a proper number of clusters will gradually emerge. Extensive experiments demonstrate the effectiveness of the proposed method.**

Index Terms*—*Federated Clustering, Competitive and Cooperative Learning, Unknown Number of Clusters

I. Introduction

Federated learning aims to realize machine learning under constraints of privacy and security [1], [2], [3]. In unsupervised federated learning tasks, clustering that partitions a dataset into compact object clusters demonstrates great potential in mining data knowledge [4], [5]. However, the settings of federated learning bring great challenges to clustering, as labels are unavailable to explicitly guide the learning process. Most existing federated clustering attempts assume that the cluster number to seek is known in advance, and can be roughly categorized into one-shot [6] and iterative approaches based on the communication frequency between client and server [7].

One-shot federated clustering learns cluster distributions locally and passes the learned knowledge to the server for global cluster distribution aggregation. k-FED [8] adopts such a paradigm to explore more comprehensive global cluster distributions through one-shot aggregation of the non-Independent and Identically Distributed (non-IID) distributions learned by the clients. However, k-FED assumes that the proper numbers of clusters for each client are given, and different clients are with clearly separable cluster distributions that can be easily learned by the k-means-type algorithms [9], [10]. In general, since the aggregation is performed in only one shot, it cannot provide sufficient opportunity for the clients to interact and complement each other for more comprehensive cluster distribution exploration.



(a) Client 1 (b) Client 2

(c) Server (d) Benchmark

Fig. 1. Direct use of existing iterative federated clustering methods on non-IID data. Even with a 'true' k^*, unexpected clustering results still easily occur.

Iterative federated clustering approaches facilitate sufficient interaction among the clients through the server by iteratively performing the following steps: (1) implement cluster distribution learning at the clients, (2) fuse privacy-protected distribution information at the server, and (3) send back the fused information to the clients for further tuning. To achieve a more comprehensive aggregation at the server, two independent works called F-FCM [11] and FFCM [12] sharing similar principles adopt fuzzy-c-means as the base clustering algorithm. The fuzzy object-cluster affiliation can

*Corresponding author

more finely reflect the partition information of data objects, which can somewhat offset the information loss due to the privacy constraints of federated learning.

Nevertheless, all the above federated clustering approaches assume the cluster number to be given to all the clients and the server in advance, severely constraining their applicability. As shown in Fig. 1, direct use of the relatively powerful iterative FFCM by setting the cluster number of all the clients and server to the global k^* may easily cause unexpected clustering results due to the non-IID of clients.

To relieve the dependence on a given k^*, we propose a new federated clustering approach called Federated Competitive and Cooperative Learning (FedCCL). It can automatically explore the cluster distributions without knowing the number of clusters. To leverage the merits of Competitive and Cooperative Learning (CCL) [13], [14] to automatically select k^* in a federated setting, we propose an asynchronous cluster centroid interactive learning mechanism. It accumulates the update intensity of each seed point within different clients, and then passes to the server for competitive client-to-seed information fusion. To address the thorny case that a global cluster is composed of several sub-clusters from different clients, we let neighboring seed points share their update intensity to achieve a cooperative seed-to-seed information fusion. As a result, the representative seeds gradually absorb the surrounding ones until they are duplicated, i.e., trapped by the corresponding clusters. Extensive experimental evaluations demonstrate the effectiveness of FedCCL. The main contributions of this work are summarized into three points:

- We propose a new federated clustering approach that does not require the 'true' cluster number, and thus enhances the universality of current federated clustering.
- This paper attempts to address a realistic but challenging non-IID case, i.e., a global cluster is composed of non-overlapping sub-clusters from different clients.
- An asynchronous competitive cooperative learning mechanism is proposed, which facilitates sufficient interactive learning of non-IID clients under federated scenarios.

II. Related Work

Federated Clustering: As data privacy issues are rarely considered and the increasing harm caused by data security problems, federated clustering has gained increasing attention for its role in protecting personal data. Dennis [8] developed a one-shot federated clustering scheme, k-FED, which utilized heterogeneity between clients under the center separation hypothesis and thus weakened the cluster separation requirements for k-FED. However, it uses Floyd's k-means in each client under the assumption that k-FED knows the cluster number from every client in advance. Most recently, a density-based method HFDPC [15] introduces a similar density chain to mitigate the "domino effect" caused by multiple local peaks in a flow pattern dataset. However, its efficiency is non-ideal, as it further utilizes data dimension reduction and image encryption for partitioning. As far as we know, most existing federated clustering methods heavily rely on the 'true' cluster number k^*, which hinders their application.

Clustering with Unknown Cluster Number: For most clustering methods, the cluster number k is a crucial parameter. Different strategies have been proposed to implement clustering with unknown k. Many efforts have been made to develop approaches without using k, e.g., the rival penalization controlled competitive learning clustering (RPCCL) [16] approach and the competitive and cooperative learning approach [13]. Recently, a method called PCL-OC [17] has further extended the automatic cluster number selection to complex mixed data scenarios. They all require the frequent interaction among seed points representing clusters, to let the redundant seeds to be eliminated. However, the federated condition that restricts the direct interaction among clients brings great challenges to the automatic selection of k.

III. Proposed Method

We first define the research problem and then present the proposed algorithm as two parts: 1) ClientUA: Client-side Update Accumulation, and 2) ServerSI: Server-side Seeds Interaction. The overall pipeline is demonstrated in Fig. 2.

Assume a dataset $X = \{X^{[1]}, X^{[2]}, ..., X^{[g]}, ..., X^{[p]}\}$ is composed of data from p different clients, where client data $X^{[g]} = \{x_1^{[g]}, x_2^{[g]}, ..., x_{n^{[g]}}^{[g]}\}$ has $n^{[g]}$ samples and sample $x_i^{[g]} = \{x_{i,1}^{[g]}, x_{i,2}^{[g]}, ..., x_{i,d}^{[g]}\}$ has d feature values, $i \in \{1, 2, ..., n^{[g]}\}$. Initially, each client employs a conventional clustering method, e.g. k-means++[18], to initialize k centroids, where k can be set as twice the value of k^* ('true' cluster number of the dataset X) or a relatively large value if the k^* is unattainable, thereby ensuring that the set k surpasses k^* and is therefore equal or greater than the 'true' cluster number for all clients. Subsequently, the centroids are transmitted to the server, and the $p * k$ centroids from p clients will be assigned to k global centroids $w_1, ..., w_k$ through the conventional clustering method, and then disseminated to the p clients.

ClientUA: To facilitate the fusion of non-IID information from different clients with unknown cluster numbers, update intensity $R^{[g]}$ is introduced to represent the sample distribution during client-side competitive learning. Additionally, we also need to compute the sample mean $b^{[g]}$, the average distance $z^{[g]}$, and the corresponding label $Q^{[g]}$, which are utilized as convergence information to support the calculation of convergence judgement function Z on the server.

For illustrative purposes, we take an arbitrary client g as an example. Assume client g has $n^{[g]}$ samples, k collections of sample label $Q_1, ..., Q_l, ..., Q_k$ and the corresponding k centroids $w_1, ..., w_l, ..., w_k$. However, since the data across each client is non-IID, the 'true' cluster number is less than or equal to the global set k, i.e., $k^{[g]*} \leq k$. For clarity, during the client-server iteration phase, $w_1, ..., w_k$ represent the global centroids received from the server, which remains unchanged throughout the operation process on the client side.

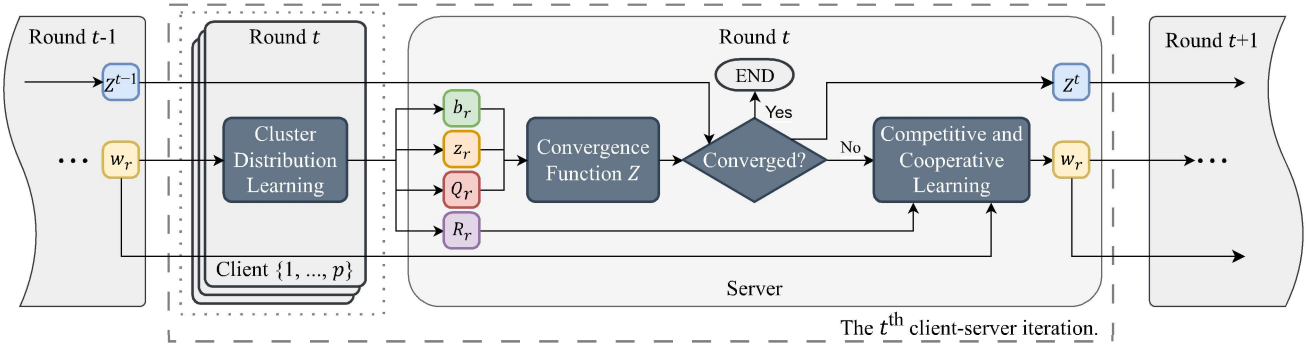


Fig. 2. Overview of the proposed FedCCL Algorithm.

The cluster to which the sample $x_i^{[g]}$ on client g belongs will be determined based on the indicator function

$$P(l \mid x_i^{[g]}) = \begin{cases} 1, & \text{if } l = arg\min_g \gamma_r \parallel x_i^{[g]} - w_r^{[g]} \parallel^2 \\ 0, & \text{otherwise} \end{cases}, \quad (1)$$

where $w_r^{[g]}$ is the r-th centroids in client g. That is, $x_i^{[g]}$ is classified into the l-th cluster ($Q_r^{[g]} = x_i^{[g]} \rightarrow Q_r^{[g]}$) if $P(l \mid x_i^{[g]}) = 1$, where $i \in \{1, 2, ..., n^{[g]}\}$. $\gamma_r^{[g]}$ is the associated win count:

$$\gamma_r^{[g]} = \frac{s_r^{[g]}}{\sum_{j=1}^{k^{[g]}} s_j^{[g]}}, \quad (2)$$

which represents the weight of $w_r^{[g]}$ under the sum of the win count of all centroids. $s_r^{[g]}$ denotes the cumulative win count of $w_r^{[g]}$ within a single iteration process. For the winner $w_c^{[g]}$, i.e. $P(c \mid x_i^{[g]}) = 1$, both win count $s_c^{[g]}$ and update intensity $R_{c,i}^{[g]}$ will be updated by

$$s_c^{[g]new} = s_c^{[g]old} + 1, \quad (3)$$

$$R_{c,i}^{[g]new} = R_{c,i}^{[g]old} + \eta(x_i^{[g]} - w_c^{[g]}), \quad (4)$$

while centroid $w_c^{[g]}$ remains unchanged. η is the learning rate.

Upon traversing all samples, we need to calculate convergence information. Nonetheless, given that the data across each client is non-IID within a federated environment, there exists a possibility that some clients may only have one single cluster. Specifically, the 'true' cluster number for client g, denoted as $k^{[g]*}$, equals 1. Under such circumstances, the computed result is meaningless as all samples within the client should be classified to the same unique centroid. Consequently, it is imperative to calculate the intermediate value of the convergence information for each client and transmit it to the server for the final convergence function. To this end, as one part of convergence information, the sample mean $b_r^{[g]}$ and the average distance $z_r^{[g]}$ from $b_r^{[g]}$ to the corresponding samples can be derived by

$$b_r^{[g]} = \sum_{i=1}^{|Q_r|} x_i^{[g]} / |Q_r^{[g]}|, \quad z_r^{[g]} = \sum_{i=1}^{|Q_r|} \parallel b_r^{[g]} - x_i^{[g]} \parallel^2, \quad (5)$$

where $x_i^{[g]} \in Q_r^{[g]}$.

ServerSI: The server primarily handles the aggregation of data received from clients, convergence function computation, and competitive and cooperative learning for global centroids, aiming to achieve a cooperative seed-to-seed information fusion. Upon receiving the sample mean b_r and the average distance z_r from each client, the server aggregates them by

$$b_r = \sum_{g=1}^{p} \frac{|Q_r^{[g]}| b_r^{[g]}}{\sum_{g=1}^{p} |Q_r^{[g]}|}, \quad z_r = \sum_{g=1}^{p} \frac{|Q_r^{[g]}| z_r^{[g]}}{\sum_{g=1}^{p} |Q_r^{[g]}|}. \quad (6)$$

Then the value of Z to judge the convergence is computed by

$$Z = \frac{1}{k} \sum_{i,j=1}^{k} \min_{i \neq j} D_{i,j}, \quad D_{i,j} = \frac{z_i + z_j}{\sqrt{\parallel b_i - b_j \parallel^2}}, \quad (7)$$

where $D_{i,j}$ is the dissimilarity between cluster Q_i and Q_j. The convergence of the proposed FedCCL algorithm is judged according to the change of Z. When the change of Z falls below a predefined small threshold ε, i.e. $Z^t - Z^{t-1} < \varepsilon$, FedCCL is judged to be convergence. Should the results fail to converge, the server will perform Competitive and Cooperative Learning with the received results and update intensity. To specific, for winner w_r based on update intensity r_r, the collaborator C_r can be calculated by

$$C_r = C_r \cup \{w_j \mid \parallel w_r - w_j \parallel \leq \parallel w_r - (r_r/\eta + w_r) \parallel\}, \quad (8)$$

where $r_r \in R_r$. Collaborator C_r comprises all centroids, including w_r, that are in proximity to w_r. As delineated in Eq. (8), the distance requirement is contingent upon r_r. A larger r_r value implies that the centroid w_r has a broad sample distribution range across all clients, suggesting that the collaborator centroids need to be identified within a larger radius for better cooperation. The positions of all centroids from collaborator C_r will be moved accordingly based on server-side sample distribution which is computed from the update intensity r_i of w_r by

$$\begin{aligned} w_u &= w_u + \eta((r_i/\eta + w_r^{old}) - w_u) \\ &= w_u + r_i + \eta w_r^{old} - \eta w_u, \end{aligned} \quad (9)$$

Algorithm 1 FedCCL

Input: Initial cluster number k with $k \gg k^*$, Dataset $X^{[1]}$, $X^{[2]}, ..., X^{[p]}$ for p clients;
Output: The global centroids $w_1, ...w_{k_*}$.

for each client $g \in \{1, ..., p\}$ *in parallel* **do**
  Client g**:** Initialize k centroids $w_1^{[g]}, ..., w_k^{[g]}$;
end for
Transfer initialization result to server;
Server: initialize k global centroids $w_1, ..., w_k$;
Initialize $Z^t \leftarrow 0$;
repeat
  Transfer global centroids $w_1, ..., w_k$ to all p clients;
  for each client $g \in \{1, ..., p\}$ *in parallel* **do**
    Client g**:** execute **ClientUA**;
  end for
  Transfer algorithm output to server;
  Update $Z^{t-1} \leftarrow Z^t$;
  Server: execute **ServerSI**;
until convergence

where $w_u \in C_r$ and $r_i \in R_r$. When all centroids and their collaborators are located in new positions, the server sends the centroids $w_1, ..., w_k$ back to each of the clients, and the algorithm returns to the ClientUA part to start a new iteration.

The design of Eq. (8) and (9) aims to enhance the adaptability of our proposed federated clustering method to various sample distributions. It also serves to prevent the occurrence of dead seeds. Considering the collaborators C_r, for w_r, Eq. (9) is transformed to $w_r = w_r + r_i + \eta w_r^{old} - \eta w_r$. That is, the movement of w_r is dominated by $r_i \in R_r$. For centroids other than w_r, the movement depends on both the update intensity and the position of w_r^{old}. In other words, all centroids in C_r will also tend to move closer to w_r when they transit to new positions according to the updated intensity. During the operation of traversing all centroids and their collaborators, neighboring centroids share their update intensity and achieve cooperative seed-to-seed information fusion. During the client-server iterative learning, the representative centroids will gradually absorb the surrounding less-representative ones until they are duplicated and trapped by the corresponding cluster distributions.

Overall FedCCL Algorithm: In FedCCL, the client-side mainly implements cluster distribution learning, and the server-side is responsible for privacy-protected distribution information fusion. The entire process is summarized in Algorithm 1. The time complexity of each iteration of FedCCL is $\mathcal{O}(kn^{[g]}dp + n^{[g]}k^2d)$, which is linear w.r.t. n.

IV. Experiments

Experimental Setup: Four experiments have been conducted to evaluate the proposed FedCCL: (1) Visualization, (2) Convergence Evaluation, (3) Clustering Performance Evaluation, and (4) Ablation Study.

Five counterparts have been compared. Federated Mean Shift (FMS) is a simple baseline federated clustering approach realized by replacing the fuzzy c-means [19] with the conventional mean shift clustering algorithm [20] for the FFCM [12]. DK++ [21] is a conventional distributed learning approach. Since it also conforms to the settings of federated learning, we also adopt it as a counterpart. Three state-of-the-art methods, i.e., the iterative learning approaches FFCM-avg1, FFCM-avg2 [12], and the one-shot learning approach k-FED [8], are also chosen for comparison. Hyper-parameters of the counterparts (if any) are set according to the corresponding source papers.

Thirteen datasets including two synthetic and eleven real benchmark datasets have been utilized. Statistics of the datasets are shown in Table I. All the public datasets are collected from the UCI machine learning repository [22]. All the datasets are pre-processed by omitting the objects with missing values. Two 2-D synthetic datasets are intuitively demonstrated in Fig. 3 using t-SNE [23], and objects belonging to the same true cluster are marked by the same color.

TABLE I
Statistics of Experimental Datasets.

No.	Dataset	Abbrev.	n	d	k^*
1	Synthetic Dataset 1	SD1	2300	2	4
2	Synthetic Dataset 2	SD2	2900	2	5
3	Drug Consumption	DC	1885	12	7
4	Avila	AL	10430	10	12
5	Abalone	AB	4178	8	29
6	Cancer	CC	570	31	2
7	Ecoli	EC	336	7	8
8	Seeds	SE	210	7	3
9	Parkinson	PA	195	22	2
10	Accent	AC	330	12	6
11	Sports Articles	SP	1000	59	2
12	Iris	IR	150	4	3
13	Segment	SG	2100	19	7

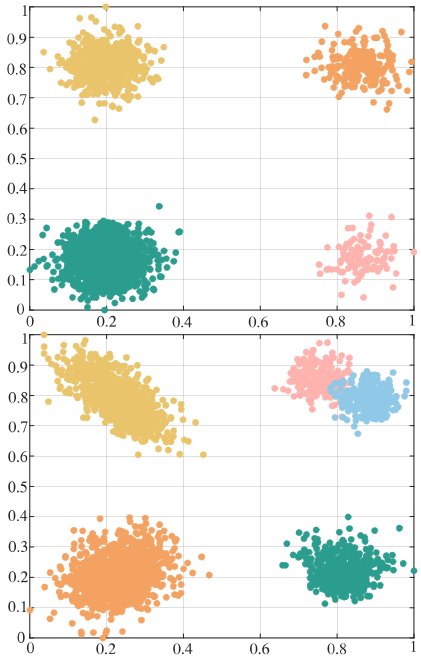
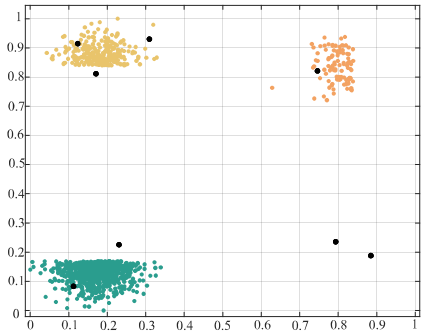
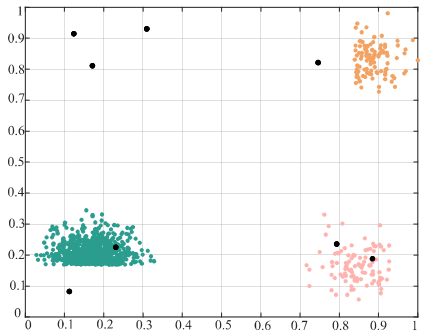


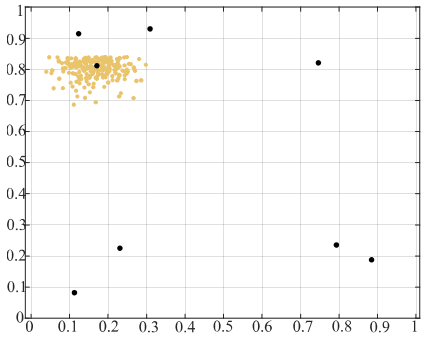
Fig. 3. Visualization of SD1 (upper) and SD2 (lower).



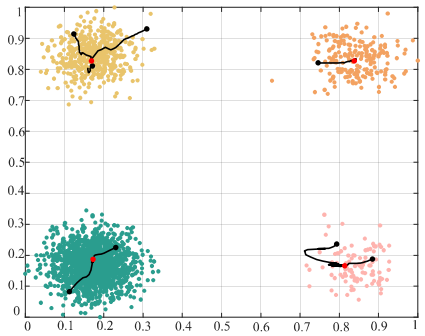
(a) Client 1



(b) Client 2



(c) Client 3



(d) Clustering Process on Server

Fig. 4. Cluster centroids and the trajectories during the learning of FedCCL. Black and red dots indicate the initial positions and final positions of the centroids, respectively.

Two validity indices have been chosen. Silhouette Coefficient index (SC) [24] is a conventional and popular internal index, which simultaneously indicates the compactness of clusters and the dispersion among clusters, with its values in [-1,1]. Calinski-Harabasz index (CH) [25] computes the ratio of the average inter-centroid distance to the average object-centroid distance within clusters, with values ranging from $(0,+\infty)$. For both of them, a higher value indicates a better clustering performance.

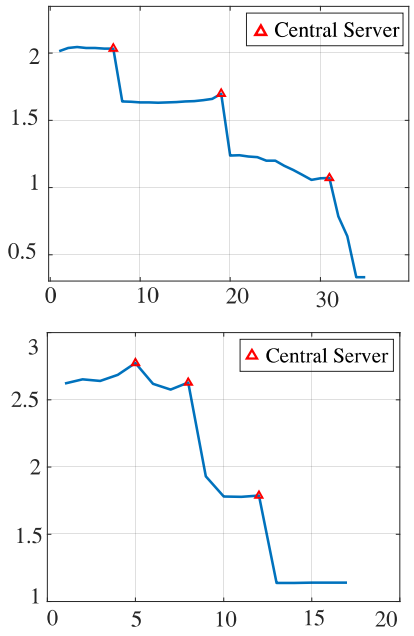


Fig. 5. Values of the FedCCL objective function on the SD1 dataset (upper) and DC dataset (lower). The red triangles mark the iterations of the server update.

TABLE II
CLUSTERING PERFORMANCE OF FEDCCL AND W/O SERVERSI.

Dataset	FedCCL	w/o ServerSI	Dataset	FedCCL	w/o ServerSI
SD1	19670.1	**20556.3**	SE	**254.5**	234.5
SD2	**19781.1**	18415.4	PA	**65.0**	64.6
DC	**831.8**	519.1	AC	**156.3**	89.3
AL	75.9	**1651.1**	SP	**455.3**	356.3
AB	**7910.1**	2997.4	IR	**330.4**	148.0
CC	**315.9**	289.0	SG	**1620.9**	949.3
EC	113.2	**116.4**			

Visualization: To intuitively validate the effectiveness of FedCCL, we split SD1 into three subsets for creating non-IID data for three clients as shown in Fig. 4(a)∼(c). Fig. 4(d) shows the distributions and trajectories of the centroids in the server. It can be observed that even though the three clients have completely non-overlapping distributions, FedCCL can still obtain the global cluster distributions through federated clustering. The trajectories also demonstrate that the asynchronous seed point updating mechanism of FedCCL effectively facilitates interaction among the seeds. After several iterations, redundant seeds are overlapped, indicating the same prominent global

TABLE III
COMPARISON OF SC PERFORMANCE ON 13 DATASETS.

Dataset	FedCCL	FFCM-avg1	FFCM-avg2	k-FED	DK++	FMS
SD1	**0.9718±0.0000**	0.5063±0.0241	0.5036±0.0215	0.8494±0.0000	0.5986±0.1798	0.8204±0.0000
SD2	**0.8766±0.0359**	0.4773±0.0261	0.4679±0.0324	0.7699±0.0000	0.6127±0.1046	0.7455±0.0423
DC	**0.5349±0.0000**	0.1675±0.1254	0.1476±0.1792	0.2104±0.0307	0.2884±0.0168	0.1812±0.0521
AL	**0.9750±0.0000**	0.2721±0.0921	0.2761±0.0701	0.0979±0.0266	0.2096±0.0194	0.4056±0.0142
AB	**0.5139±0.0228**	0.3664±0.3380	0.4863±0.2576	0.1853±0.0087	0.2314±0.0162	-0.2242±0.1359
CC	**0.5963±0.0664**	0.2873±0.0509	0.3051±0.0440	0.3809±0.0203	0.3778±0.0000	0.4624±0.0762
EC	**0.4739±0.0806**	0.3442±0.4041	0.4500±0.3016	0.3246±0.0073	0.2852±0.0370	0.3537±0.0468
SE	**0.5518±0.0399**	0.4321±0.0753	0.4321±0.0753	0.3754±0.0358	0.3229±0.0000	0.3605±0.0217
PA	**0.6284±0.1702**	0.4419±0.1549	0.4419±0.1549	0.4517±0.0328	0.2763±0.0000	0.6016±0.0392
AC	**0.6385±0.1369**	0.2656±0.1390	0.2358±0.1346	0.0992±0.0084	0.1831±0.0226	0.0777±0.0425
SP	0.5466±0.0036	0.2800±0.0516	0.2796±0.0507	0.5194±0.0400	0.3441±0.0028	**0.5844±0.0573**
IR	**0.6579±0.0181**	0.5672±0.0611	0.6119±0.2075	0.4818±0.0198	0.4955±0.0107	0.4487±0.0148
SG	**0.5702±0.0130**	0.3819±0.1113	0.3865±0.1020	0.3117±0.0017	0.3197±0.0183	0.3556±0.0379
Ave. Rank	**1.0769**	4.0000	3.8462	4.0000	4.4615	3.6154

cluster, thus intuitively demonstrating the autonomous k selection ability of FedCCL.

Convergence Evaluation: To evaluate the efficiency in convergence of FedCCL, we plot the values of the objective function of FedCCL on two datasets in Fig. 5. It can be observed that FedCCL converges quickly within 50 iterations in most cases. Moreover, the objective function always experiences a steep decline after the server updates, confirming that the designed server seeds interaction mechanism is highly effective. It is also noteworthy that, since only limited statistics are permitted to be communicated between clients and server, and the data distribution varies on the clients and server, the convergence curve in Fig. 5 is not monotonically decreasing. Such an effect is reasonable because the learning objective can be viewed as heterogeneous at different clients and the server.

Clustering Performance Evaluation: To further validate the effectiveness of FedCCL, we have also compared it with the existing approaches in the challenging non-IID scenario. Specifically, we generate 20 different sets of data distributions for clients using k-means for each dataset. The number of clients is uniformly set to 5 in this comparison and also in the following ablation study experiments. As FedCCL does not require the 'true' cluster number k^*, we randomly select the k from the range $[k^*, 2k^*]$ as the initial cluster number. The performance in terms of SC index is shown in Table III. Due to space limitation, results w.r.t. CH index is omitted here. The best and the second-best results are highlighted using boldface and underline, respectively. The 'Ave. Rank' row reports the average ranks of different counterparts across all datasets.

It can be observed from Table III that FedCCL outperforms the other counterparts in general, indicating its effectiveness. Specifically, FedCCL surpasses the counterparts on almost all the datasets, except for the SP dataset where it still achieves the second-best result. The above observations illustrate that the proposed method can effectively find the global optimal position of centroids by maximizing both the intra-cluster density and the inter-cluster dispersion.

Ablation Study: We conduct ablation study by using the discriminative CH index to validate the effectiveness of the core server competition process. Due to space limitation, results w.r.t. SC index is omitted here. The version of FedCCL without ServerSI (w/o ServerSI) is formed by replacing the ServerSI with the simple cluster centroid aggregation of FFCM. That is, the global centroid is computed as a weighted average of the centroids from clients, where the weight assigned to each client's centroid is proportional to the number of objects it represents. It can be observed from Table II that FedCCL outperforms w/o ServerSI in most cases. This intuitively illustrates the effectiveness of ServerSI, which can let the seed points sufficiently interact with the neighboring ones. As a result, the detailed local distribution information can be iteratively propagated across different seed points to collaboratively eliminate redundant seeds and sketch the global cluster distributions.

V. CONCLUSION

This paper has proposed a new federated clustering approach called FedCCL that can well mine global cluster distributions upon heterogeneous data distributions of clients. It advances federated clustering to a more challenging but realistic scenario, i.e., all the clients can be extremely non-IID and the 'true' number of clusters of the clients and the server are all unknown. More specifically, FedCCL assigns excessive seed points to the clients to outline their local distribution using aggregated update intensity of seed points received locally. To address the potential contradiction among seeds caused by the clients' heterogeneity, interactions across both clients and seeds have been facilitated to more comprehensively explore clusters. FedCCL is easy to use as it is robust to an easy-to-set learning rate. Comprehensive experiments have been conducted to illustrate its efficacy.

Despite the effectiveness of FedCCL, there are still some noteworthy limitations. That is, we assume static federated clustering on pure numerical data, and thus FedCCL is incompetent to asynchronous updates of clients. The next promising avenue would involve dynamic federated clustering of datasets described by a mixture of numerical and categorical attributes.

ACKNOWLEDGEMENTS

[redacted]

REFERENCES

[1] S. Banabilah, M. Aloqaily, E. Alsayed, N. Malik, and Y. Jararweh, "Federated learning review: Fundamentals, enabling technologies, and future applications," *Information Processing & Management*, vol. 59, no. 6, p. 103061, 2022.

[2] X. Yin, Y. Zhu, and J. Hu, "A Comprehensive Survey of Privacy-preserving Federated Learning," *ACM Computing Surveys*, vol. 54, no. 6, pp. 1–36, 2022.

[3] C. Zhang, Y. Xie, H. Bai, B. Yu, W. Li, and Y. Gao, "A survey on federated learning," *Knowledge-Based Systems*, p. 106775, 2021.

[4] E. Lubana, C. I. Tang, F. Kawsar, R. Dick, and A. Mathur, "Orchestra: Unsupervised Federated Learning via Globally Consistent Clustering," in *ICML*, vol. 162, 2022, pp. 14 461–14 484.

[5] A. Nelus, R. Glitza, and R. Martin, "Unsupervised Clustered Federated Learning in Complex Multi-source Acoustic Environments," in *EUSIPCO*, 2021, pp. 1115–1119.

[6] S. Xie, Y. Wu, K. Liao, L. Chen, C. Liu, H. Shen, M. Tang, and L. Sun, "Fed-SC: One-Shot Federated Subspace Clustering over High-Dimensional Data," in *ICDE*, 2023, pp. 2905–2918.

[7] A. Ghosh, J. Chung, D. Yin, and K. Ramchandran, "An Efficient Framework for Clustered Federated Learning," in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 19 586–19 597.

[8] D. K. Dennis, T. Li, and V. Smith, "Heterogeneity for the Win: One-Shot Federated Clustering," in *ICML*, ser. Proceedings of Machine Learning Research, vol. 139, 2021, pp. 2611–2620.

[9] A. Kumar and R. Kannan, "Clustering with Spectral Norm and the k-Means Algorithm," in *FOCS*, 2010, pp. 299–308.

[10] P. Awasthi and O. Sheffet, "Improved Spectral-Norm Bounds for Clustering," in *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, 2012, pp. 37–49.

[11] W. Pedrycz, "Federated FCM: Clustering Under Privacy Requirements," *IEEE Transactions on Fuzzy Systems*, pp. 3384–3388, 2022.

[12] S. Morris and W. Anna, "Towards Federated Clustering: A Federated Fuzzy c-Means Algorithm (FFCM)," 2022.

[13] Y.-M. Cheung, "A Competitive and Cooperative Learning Approach to Robust Data Clustering," in *Neural Networks and Computational Intelligence - 2004*, 2004, pp. 131–136.

[14] L.-T. Law and Y.-M. Cheung, "Color image segmentation using rival penalized controlled competitive learning," in *IJCNN*, 2003, pp. 108–112.

[15] S. Ding, C. Li, X. Xu, L. Guo, L. Ding, and X. Wu, "Horizontal Federated Density Peaks Clustering," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–10, 2023.

[16] Y.-M. Cheung, "Rival penalization controlled competitive learning for data clustering with unknown cluster number," in *ICONIP*, vol. 1, 2002, pp. 467–471.

[17] Y.-M. Cheung and J. Hong, "Categorical-and-numerical-attribute data clustering based on a unified similarity metric without knowing cluster number," *Pattern Recognition*, vol. 46, no. 8, pp. 2228–2238, 2013.

[18] D. Arthur and S. Vassilvitskii, "K-means++: The advantages of careful seeding," *SODA*, pp. 1027–1035, 2007.

[19] J. C. Bezdek, R. Ehrlich, and W. Full, "FCM: The fuzzy c-means clustering algorithm," *Computers & Geosciences*, vol. 10, no. 2, pp. 191–203, 1984.

[20] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE transactions on pattern analysis and machine intelligence*, vol. 17, no. 8, pp. 790–799, 1995.

[21] B. Bahmani, B. Moseley, A. Vattani, R. Kumar, and S. Vassilvitskii, "Scalable K-Means+," *VLDB*, vol. 5, no. 7, 2012.

[22] A. Asuncion and D. Newman, "UCI machine learning repository," 2007.

[23] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." *Journal of machine learning research*, vol. 9, no. 11, 2008.

[24] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Journal of computational and applied mathematics*, vol. 20, pp. 53–65, 1987.

[25] T. Caliński and J. Harabasz, "A dendrite method for cluster analysis," *Communications in Statistics-theory and Methods*, vol. 3, no. 1, pp. 1–27, 1974.

ANDI: ANy-type Attributed Data Imputation via Cluster-Guided Missing Value Inference

[redacted] Xinxi Chen, Zexi Tan, [redacted]

***Abstract*—Clustering algorithms are usually dependent on complete datasets and often face the problem of missing data. Missing values imputation provides a promising solution to this problem. However, most available imputation methods are restricted to one type of attribute: numerical or categorical. In particular, categorical attributes can also be subdivided into nominal type and ordinal type according to whether there is an ordering relationship between their values. For such heterogeneous data, few methods pay attention to the possible relationship between these three types, and usually treat different types separately and then combine. Moreover, most methods fill the missing values based on the observed complete objects and do not take into account the information brought by the observed incomplete objects. This leads to the loss of effective information and the impairment of reasoning ability. Therefore, we propose a method that can comprehensively exploit the data statistics and mine the relationship between heterogeneous attributes for more accurate missing value imputation in heterogeneous data. The effectiveness of the proposed method is comprehensively evaluated on various datasets.**

***Index Terms*—missing values imputation, clustering, heterogeneous attributes, dissimilarity measure, natural neighbor.**

I. Introduction

Handling missing values is a crucial step before cluster analysis. Most existing clustering methods require fully observed datasets without missing values. However, missing values are pervasive in real-world datasets, because primary data with missing values may be collected due to various accidents and errors, such as operational mistakes, equipment failures, artificial damage, etc.

The ongoing development of research techniques has higher requirements for datasets and brings challenges to data analysts due to the complex interactions and nonlinear relationships within datasets, which are difficult to measure, and the heterogeneous datasets, which often appear in various research fields and contain nominal, ordinal, and numerical attributes. When missing values occur, it's vital to handle them properly, as there is no universally best imputation method; it depends on the type of data at hand. Direct deletion approach[1] (omits incomplete data and only uses the remaining data) and mean/mode substitution (filling missing values with the average values for numerical attributes and the most frequent value for categorical attributes)[2] are rapid and easy to implement, but both of them are not the best choice as they may miss valuable information and generally leads to serious biases. More popular approaches, like K-Nearest Neighbors-based Missing values Imputation (KNNMI)[3][4] and K-Means Clustering-based Missing values Imputation(KMCMI)[5][6] for numerical data, C4.5[7] and association rule[8] for categorical data. Imputation methods for heterogeneous data is rather race in the literature. It was first introduced by Rubin[9] with multiple imputation. Then an approach based on maximum likelihood estimation is proposed, which combining the multivariate normal model for continuous and the Poisson/multinomial model for categorical data[10]. A more refined method MICE[11] relies on tuning parameters or specifying a parametric model, missing values are repeatedly imputed to create a complete dataset. Another powerful MissForest[12] is based on Random Forest[13].

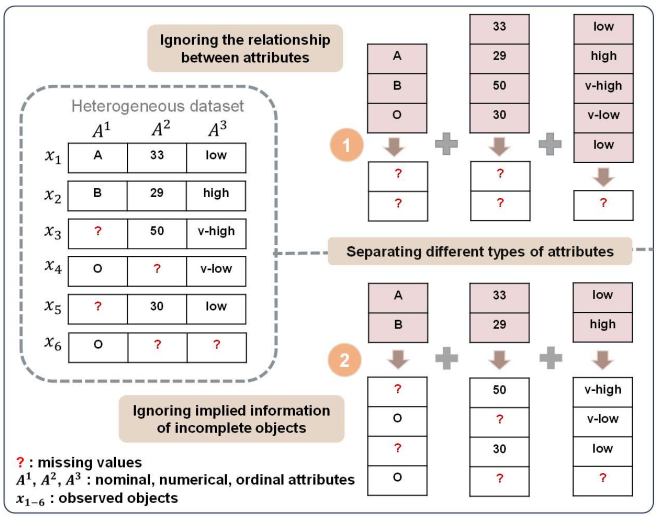


Fig. 1. Three unresolved challenges in existing imputation methods. The left side of the figure shows a heterogeneous dataset with only two complete objects, while the right side illustrates two main imputation modes. The first mode estimates the missing values in each column using the complete data from that column, the second mode estimates the missing values in incomplete objects using the complete objects.

In general, most existing imputation methods are limited to one type of attribute. In addition, these methods only fill the missing values based on the observed complete objects, ignoring the information contained in the observed incomplete objects. Furthermore, they often treat the attributes of samples as independent, overlooking the imputation information implied by the interdependence between different attributes. When the dataset contains different types of attributes and a large number of incomplete objects, the effectiveness of

TABLE I
EXPLANATIONS OF SYMBOLS.

Symbols	Explanations
x_i	Datasets objects
A^r	The rth attribute of an object x_i
o_m^r	The mth unique value of an attribute A^r
O^s	The unique value set of the attribute A^s
c_i	Cluster centroids
D	Attribute distance matrix
T and S	Incomplete objects set and the whole dataset
n and k	Number of objects and clusters

these methods is significantly reduced. Fig.1 provides a simple example to illustrate these three challenges.

This paper, therefore, proposes a new missing value imputation method named ANy-type Attributed Data Imputation (ANDI). We first define a dissimilarity measure for the heterogeneous data with missing values, then we present a clustering method for incomplete heterogeneous datasets. Finally, we propose to infer the missing values according to the natural neighbor relationship in the clusters. Comparison experimental results show the excellent performance of the proposed method.

The main contributions of this paper are three-fold:

- A general metric is defined for heterogeneous data with missing values, which can calculate the dissimilarity between complete data and incomplete data in heterogeneous data.
- A new imputation paradigm is proposed, which exploits the inter-dependencies between heterogeneous attributes and takes full advantage of observed incomplete data to guide more accurate missing values inference.
- A clustering method for incomplete heterogeneous data sets is proposed, which establishes a complementary interaction between imputation and clustering to achieve better performance.

II. PROPOSED METHOD

We propose a method to infer missing values by leveraging the interdependence between attributes and incomplete objects clustering results. Table. I lists the symbols used in the paper.

To utilize all available information, we cluster all objects using a distance matrix that includes distances between complete and incomplete objects. Here, we define a clustering process named Iterative Centroid Replacement Clustering (ICRC). We first randomly select k objects as the initial centroids and iteratively update the centroids until they no longer change. In each iteration, we try to replace the current centroid point with all the non-centroid points and calculate the replacement loss. The non-centroid point that brings the minimum loss is taken as the new centroid point. The replacement loss is computed as follows:

$$\text{Cost} = \sum_{i=1}^{n} (\Psi(x_i, c_j^*) - \Psi(x_i, c_j)), \tag{1}$$

Algorithm 1: Iterative Centroid Replacement Clustering

Input: Dataset S, number of clusters k, objects distance matrix DT.
Output: k clusters of S, k cluster centroids.

```
c_j ← k objects as initial centroids;
for object x_i in S do
    Assign x_i to the cluster represented by the nearest centroid based on DT;
end for
for each centroid c_j do
    for each unselected non-centroid object x_i do
        Calculate the Cost of replacing x_i with c_j ;
    end for
end for
if any Cost < 0
    c_j^* ← non-centroid object in H with the minimum Cost;
    Swap the corresponding centroid c_j with c_j^*;
end if
```

where x_i is the remaining non-centroid objects, c_j^* is the new centroid, c_j is the old centroid, and the function $\Psi(\cdot,\cdot)$ denotes the dissimilarity between two objects. Algorithm 1 gives a representation of the ICRC.

For missing value imputation, we put the objects with missing values in a set T and sort them by increasing number of missing values. Each time we take the first object from T as the target object for inference. And then the entire dataset, including incomplete objects, are involved in clustering. The target object's cluster is identified and denoted as target cluster. In the target cluster, we calculate the natural neighbors[14] of the target object, and use their values as the reference values for imputation. Depending on the attribute type, the mean/mode of reference values is chosen for numerical/nominal and ordinal attributes. For objects with multiple missing values, we prioritize attributes with the highest interdependence with complete attributes for imputation. After imputing a missing attribute, we recalculate the target object's cluster and impute the next missing attribute based on the new target cluster's natural neighbors. This process continues until all objects in T are processed. The final output is a complete dataset and k clusters. Algorithm 2 gives a representation of the ANDI method.

In the following, we will give a description of the three elements that will affect the reasoning results, which are the distance matrix of incomplete objects, interdependence degree between attributes, and natural neighbors of target object.

For the distance matrix calculation of incomplete objects, we first give the definition of the distance $\Psi(x_i, x_j)$ between two data objects x_i and x_j in the complete cases:

$$\Psi(x_i, x_j) = \sqrt{\textstyle\sum_{r=1}^{d} \Psi^r(x_i^r, x_j^r)^2}, \tag{2}$$

where the $\Psi^r(x_i^r, x_j^r)$ is the dissimilarity of two possible value x_i^r and x_j^r of object x_i and x_j in attribute A^r. When

Algorithm 2: ANy-type Attributed Data Imputation

Input: Dataset S, number of clusters k, set T of incomplete objects divided from S.
Output: Complete dataset S, k clusters of S.

$x_i \leftarrow$ objects of sorted indices in T, w.r.t. increasing amount of missing values;
while T is not empty **do**
  $DT \leftarrow$ store objects distance matrix of S using Eq.(7);
  k clusters, centroids = $ICRC(S, k, DT)$;
  $c_j \leftarrow$ cluster containing x_i;
  $NaN_{x_i} \leftarrow$ natural neighbors of x_i in c_j;
  for each missing attribute in x_i **do**;
    $A^r \leftarrow$ attribute with highest interdependence using Eq.(8);
    $x_i \leftarrow$ impute x_i^r using A^r of NaN_{x_i};
  end for
  Remove x_i from T, update x_i in S;
end while

missing values occur in x_i or x_j or both of them, we modify the distance definition. That is, we calculate this distance between the complete attribute values and then normalize to compensate for the missing values[15]. Suppose md is the number of attributes which are missing (in one object or the other or both), then the distance $M\Psi(x_i, x_j)$ is computed as follows:

$$M\Psi(x_i, x_j) = \sqrt{\sum_{r=1}^{d} \frac{d}{d - md} M\Psi^r(x_i^r, x_j^r)^2}, \tag{3}$$

where $M\Psi^r(x_i^r, x_j^r)$ is defined as

$$M\Psi^r(x_i^r, x_j^r) = \begin{cases} 0, & \text{if } x_i^r \text{ or } x_i^r \text{ is missing} \\ \Psi^r(x_i^r, x_j^r), & \text{otherwise.} \end{cases}$$

In other words, we assume that the distances between the missing attributes are equal to the average of the distances between the complete attributes. In this way, the bias induced by the traditional method (which only uses the distance between complete attributes as the distance) can be reduced.

Through the innovative design of graph-based unified dissimilarity (GUD)[16], we recognize that nominal, ordinal, and numerical attributes can be uniformly represented using graphs. Since numerical attributes can approximate ordinal attributes, indicating an order relationship between their possible values, this paper attempts to place numerical attributes in an ordinal attribute space, preserving the original order relationship within numerical attributes and calculating the dependence between numerical attributes and other attributes. The $\psi^{rs}(o_m^r, o_h^r)$ represents the dissimilarity between values o_m^r and o_h^r as reflected by A^s, is defined as

$$\psi^{rs}(o_m^r, o_h^r) = \sum_{g=1}^{v_{mh}^r - 1} |u_g^{rs} - u_t^{rs}| \cdot t_{gt}^{rs}, \tag{4}$$

where $t = g + 1$. u_g^{rs} and u_t^{rs} are the conditional probability distributions(CPD) of the values in O^s as given the g-th and t-th values in O_{mh}^r. The CPD difference $|u_g^{rs} - u_t^{rs}|$ describes the differences between the corresponding values of A_g^{rs} and A_t^{rs} that should be transported for offsetting in the graph transformation. Vector $t_{gt}^{rs} = [t_{gt1}^{rs}, t_{gt2}^{rs}, ..., t_{gtv^s}^{rs}]^T$ stores the minimum total edge lengths that should be taken to transport each of the differences. The CPD of the values in O^s as given o_m^r is defined as

$$u_m^{rs} = [p(o_1^s|o_m^r), p(o_2^s|o_m^r), ..., p(o_{v^s}^s|o_m^r)]^T, \tag{5}$$

$$p(o_g^s|o^r m) = \frac{\sigma(X_g^s \cap X_m^r)}{\sigma(X_m^r)}, \tag{6}$$

where $X_m^r = \{x_i | x_i^r = o_m^r, i = 1, 2, ..., n\}$ is a subset of X with the rth values of all its objects equal to o_m^r, and the function $\sigma(\cdot)$ counts the cardinality of a set.

Then we discuss the interdependence degree between attributes, which is used to guide the preferential selection of attributes for imputation. The distance $\Psi^r(o_m^r, o_h^r)$ is defined as

$$\Psi^r(o_m^r, o_h^r) = \sum_{s=1}^{d} \psi^{rs}(o_m^r, o_h^r) \cdot w^{rs}, \tag{7}$$

where the $\psi^{rs}(o_m^r, o_h^r)$ partially reflects the dependence of A^s on A^r and w^{rs} controls the contribution of A^s to $\Psi^r(o_m^r, o_h^r)$. If the dissimilarities between different values of A^r reflected by A^s are consistently higher than those reflected by other attributes, A^r and A^s are considered to have stronger interdependence[16]. The weight w^{rs} is defined as

$$w^{rs} = \frac{\sum_{q=1}^{v^{r<*>}-1} \sum_{c=q+1}^{v^{r<*>}} \psi^{rs}(o_q^r, o_c^r)}{\frac{v^{r<*>}(v^{r<*>}-1)}{2} \cdot (v_{qc}^r - 1)}, \tag{8}$$

where o_q^r and o_c^r are the q-th and c-th unique values in the set $O^{r<*>}$, and v_{qc}^r is the number of intermediate values on the shortest path between o_q^r and o_c^r. When multiple attributes in the target object have missing values, we designate the missing attribute as A^r and the complete attribute as A^s. We then impute the missing attribute with the highest interdependence with the complete attribute first, maximizing the likelihood of deriving the correct value based on the available information.

Finally, the natural neighbor is introduced. The proposed ANDI aims to find similar objects to the target object as reference. Based on Natural Neighbor (NaN) searching[14], if x_i belongs to the neighbors of x_j and x_j belongs to the neighbors of x_i, then x_i and x_j are natural neighbor of each other. NaN Searching does not require additional parameters, and each object can have a varying number of natural neighbors. Accordingly, by natural neighbors, we can identify the most qualified objects to provide information for the target object. The natural neighbor of the object X_i is defined as follows:

$$x_j \in NaN(x_i) \Leftrightarrow (x_i \in NN_r(x_j)) \wedge (x_j \in NN_r(x_i)), \tag{9}$$

TABLE II
DESCRIPTION OF DATASETS. $d^{<n>}$, $d^{<o>}$, $d^{<u>}$ INDICATE THE NUMBERS OF NOMINAL, ORDINAL AND NUMERICAL ATTRIBUTES, RESPECTIVELY. n AND k INDICATE THE NUMBERS OF OBJECTS AND CLUSTERS, RESPECTIVELY.

No.	Dataset	Abbrev.	$d^{<n>}$	$d^{<o>}$	$d^{<u>}$	n	k
1	Iris	IR	0	0	4	150	3
2	Wine	WI	0	0	13	178	3
3	Hayes Roth	HR	2	2	0	160	3
4	Zoo	ZO	15	1	0	101	7
5	Diagnosis	DS	5	0	1	120	2
6	Teacher Assistant	TA	4	0	1	151	3

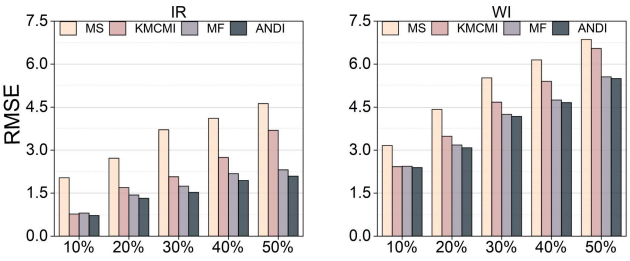


Fig. 2. NRMSE of the imputation methods on pure numerical datasets and five different rates of missing values.

where r is the number of search cycles when dataset S reaches a natural steady state. $NaN(x_i)$ is the set that contains the natural neighbors of object x_i, whereas $NN_r(x_i)$ is the set that contains r nearest neighbors of x_i.

The time complexity of the ANDI method can be divided into two parts: First, clustering all objects, which includes calculating the distance matrix $O(n^2)$ and initial clustering $O(Ink)$. Second, the imputation process, which includes sorting incomplete objects $O(m \log m)$, and iterative imputation $O(m(Ink+an_c \log n_c))$, where n_c is the number of objects in the target cluster. Combining these, the total time complexity is $O(n^2)+O(mInk)+O(man_c \log n_c)$, where n is the number of objects, m is the number of incomplete objects, k is the number of clusters, I is the number of iterations, and a is the number of missing attributes.

III. EXPERIMENTS

Comparison Methods. The ANDI is compared with four methods across 6 datasets, which include three types: pure numerical, pure categorical, and heterogeneous attributes. These datasets is shown in Table II. Mean/mode substitution (MS, denoted as MMS for heterogeneous datasets) is a simple method for imputing missing values in both numerical and categorical data. For numerical data, we also use K-Means Clustering-based Missing values Imputation (KMCMI)[5], which involves two steps: forming clusters with K-Means and using cluster information to handle missing values. For categorical or heterogeneous data, we use MissForest (MF)[12], which treats the missing data problem as a prediction problem, imputing missing values by regressing each variable against all others and predicting missing values using the fitted forest[17]. Another approach is K-Nearest Neighbors-based Missing values Imputation (KNNMI)[3], which substitutes missing values with the mean of the k nearest complete neighbors. And in our experiment the k neighbors is set to the square root of the number of observed complete objects suggested by Lall and Sharma[18]. Effective imputation methods enhance data quality and clustering performance, which in turn reflects the imputation method's quality. We will choose a classical clustering method based on the data type after imputation: K-Means[19] for numerical data, K-Modes[20] for categorical data, and K-prototypes[21] for heterogeneous data. Additionally, the number of clusters is set to the true number of label classes in the dataset.

TABLE III
CLUSTER PERFORMANCE ARI AND CVI ON IR.

Missing Rate	Indicator	MS	MF	KMCMI	ANDI
10%	ARI	0.659	0.656	0.615	0.728
	CVI	0.439	0.506	0.062	0.596
20%	ARI	0.645	0.684	0.655	0.709
	CVI	0.429	0.519	0.048	0.522
30%	ARI	0.522	0.663	0.667	0.730
	CVI	0.304	0.507	0.036	0.580
40%	ARI	0.404	0.622	0.602	0.652
	CVI	0.326	0.512	0.029	0.522
50%	ARI	0.346	0.611	0.434	0.679
	CVI	0.291	0.533	0.025	0.566

Evaluation and Settings. We refer to the complete data before generating missing values as the original data, and then we will set the missing rates as 10%, 20%, 30%, 40%, 50%, to remove the original data completely at random in each experiment. For each experiment we will repeat 10 times and take the average as the final result. To evaluate the imputation efficiency of different imputation methods for different types of missing value data, and the effect of clustering after imputation, we considered two different criteria. First, we assess the distance between the original and imputed data. For numerical data, we use root mean squared error (RMSE). For categorical data, we use the simple matching method for Accuracy. For heterogeneous data, we use modified RMSE (mRMSE)[22]. The specific formulas outlined below:

$$RMSE = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(\hat{x_i^r} - x_i^r)^2}, \tag{10}$$

$$Accuracy = \frac{\sum_{i=1}^{m} I(\hat{x_i^r}, x_i^r)}{m}, \tag{11}$$

where m is the number of missing values, $\hat{x_i^r}$ and and x_i^r are the value of the original data and imputed data at the i-th missing value position. $I(\cdot,\cdot)$ is an indicator function that takes the value 1 when $\hat{x_i^r} = x_i^r$, and 0 otherwise.

$$mRMSE = \sqrt{RMSE_0^2 + RMSE_1^2}, \tag{12}$$

TABLE IV
CLUSTER PERFORMANCE ARI AND CVI ON WI.

Missing Rate	Indicator	MS	MF	KMCMI	ANDI
10%	ARI	0.816	0.812	0.826	0.831
	CVI	0.260	0.300	0.003	0.306
20%	ARI	0.653	0.776	0.772	0.786
	CVI	0.221	0.318	0.002	0.322
30%	ARI	0.830	0.847	0.776	0.864
	CVI	0.222	0.363	0.002	0.371
40%	ARI	0.601	0.689	0.561	0.657
	CVI	0.156	0.342	0.001	0.382
50%	ARI	0.503	0.692	0.501	0.696
	CVI	0.138	0.349	0.001	0.363

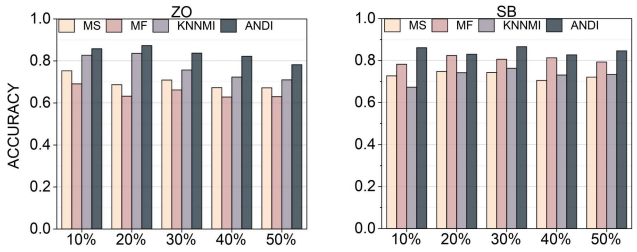


Fig. 3. Accuracy of the imputation method on pure categorical datasets and five different rates of missing values.

where $RMSE_0$ and $RMSE_1$ are $RMSE$s of categorical attributes M_0 ($d^{<n>}$ or $d^{<o>}$) and numerical attributes $M_1(d^{<u>})$, respectively and are defined in

$$RMSE_0 = \sqrt{\frac{1}{|M_0|}\sum_{A^r \in M_0} 1_{\hat{x_i^r} \neq x_i^r}}, \tag{13}$$

$$RMSE_1 = \sqrt{\frac{1}{|M_1|}\sum_{A^r \in M_1} (\hat{x_i^r} - x_i^r)^2}. \tag{14}$$

Second, we evaluate the clustering effect after imputation using the Adjusted Rand Index (ARI)[23] and the Silhouette index (CVI)[24].

Results. The best and the second-best clustering results on each dataset are highlighted using deep pink and light beige, respectively. For pure numerical datasets, the results are given in Fig. 2, Table III and Table IV. We can see that ANDI performs better than the other methods, reducing imputation error in many cases by more than 10%. All the methods bring higher clustering effects as the error of imputation decreases, which shows that correctly filling the dataset can greatly affect the clustering results. For pure categorical datasets, as shown in Fig. 3, we can see that ANDI consistently imputes missing values better than the compared methods, with the accuracy improvement ranging from 10% to 30%. Furthermore, as shown in Table V and Tabble VI, ANDI integrates imputation into the clustering process, resulting in excellent clustering efficiency. For heterogeneous datasets, the results are shown in Fig. 4, Table VII and Table VIII. We can see that ANDI performs well, sometimes reducing the average mRMSE by up

TABLE V
CLUSTER PERFORMANCE ARI AND CVI ON ZO.

Missing Rate	Indicator	MS	MF	KNNMI	ANDI
10%	ARI	0.635	0.633	0.625	0.875
	CVI	0.371	0.361	0.432	0.562
20%	ARI	0.528	0.507	0.589	0.804
	CVI	0.252	0.249	0.375	0.505
30%	ARI	0.456	0.527	0.563	0.570
	CVI	0.248	0.232	0.367	0.446
40%	ARI	0.160	0.316	0.379	0.712
	CVI	0.182	0.187	0.282	0.586
50%	ARI	0.072	0.190	0.398	0.464
	CVI	0.202	0.167	0.390	0.675

TABLE VI
CLUSTER PERFORMANCE ARI AND CVI ON SB.

Missing Rate	Indicator	MS	MF	KNNMI	ANDI
10%	ARI	0.633	0.864	0.717	1
	CVI	0.249	0.396	0.286	0.484
20%	ARI	0.560	0.890	0.620	0.936
	CVI	0.231	0.425	0.296	0.498
30%	ARI	0.317	0.881	0.529	1
	CVI	0.170	0.453	0.271	0.53
40%	ARI	0.229	0.728	0.313	0.936
	CVI	0.136	0.347	0.240	0.537
50%	ARI	0.156	0.591	0.348	0.953
	CVI	0.139	0.284	0.289	0.61

to 30% compared to KNNMI and slightly better than MF by an average of 10%. Datasets have multiple possible values for each attribute, which contain abundant relationships between attribute values, and these relationships can provide better support for ANDI, resulting in strong performance on these datasets.

In summary, the performance of different methods on various types of datasets suggests that effective imputation can enhance clustering accuracy. Therefore, it is crucial to handle missing values appropriately before clustering tasks. In the ANDI mechanism, clustering and missing value imputation form a mutually beneficial relationship. Specifically, clustering provides references for missing values, while missing value completion supplies additional sample information for clustering. This synergy allows ANDI to achieve effective imputation and clustering performance in a single task, potentially eliminating the need for separate missing value handling as a preprocessing stage. Furthermore, ANDI consistently demonstrates stable performance across different types of datasets, underscoring its robustness.

IV. CONCLUDING REMARKS

The proposed ANDI method is designed to handle incomplete heterogeneous data, managing datasets containing arbitrary combinations of numerical, nominal, and ordinal attributes. The advantage of ANDI lies in its ability to fully

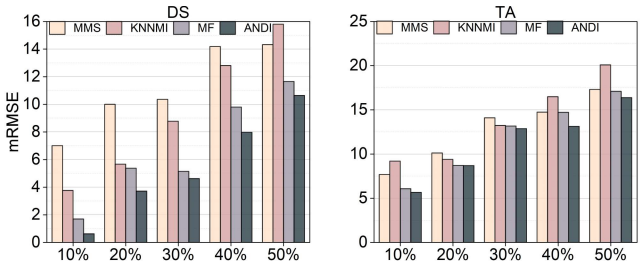


Fig. 4. mRMSE of the imputation method on heterogeneous datasets and five different rates of missing values.

TABLE VII
CLUSTER PERFORMANCE ARI AND CVI ON DS.

Missing Rate	Indicator	MMS	MF	KNNMI	ANDI
10%	ARI	0.196	0.189	0.139	0.226
	CVI	0.343	0.388	0.391	0.487
20%	ARI	0.181	0.289	0.208	0.253
	CVI	0.361	0.372	0.403	0.378
30%	ARI	0.108	0.171	0.244	0.314
	CVI	0.311	0.401	0.348	0.396
40%	ARI	0.069	0.095	0.285	0.249
	CVI	0.351	0.404	0.419	0.451
50%	ARI	0.036	0.034	0.034	0.174
	CVI	0.325	0.285	0.589	0.442

exploiting remaining available information, including inter-object and inter-attribute relationships, to infer missing values. However, further improvements to ANDI are essential, as it shares a common drawback with general clustering methods: the random initialization of center points can significantly impact clustering results, thereby affecting subsequent imputation results. This impact is even more profound in cases of data imbalance.

TABLE VIII
CLUSTER PERFORMANCE ARI AND CVI ON TA.

Missing Rate	Indicator	MMS	MF	KNNMI	ANDI
10%	ARI	0.011	0.018	0.019	0.026
	CVI	0.142	0.120	0.166	0.164
20%	ARI	0.009	0.024	0.040	0.024
	CVI	0.135	0.167	0.111	0.163
30%	ARI	0.004	0.002	-0.001	0.053
	CVI	0.244	0.126	0.129	0.319
40%	ARI	-0.001	-0.005	-0.006	0.002
	CVI	0.126	0.154	0.182	0.36
50%	ARI	0.002	0.004	0.013	0.003
	CVI	0.125	0.188	0.154	0.318

REFERENCES

[1] M. Pattanodom, N. Iam-On, and T. Boongoen, "Clustering data with the presence of missing values by ensemble approach," in *2016 second asian conference on defence technology (acdt)*, pp. 151–156, IEEE, 2016.

[2] R. J. Little and D. B. Rubin, *Statistical analysis with missing data*, vol. 793. John Wiley & Sons, 2019.

[3] O. Troyanskaya, M. Cantor, G. Sherlock, P. Brown, T. Hastie, R. Tibshirani, D. Botstein, and R. B. Altman, "Missing value estimation methods for dna microarrays," *Bioinformatics*, vol. 17, no. 6, pp. 520–525, 2001.

[4] S. Yenduri and S. S. Iyengar, "Performance evaluation of imputation methods for incomplete datasets," *International Journal of Software Engineering and Knowledge Engineering*, vol. 17, no. 01, pp. 127–152, 2007.

[5] E. R. Hruschka, E. R. Hruschka, and N. F. Ebecken, "Towards efficient imputation by nearest-neighbors: A clustering-based approach," in *AI 2004: Advances in Artificial Intelligence: 17th Australian Joint Conference on Artificial Intelligence, Cairns, Australia, December 4-6, 2004. Proceedings 17*, pp. 513–525, Springer, 2005.

[6] S. Gajawada and D. Toshniwal, "Missing value imputation method based on clustering and nearest neighbours," *International Journal of Future Computer and Communication*, vol. 1, no. 2, pp. 206–208, 2012.

[7] J. R. Quinlan, *C4. 5: programs for machine learning*. Elsevier, 2014.

[8] W. Zhang, "Association-based multiple imputation in multivariate datasets: A summary," in *2000 IEEE 16th International Conference on Data Engineering (ICDE'00)*, 2000.

[9] D. B. Rubin, "Multiple imputations in sample surveys-a phenomenological bayesian approach to nonresponse," in *Proceedings of the survey research methods section of the American Statistical Association*, vol. 1, pp. 20–34, American Statistical Association Alexandria, VA, 1978.

[10] R. J. Little and M. D. Schluchter, "Maximum likelihood estimation for mixed continuous and categorical data with missing values," *Biometrika*, vol. 72, no. 3, pp. 497–512, 1985.

[11] M. J. Azur, E. A. Stuart, C. Frangakis, and P. J. Leaf, "Multiple imputation by chained equations: what is it and how does it work?," *International journal of methods in psychiatric research*, vol. 20, no. 1, pp. 40–49, 2011.

[12] D. J. Stekhoven and P. Bühlmann, "Missforest—non-parametric missing value imputation for mixed-type data," *Bioinformatics*, vol. 28, no. 1, pp. 112–118, 2012.

[13] S. J. Rigatti, "Random forest," *Journal of Insurance Medicine*, vol. 47, no. 1, pp. 31–39, 2017.

[14] Q. Zhu, J. Feng, and J. Huang, "Natural neighbor: A self-adaptive neighborhood method without parameter k," *Pattern Recognition Letters*, vol. 80, pp. 30–36, 2016.

[15] J. K. Dixon, "Pattern recognition with partly missing data," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 10, pp. 617–621, 1979.

[16] Y. Zhang and Y.-M. Cheung, "Graph-based dissimilarity measurement for cluster analysis of any-type-attributed data," *IEEE transactions on neural networks and learning systems*, vol. 34, no. 9, pp. 6530–6544, 2022.

[17] F. Tang and H. Ishwaran, "Random forest missing data algorithms," *Statistical Analysis and Data Mining: The ASA Data Science Journal*, vol. 10, no. 6, pp. 363–377, 2017.

[18] U. Lall and A. Sharma, "A nearest neighbor bootstrap for resampling hydrologic time series," *Water resources research*, vol. 32, no. 3, pp. 679–693, 1996.

[19] J. A. Hartigan and M. A. Wong, "Algorithm as 136: A k-means clustering algorithm," *Journal of the royal statistical society. series c (applied statistics)*, vol. 28, no. 1, pp. 100–108, 1979.

[20] Z. Huang, "Extensions to the k-means algorithm for clustering large data sets with categorical values," *Data mining and knowledge discovery*, vol. 2, no. 3, pp. 283–304, 1998.

[21] Z. Huang, "Clustering large data sets with mixed numeric and categorical values," proceedings of 1st pacific-asia conference on knowledge discouvery and data mining," 1997.

[22] Y. Gu, S. Zhang, L. Qiu, Z. Wang, and L. Zhang, "A layered knn-svm approach to predict missing values of functional requirements in product customization," *Applied Sciences*, vol. 11, no. 5, p. 2420, 2021.

[23] L. Hubert and P. Arabie, "Comparing partitions," *Journal of classification*, vol. 2, pp. 193–218, 1985.

[24] A. Starczewski and A. Krzyżak, "Performance evaluation of the silhouette index," in *Artificial Intelligence and Soft Computing: 14th International Conference, ICAISC 2015, Zakopane, Poland, June 14-18, 2015, Proceedings, Part II 14*, pp. 49–58, Springer, 2015.

Best Paper Award

This certificate is proudly presented to

[redacted]

For the paper titled

"Federated Clustering with Unknown Number of Clusters"

co-authored by [redacted], *Yunfan Zhang,* [redacted]
The 6th International Conference on Data-driven Optimization of Complex Systems (DOCS 2024).

General Chair: Yaochu Jin

General Chair: Fei Wu

IEEE Computational Intelligence Society

August 16-18, 2024
Hangzhou, China

国 家 知 识 产 权 局

510620

广东省广州市天河区体育西路 191 号 B 塔 4416 广州粤高专利商标代理有限公司
林丽明(020-32502947)

发文日：

2025 年 03 月 11 日

申请号或专利号：202411660127.6 **发文序号：**2025031100714970

申请人或专利权人：学校

发明创造名称：一种疾控大数据分析方法及系统

发 明 专 利 申 请 进 入 实 质 审 查 阶 段 通 知 书

上述专利申请，根据申请人提出的实质审查请求，经审查，符合专利法第 35 条及实施细则第 113 条的规定，该专利申请进入实质审查阶段。

提示：

1.根据专利法实施细则第 57 条第 1 款的规定，发明专利申请人自收到本通知书之日起 3 个月内，可以对发明专利申请主动提出修改。

2.申请文件修改格式要求：

对权利要求修改的应当提交相应的权利要求替换项，涉及权利要求引用关系时，则需要将相应权项一起替换补正。如果申请人需要删除部分权项，申请人应该提交整理后连续编号的部分权利要求书。

对说明书修改的应当提交相应的说明书替换段，不得增加和删除段号，仅只能对有修改部分段进行整段替换。如果要增加内容，则只能增加在某一段中；如果需要删除一个整段内容，应该保留该段号，并在此段号后注明：“此段删除”字样。段号以国家知识产权局回传的或公布/授权公告的说明书段号为准。

对说明书附图修改的应当以图为单位提交相应的替换附图。

对说明书摘要文字部分修改的应当提交相应的替换页。对摘要附图修改的应当重新指定。

同时，申请人应当在补正书或意见陈述书中标明修改涉及的权项、段号、图、页。

审 查 员：自动审查　　审查部门：初审及流程管理部

联系电话：010-62356655

国家知识产权局 专利审查业务章 11010813597 34

国 家 知 识 产 权 局

510620

广东省广州市天河区体育西路 191 号 B 塔 4416 广州粤高专利商标代理有限公司

冯振宁(0510-83786820)

发文日：

2024 年 12 月 26 日

申请号：202411932856.2　　**发文序号：**2024122600822270

专 利 申 请 受 理 通 知 书

根据专利法第 28 条及其实施细则第 43 条、第 44 条的规定，申请人提出的专利申请已由国家知识产权局受理。现将确定的申请号、申请日等信息通知如下：

申请号：2024119328562

申请日：2024 年 12 月 26 日

申请人：[redacted]

发明人：[redacted]甘国基,龙志全[redacted]

发明创造名称：一种基于细微运动与表观信息补偿的说话人合成方法及系统

经核实，国家知识产权局确认收到文件如下：

权利要求书 1 份 5 页,权利要求项数 ： 10 项

说明书 1 份 13 页

说明书附图 1 份 4 页

说明书摘要 1 份 1 页

发明专利请求书 1 份 5 页

实质审查请求书 文件份数：1 份

申请方案卷号：YGZS2413147UX414

提示：

1.申请人收到专利申请受理通知书之后，认为其记载的内容与申请人所提交的相应内容不一致时，可以向国家知识产权局请求更正。

2.申请人收到专利申请受理通知书之后，再向国家知识产权局办理各种手续时，均应当准确、清晰地写明申请号。

审 查 员：自动受理

联系电话：010-62356655

审查部门：初审及流程管理部

国家知识产权局 专利申查业务章 1101081359734

国 家 知 识 产 权 局

510620

广东省广州市天河区体育西路 191 号 B 塔 4416 广州粤高专利商标代理有限公司

冯振宁(0510-83786820)

发文日：

2025 年 01 月 10 日

申请号：202510043748.8　　**发文序号：**2025011001726310

专 利 申 请 受 理 通 知 书

根据专利法第 28 条及其实施细则第 43 条、第 44 条的规定，申请人提出的专利申请已由国家知识产权局受理。现将确定的申请号、申请日等信息通知如下：

申请号：2025100437488

申请日：2025 年 01 月 10 日

申请人：[redacted]

发明人：[redacted]甘国基,龙志全[redacted]

发明创造名称：一种基于时空上下文场景关系传播的视频显著性物体检测方法及系统

经核实，国家知识产权局确认收到文件如下：

权利要求书 1 份 4 页,权利要求项数 ： 10 项

说明书 1 份 11 页

说明书附图 1 份 2 页

说明书摘要 1 份 1 页

发明专利请求书 1 份 5 页

实质审查请求书 文件份数：1 份

申请方案卷号：YGZS2414020UX414

提示：

1.申请人收到专利申请受理通知书之后，认为其记载的内容与申请人所提交的相应内容不一致时，可以向国家知识产权局请求更正。

2.申请人收到专利申请受理通知书之后，再向国家知识产权局办理各种手续时，均应当准确、清晰地写明申请号。

审 查 员：自动受理

联系电话：010-62356655

审查部门：初审及流程管理部

国家知识产权局 专利申查业务章 1101081359734