Information Sciences 740 (2026) 123224

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Structure-aware granular ball clustering

Qijia Wang, Mingjing Du*

Jiangsu Key Laboratory of Educational Intelligent Technology, School of Artificial Intelligence and Computer Science, Jiangsu Normal University, Xuzhou, 221116, China

HIGHLIGHTS

- Structure-aware granular ball generation enhances scalability.
- Soft affiliation graph via Gaussian kernels models global structure.
- Dual-constraint connectivity criterion identifies underlying cluster structures.
- SAGBC outperforms 10 baselines in clustering accuracy and efficiency.

ARTICLE INFO

Keywords:
Granular ball
Clustering
Granular computing
Structure awareness
Large-scale data
Shared nearest neighbors

ABSTRACT

In the context of the growing prevalence of large-scale data, traditional clustering algorithms face significant bottlenecks in terms of computational efficiency and structural representation capability. This paper proposes a Structure-Aware Granular Ball Clustering method (SAGBC). The SAGBC framework is the first to be developed based on the soft affiliation graph and the dual-constraint connectivity criterion for clustering. Specifically, SAGBC adopts granular balls as the fundamental modeling units and constructs a soft affiliation graph between data points and granular balls to achieve structure-aware representations of complex cluster formations. A dual-constraint connectivity criterion, which integrates spatial proximity with structural similarity, is developed to form connected components of granular balls and effectively capture underlying cluster structures. Extensive experiments on 17 synthetic and real-world datasets demonstrate that SAGBC achieves higher clustering accuracy and computational efficiency than 10 baseline methods, validating its effectiveness for large-scale data analysis. The source code is publicly available at https://github.com/Du-Team/SAGBC.

1. Introduction

Clustering, as a fundamental unsupervised learning method, plays a vital role in structure discovery, latent pattern identification [1], and feature-level data preprocessing [2]. However, with the rapid expansion of data volume in today's information-intensive society, traditional clustering algorithms face growing challenges in scalability and efficiency [3]. To meet the demands of large-scale data analysis, various approaches have been explored, including anchor-based strategies [4], parallel computing frameworks [5], and structure-preserving approximation techniques [6]. Among these, granular ball computing has emerged as an increasingly prominent solution, leveraging information granules rather than individual data points to achieve computational efficiency [7]. By abstracting data into spherical information granules, granular ball clustering not only reduces computational costs but also enhances robustness,

* Corresponding author.
Email addresses: wwqqjj98@163.com (Q. Wang), dumj@jsnu.edu.cn (M. Du).

https://doi.org/10.1016/j.ins.2026.123224
Received 4 November 2025; Received in revised form 7 February 2026; Accepted 7 February 2026
Available online 11 February 2026

making it a promising paradigm for handling large-scale and complex datasets [8]. However, existing granular ball clustering methods still face several challenges that limit their effectiveness in large-scale and complex data scenarios [9].

A primary challenge lies in the computational inefficiency of existing granular ball generation approaches, which always rely on recursive splitting operations across the entire dataset, making them increasingly impractical as data volume grows [10]. Furthermore, the use of a hard partitioning scheme in existing granular ball generation, where each data point is rigidly allocated to a single granular ball, may lead to artificial boundaries during the clustering process, which can distort natural cluster structures, particularly in regions with density transitions [11]. Additionally, in the clustering stage, most existing granular ball merging mechanisms rely solely on a single distance-based criterion, neglecting other connectivity information that could better reflect the underlying data topology, resulting in unsatisfactory clustering outcomes [12]. In summary, these limitations reveal the fundamental issues with existing granular ball clustering methods: high computational costs in large-scale data; rigid data partitioning schemes that are prone to distorting natural cluster structures; and an over-reliance on a single distance-based criterion during the clustering phase, which overlooks other crucial connectivity information, thereby affecting the clustering results.

To address these challenges, this paper proposes a structure-aware granular ball clustering framework tailored for large-scale data. In the granular ball generation stage, a representative subset is sampled to generate granular balls that capture global structural patterns, and a soft affiliation graph is constructed via Gaussian kernels to preserve the essential topological relationships for subsequent similarity computation. In the granular ball connection stage, connectivity among granular balls is established based on dual constraints of spatial proximity and structural similarity, resulting in their connected components. Finally, cluster assignments are derived by propagating labels from the most strongly affiliated granular balls to each data point through the soft affiliation graph.

The key innovations of this work are summarized as follows:

- A structure-aware granular ball generation mechanism is proposed, which constructs granular balls on randomly sampled data, thereby reducing full-data dependency and enhancing scalability to large-scale datasets.
- A soft affiliation graph based on Gaussian kernels is developed, which enables each data point to establish associations with multiple granular balls, thereby capturing a more comprehensive topological structure.
- A dual-constraint connectivity criterion is introduced, which integrates spatial proximity with structural similarity to form connected components of granular balls, accurately identifying underlying cluster structures.

The remainder of this paper is organized as follows: Section 2 reviews existing work in the relevant field; Section 3 introduces the proposed structure-aware granular ball clustering framework; Section 4 evaluates the method through benchmark dataset experiments compared to representative baselines; Section 5 concludes the study and discusses future research directions.

2. Related work

This section reviews two research directions closely related to the present study: scalable clustering for large-scale data and granular ball clustering.

2.1. Scalable clustering for large-scale data

To enhance scalability [13], recent research on large-scale clustering has focused on efficient partitioning and approximation strategies [14]. Several methods adopt spatial or randomized partitioning schemes to reduce computational complexity while preserving clustering quality [15]. For example, RP-DBSCAN [16] adopts a pseudo-random partitioning scheme combined with grid-based indexing, effectively leveraging spatial locality and random balancing to support efficient clustering on large-scale datasets. Similarly, P-Kmeans [17] introduces a lightweight partitioning mechanism designed for resource-constrained environments. By incorporating a constant-factor approximation control, it further enhances the efficiency of clustering under limited computational budgets.

Approximation-based methods, on the other hand, seek to reduce the complexity of clustering through representative sampling and graph simplification. U-SPEC [18] constructs an approximate similarity graph using a hybrid selection strategy that combines random sampling with k-means refinement, effectively balancing cost and accuracy. Based on this, U-SENC [18] extends U-SPEC into an ensemble framework that aggregates multiple approximated graphs, thereby improving the robustness and stability in large-scale clustering scenarios. Building upon such approximation strategies, another widely adopted direction is anchor-based modeling, where a compact set of representative anchors is used to summarize the data and construct a reduced graph for clustering. Within this paradigm, CDC [19] presents a unified framework that adaptively learns high-quality anchors via a similarity-preserving regularizer, reducing computational complexity.

Beyond basic scalability, recent studies exploit local density information and prior knowledge to guide clustering and reduce unnecessary computation. For instance, SR-PCM-HDP [20] selects high-density points to guide initialization and employs a self-regulating process to prune redundant clusters during optimization. Meanwhile, CWLP [21] incorporates both pairwise constraints and cluster-ratio priors into a unified spectral clustering framework, demonstrating how dual weak supervision can improve structure discovery.

Unlike existing methods, the proposed method introduces a higher-level data abstraction through structure-aware granular balls and their soft affiliation graph. Both computation and clustering are conducted at the granular ball level, rather than directly on raw data points, which significantly reduces computational overhead while preserving clustering quality.

2.2. Granular ball clustering

To achieve efficient and robust representations of data, the GBkNN algorithm [7] introduces a model based on granular ball coverage within the sample space. This approach notably improves processing efficiency, robustness, and the interpretability of data. Building upon this foundational representation, numerous granular ball clustering techniques have since emerged. For example, Ball k-means [8] speeds up clustering by utilizing the distances between individual data points and the centers of granular balls. Clusters are represented through these granular balls, where their well-defined geometric forms and coarse-grained summaries facilitate the detection of neighboring clusters at a coarse level. This reduces redundant distance computations and boosts the performance of standard k-means algorithms.

To better manage complex and non-spherical data distributions, GBC [10] employs an adaptive mechanism for generating granular balls that adjusts each ball's shape and size dynamically, guided by heuristic quality metrics and splitting strategies. This granular ball structure serves as the basis for a hierarchical clustering framework that effectively organizes data with intricate distributions. GBG++ [22] utilizes an attention mechanism and distribution-driven center initialization to reduce computational overhead and eliminate instability. Subsequent research, such as GB-POJG [23], incorporates the principle of justifiable granularity to optimize ball quality through integrated coverage and specificity metrics. Furthermore, the LGBQPC method [24] leverages these quality indices within a local structural analysis framework to accurately identify clusters with arbitrary shapes and non-uniform densities.

Similarly, Cheng et al. [25] propose a density peak clustering method founded on granular balls, where density calculations are performed at the ball level instead of individual sample points, with clustering driven by distances between ball centers. The GBDBSCAN algorithm [26] applies an unsupervised KNN-based partitioning strategy to form granular balls, quantifying ball density via radius and categorizing balls into core and non-core types accordingly. This innovation significantly improves DBSCAN's efficiency. To address the high computational demands commonly encountered in spectral clustering, Xie et al. [27] develop a granular ball-based method that models data as granular balls and constructs an adjacency matrix from their relationships. This approach facilitates efficient spectral clustering without costly pairwise distance computations.

Addressing robustness against noise in granular ball clustering, particularly for irregular cluster shapes, the GBMST method [12] integrates granular ball computations with graph theory by building a minimum spanning tree (MST) from the centers of granular balls. By pruning edges linked to low-density balls, this method naturally isolates noise points and accommodates clusters of complex geometries. To mitigate parameter sensitivity in high-dimensional clustering, the W-GBC algorithm [11] introduces an iterative global weighting approach applied locally within granular balls. This reduces dependence on extensive parameter tuning and enhances both adaptability and scalability. In scenarios involving data streams, the GBFuzzyStream algorithm [28] incorporates granular ball structures within a streaming clustering framework.

For large-scale datasets, the GB-USC method [29] combines random sampling and granular ball computing to select representative anchors. It then constructs a bipartite graph between data points and anchors, applies a graph-based dimensionality reduction approach to derive low-dimensional manifold embeddings, and finally clusters the data using k-means.

Beyond clustering, granular computing has demonstrated significant efficiency and robustness in classification tasks. To accelerate twin support vector machines, GBTSVM [30] utilizes granular balls as fundamental inputs to construct decision planes, effectively eliminating costly matrix operations in large-scale scenarios. This framework is further enhanced by Pin-GBTSVM [31], which integrates a pinball loss function to improve stability against noise and outliers while maintaining high computational efficiency.

Unlike conventional granular ball methods that require all data points to be strictly assigned to individual balls, our method allows data points to establish soft, non-exclusive affiliations with multiple granular balls. Specifically, a Gaussian kernel is employed to quantify the strength of these affiliations, resulting in a sparse and structure-aware membership graph, which facilitates granular ball-level merging and enables a top-down structural propagation from granular balls to the entire dataset.

3. Method

This section first presents the overall framework of the proposed algorithm. Then, the key components and their corresponding techniques are discussed in detail. The framework is illustrated in Fig. 1, which outlines the algorithm's three main stages: (1) structure-aware granular ball generation, (2) granular ball connected component construction, and (3) data point cluster assignment.

Specifically, structure-aware granular ball generation integrates representative sampling and recursive splitting to obtain structure-aware granular balls, and further constructs a Gaussian-kernel-based soft affiliation graph between data points and granular balls to quantify point-to-ball associations. Granular ball connected component construction organizes the granular balls into a connectivity graph by considering spatial proximity and structural similarity, and then extracts connected components to capture the global cluster structure. Data point cluster assignment finalizes the clustering by assigning each data point according to the connected component membership of its affiliated granular balls.

The notations used throughout this paper are outlined in Table 1 for easy reference.

3.1. Structure-aware granular ball generation

3.1.1. Representative sampling

Existing granular ball generation approaches often rely on recursive splitting operations over the entire dataset, which can become computationally prohibitive for large-scale data.

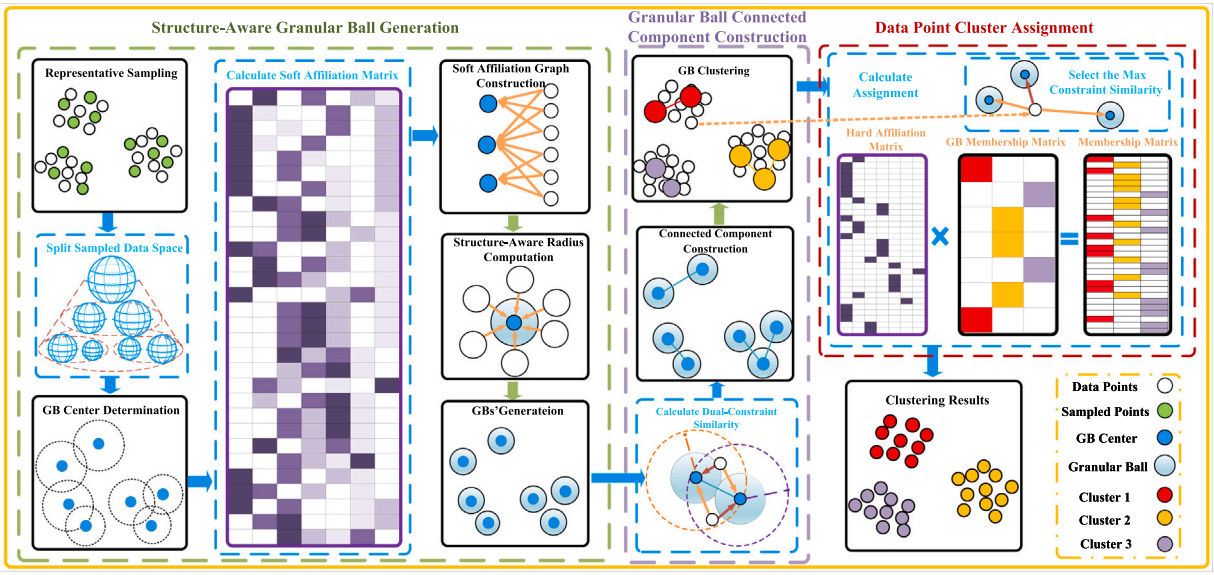


Fig. 1. Overall structure of SAGBC framework.

Table 1
Notations.

Notations	Descriptions
Dataset and Sampling	
$\boldsymbol{X}$	Original dataset, $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n\}$
n	Total number of data points
d	Feature dimensionality
$\boldsymbol{Z}$	Sampled subset of data points, $\boldsymbol{Z} \subset \boldsymbol{X}$
t	Number of sampled data points, $t \ll n$
τ	Sampling ratio, $t = \lfloor \tau \cdot n \rfloor$
Structure-Aware Granular Ball Generation	
$\boldsymbol{G}$	Granular ball set, $\boldsymbol{G} = \{\boldsymbol{g}_1, \ldots, \boldsymbol{g}_m\}$
m	Total number of generated granular balls
$\boldsymbol{\Theta}$	Set of granular ball centers, $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_m\}$
$\boldsymbol{\theta}_k$	Center of the k-th granular ball $\boldsymbol{g}_k$
$\boldsymbol{R}$	Set of structure-aware radii
r_k	Structure-aware radius of the k-th granular ball
$\bar{r}_k$	Search radius of $\boldsymbol{g}_k$, $\bar{r}_k = \gamma \cdot r_k$
γ	Scaling factor for the search radius
$\boldsymbol{b}$	Intermediate subclusters/clusters
$\boldsymbol{W}$	Soft affiliation matrix, $\boldsymbol{W} \in \mathbb{R}^{n \times m}$
w_{ik}	Gaussian kernel similarity between point $\boldsymbol{x}_i$ and center $\boldsymbol{\theta}_k$
σ	Gaussian kernel bandwidth
λ	Number of nearest granular balls per point
$\hat{\boldsymbol{X}}_k$	Set of data points affiliated with granular ball $\boldsymbol{g}_k$
Granular Ball Connected Component Construction	
s_{kp}^{spa}	Spatial similarity between granular balls $\boldsymbol{g}_k$ and $\boldsymbol{g}_p$
s_{kp}^{str}	Structural similarity between granular balls $\boldsymbol{g}_k$ and $\boldsymbol{g}_p$
s_{kp}	Dual-constraint similarity, $s_{kp} = s_{kp}^{\text{spa}} \times s_{kp}^{\text{str}}$
$\mathbb{I}(\cdot)$	Indicator function
$\boldsymbol{M}$	Granular ball graph, $\boldsymbol{M} = (\boldsymbol{V}, \boldsymbol{E})$
$\boldsymbol{V}$	Set of vertices (granular balls)
$\boldsymbol{E}$	Set of edges between granular balls
$\boldsymbol{S}^{\text{spa}}$	Spatial similarity matrix
$\boldsymbol{S}^{\text{str}}$	Structural similarity matrix
$\boldsymbol{S}$	Dual-constraint similarity matrix
$\hat{C}$	Set of connected components, $\hat{C} = \{\hat{\boldsymbol{C}}_1, \ldots, \hat{\boldsymbol{C}}_c\}$
$\hat{\boldsymbol{C}}_j$	j-th connected component (cluster of granular balls)
Data Point Cluster Assignment	
$\hat{y}_k$	Cluster label of granular ball $\boldsymbol{g}_k$
y_i	Final cluster label of data point $\boldsymbol{x}_i$
$\hat{\boldsymbol{G}}_i$	Set of granular balls affiliated with data point $\boldsymbol{x}_i$

To address this challenge, we adopt a structure-aware granular ball modeling strategy by introducing a representative sampling mechanism. This mechanism extracts a subset of data points to serve as the basis for granular ball construction, allowing the approach to retain critical topological features while significantly reducing computational overhead.

To facilitate the construction of granular balls, a representative subset is first extracted from the original dataset $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n\}$ through a sampling process. Specifically, a subset $\boldsymbol{Z} = \{\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_t\} \subset \boldsymbol{X}$ is selected, where $t \ll n$. Sampling probabilities can be designed to capture underlying data characteristics such as local density or centrality. In its most straightforward form, uniform random sampling is employed, where each data point is independently selected with equal probability. This process involves generating a set of random indices, which are integer values uniformly sampled from the range of available data points. Each index corresponds to a specific data point in the dataset, ensuring that each point has an equal chance of being selected. By drawing these indices randomly, the approach guarantees unbiased selection, allowing for a representative sample that reflects the entire dataset without any uniform random sampling.

3.1.2. Granular ball center determination

To determine the centers of granular balls, a farthest-point-based binary splitting process is performed on the sampled subset. Specifically, the two points with the greatest pairwise distance within the sample set are identified as initial seeds, and the remaining points are assigned to the nearer seed to form two subclusters. The geometric centroid of each subcluster is then computed and designated as its provisional center. This procedure is recursively applied within each subcluster until its size falls below a predefined threshold, at which point the geometric center of the final subcluster is taken as the granular ball center. Formally, we denote the center of the k-th granular ball $\boldsymbol{g}_k$ as $\boldsymbol{\theta}_k$.

3.1.3. Soft affiliation graph construction

Unlike traditional granular ball construction approaches that traverse all data points, we determine the granular ball centers using a sampled subset of the dataset. Subsequently, we establish connections between these centers and all data points in the dataset (not limited to the sampled subset) to capture the global structural relationships. To further strengthen the representation ability of granular balls, we introduce a soft assignment mechanism based on a Gaussian kernel similarity. Specifically, the affiliation between a granular ball center $\boldsymbol{\theta}_k$ and data points $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n\}$ is measured as

$$w_{ik} = \exp\left(-\frac{\|\boldsymbol{x}_i - \boldsymbol{\theta}_k\|_2^2}{2\sigma^2}\right), \tag{1}$$

where $\boldsymbol{\theta}_k$ denotes the center of the k-th granular ball and σ is the kernel bandwidth controlling the decay rate of similarity. To ensure the framework remains robust across diverse datasets without manual tuning, σ is adaptively determined as the average Euclidean distance between data points and their λ nearest granular ball centers. Notably, these weights collectively form the affiliation matrix $\boldsymbol{W} \in \mathbb{R}^{n \times m}$.

Each data point is linked to its λ nearest granular balls according to w_{ik}, yielding a sparse affiliation graph that encodes the soft membership relationships between data points and granular balls.

3.1.4. Structure-aware radius computation

The radius of each granular ball is estimated based on the resulting soft affiliation graph. Unlike conventional approaches that compute the radius directly from all internal points, the proposed approach adaptively determines it using a weighted averaging scheme that accommodates local data distribution. The proposed structure-aware radius is formally defined as:

Definition 1. For a granular ball $\boldsymbol{g}_k$, the structure-aware radius is defined as the weighted average distance between its center and all data points connected to it in the affiliation graph:

$$r_k = \frac{\sum_{\boldsymbol{x}_i \in \hat{\boldsymbol{X}}_k} w_{ik} \cdot \|\boldsymbol{x}_i - \boldsymbol{\theta}_k\|_2}{\sum_{\boldsymbol{x}_i \in \hat{\boldsymbol{X}}_k} w_{ik}}, \tag{2}$$

where $\boldsymbol{\theta}_k$ denotes the center of $\boldsymbol{g}_k$, $\hat{\boldsymbol{X}}_k$ represents the set of data points affiliated with $\boldsymbol{\theta}_k$, and w_{ik} is the Gaussian-kernel similarity weight between $\boldsymbol{x}_i$ and $\boldsymbol{\theta}_k$.

The structure-aware granular ball generation strategy is detailed in Algorithm 1. The constructed granular balls capture the local structural characteristics of the data, while the soft affiliation graph models the global relationships between data points and granular balls. Together, they provide complementary structural information that facilitates subsequent granular ball connection and the overall clustering process.

3.2. Granular ball connected component construction

Most existing granular ball connection mechanisms rely solely on distance-based criteria, which often fail to deliver satisfactory results. To address this limitation, we introduce a dual-constraint connectivity criterion that establishes connections between granular balls by incorporating both spatial and structural information. Building upon this criterion, we further enhance the efficiency of the connection process by developing a multi-threaded processing framework. The framework distributes granular balls in a balanced manner across threads, where each thread applies the dual-constraint criterion to determine valid connections between granular

Algorithm 1 Structure-Aware granular ball generation.

Input: Dataset $\boldsymbol{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$, sampling ratio τ, neighbor count λ, bandwidth σ
Output: Granular ball set $\boldsymbol{G} = \{(\boldsymbol{\theta}_k, r_k)\}_{k=1}^{|\boldsymbol{m}|}$, soft affiliation graph $\boldsymbol{W}$

```
// Step 1: Sampling
```
Sample $\boldsymbol{Z} \subset \boldsymbol{X}$ with $t = \lfloor \tau \cdot n \rfloor$ via random sampling
```
// Step 2: Center Determination
```
Initialize $\boldsymbol{G} \leftarrow \emptyset$, $\boldsymbol{\Theta} \leftarrow \emptyset$, $\boldsymbol{R} \leftarrow \emptyset$, $k \leftarrow 1$, $\boldsymbol{W} \leftarrow \boldsymbol{0}^{n \times m}$
Construct root cluster $\boldsymbol{b}_0$ covering $\boldsymbol{Z}$ and enqueue into $\boldsymbol{Q}$
while $\boldsymbol{Q} \neq \emptyset$ **do**
  Dequeue cluster $\boldsymbol{b}$ from $\boldsymbol{Q}$
  if $|\boldsymbol{b}| > \sqrt{t}$ **then**
    Find farthest pair $\boldsymbol{z}_\alpha, \boldsymbol{z}_\beta \in \boldsymbol{b}$
    $\boldsymbol{b}_1 \leftarrow \{\boldsymbol{z}_i \in \boldsymbol{b} | \|\boldsymbol{z}_i - \boldsymbol{z}_\alpha\| \leq \|\boldsymbol{z}_i - \boldsymbol{z}_\beta\|\}$
    $\boldsymbol{b}_2 \leftarrow \boldsymbol{b} \setminus \boldsymbol{b}_1$
    Enqueue $\boldsymbol{b}_1$, $\boldsymbol{b}_2$ into $\boldsymbol{Q}$
  else
    $\boldsymbol{\theta}_k \leftarrow \frac{1}{|\boldsymbol{b}|} \sum_{\boldsymbol{z}_i \in \boldsymbol{b}} \boldsymbol{z}_i$
    $\boldsymbol{\Theta} \leftarrow \boldsymbol{\Theta} \cup \{\boldsymbol{\theta}_k\}$
    $\boldsymbol{G} \leftarrow \boldsymbol{G} \cup \{(\boldsymbol{\theta}_k, \emptyset)\}$
    $k \leftarrow k + 1$
```
// Step 3: Affiliation Graph Construction
```
for $\boldsymbol{x}_i \in \boldsymbol{X}$ **do**
  for $\boldsymbol{\theta}_j \in \boldsymbol{\Theta}$ **do**
    $w_{ij} \leftarrow \exp(-\|\boldsymbol{x}_i - \boldsymbol{\theta}_j\|^2/(2\sigma^2))$
  Set $w_{ij} = 0$ for all but top-λ largest values
```
// Step 4: Radius Computation
```
for $\boldsymbol{\theta}_k \in \boldsymbol{\Theta}$ **do**
  $r_k \leftarrow \frac{\sum_i w_{ik} \|\boldsymbol{x}_i - \boldsymbol{\theta}_k\|}{\sum_i w_{ik}}$
  $\boldsymbol{R} \leftarrow \boldsymbol{R} \cup \{r_k\}$
  Update granular ball as $(\boldsymbol{\theta}_k, r_k)$
return $\boldsymbol{G}, \boldsymbol{W}$

balls, thereby constructing their global connected components. This procedure corresponds to the granular ball connected component construction stage in Fig. 1.

The proposed dual-constraint connectivity criterion integrates both spatial and structural information through complementary similarity measures. In order to define the spatial similarity, we first introduce a search radius for each granular ball that adaptively scales with its spatial characteristics. The search radius is formally defined as follows:

Definition 2. For a granular ball $\boldsymbol{g}_k$, the search radius is defined as a scaled extension of its structure-aware radius:

$$\bar{r}_k = \gamma \cdot r_k, \tag{3}$$

where r_k denotes the structure-aware radius of $\boldsymbol{g}_k$ and γ is a scaling factor that controls the extent of the search range.

In our experiments, γ is set to 2 to ensure sufficient coverage for connection discovery while maintaining local relevance.

Based on this definition, the spatial similarity between two granular balls is determined by whether their centers lie within each other's search range. We formalize this similarity as follows.

Definition 3. Given two granular balls $\boldsymbol{g}_k$ and $\boldsymbol{g}_p$, their spatial similarity is defined as:

$$s_{kp}^{\text{spa}} = \mathbb{I}\left(\left\|\boldsymbol{\theta}_k - \boldsymbol{\theta}_p\right\|_2 \leq \min(\bar{r}_k, \bar{r}_p)\right), \tag{4}$$

where $\boldsymbol{\theta}_k$ and $\boldsymbol{\theta}_p$ denote the centers of $\boldsymbol{g}_k$ and $\boldsymbol{g}_p$, respectively, $\bar{r}_k$ and $\bar{r}_p$ are their search radii, and $\mathbb{I}(\cdot)$ is the indicator function that returns 1 if the condition is satisfied and 0 otherwise.

This definition ensures that only granular ball pairs within the intersection of their respective search ranges are considered for further similarity evaluation.

While spatial similarity effectively captures spatial proximity between granular balls, it may be sensitive to variations in local density. To complement this measure, we incorporate structural information by considering the extent to which two granular balls share nearest neighbors. Intuitively, if their neighbor sets exhibit a high degree of overlap, these granular balls are potentially structurally similar in the underlying data distribution. We formalize this similarity as follows.

Definition 4. Given two granular balls $\boldsymbol{g}_k$ and $\boldsymbol{g}_p$, their structural similarity is defined as:

$$s_{kp}^{\text{str}} = \frac{|\hat{\boldsymbol{X}}_k \cap \hat{\boldsymbol{X}}_p|}{\min(|\hat{\boldsymbol{X}}_k|, |\hat{\boldsymbol{X}}_p|)}, \tag{5}$$

where $\hat{\boldsymbol{X}}_k$ and $\hat{\boldsymbol{X}}_p$ denote the sets of data points affiliated with $\boldsymbol{g}_k$ and $\boldsymbol{g}_p$, respectively.

This normalized measure evaluates the proportion of common data points in the smaller neighborhood, ranging from 0 (no overlap) to 1 (complete inclusion).

Since spatial and structural similarities capture complementary aspects of the relationship between granular balls, we combine them into a unified measure to holistically evaluate pairwise relationships between granular balls.

Definition 5. Given two granular balls $\boldsymbol{g}_k$ and $\boldsymbol{g}_p$, their dual-constraint similarity is defined as:

$$s_{kp} = s_{kp}^{\text{spa}} \times s_{kp}^{\text{str}}, \tag{6}$$

where s_{kp}^{spa} and s_{kp}^{str} are spatial and structural similarities, respectively.

Based on this similarity, the connectivity criterion is straightforward: two granular balls are connected if $s_{kp} > 0$, and are left unconnected otherwise. It is worth noting that since this similarity is computed as the product of the spatial and structural constraints, a non-zero value signifies that both conditions are met, which enhances the efficiency of filtering by simultaneously discarding connections that violate either criterion.

The resulting connected components preserve only meaningful connections where granular balls are both spatially proximate and structurally related, effectively filtering out spurious or weak relationships while maintaining the essential data topology. The detailed strategy for granular ball connected component construction is outlined in Algorithm 2.

Algorithm 2 Granular ball connected component construction.

Input: Granular ball set $\boldsymbol{G} = \{\boldsymbol{g}_k\}_{k=1}^m$ with centers $\{\boldsymbol{\theta}_k\}_{k=1}^m$, radii $\{r_k\}_{k=1}^m$, affiliated sets $\{\hat{\boldsymbol{X}}_k\}_{k=1}^m$, scaling factor γ
Output: Connected component set $\hat{C} = \{\hat{\boldsymbol{C}}_j\}_{j=1}^c$
Initialize graph $\boldsymbol{M} = (\boldsymbol{V} \leftarrow \boldsymbol{G}, \boldsymbol{E} \leftarrow \emptyset)$ Initialize similarity matrices $\boldsymbol{S}^{\text{spa}} \leftarrow \boldsymbol{0}^{n\times n}$, $\boldsymbol{S}^{\text{str}} \leftarrow \boldsymbol{0}^{n\times n}$, $\boldsymbol{S} \leftarrow \boldsymbol{0}^{n\times n}$
for $\boldsymbol{g}_k \in \boldsymbol{G}$ ***in parallel*** **do**
  $\bar{r}_k \leftarrow \gamma \cdot r_k$
  for $\boldsymbol{g}_p \in \boldsymbol{G}$ *where* $p > k$ **do**
    `// Spatial Similarity`
    $s_{kp}^{\text{spa}} \leftarrow \mathbb{I}(\|\boldsymbol{\theta}_k - \boldsymbol{\theta}_p\|_2 \le \min(\bar{r}_k, \bar{r}_p))$
    if $s_{kp}^{spa} = 1$ **then**
      `// Structural Similarity`
      $s_{kp}^{\text{str}} \leftarrow \frac{|\hat{\boldsymbol{X}}_k \cap \hat{\boldsymbol{X}}_p|}{\min(|\hat{\boldsymbol{X}}_k|, |\hat{\boldsymbol{X}}_p|)}$
      `// Dual-constraint Similarity`
      $s_{kp} \leftarrow s_{kp}^{\text{spa}} \times s_{kp}^{\text{str}}$ `// Dual-constraint Connectivity Criterion`
      if $s_{kp} > 0$ **then**
        Acquire lock on $\boldsymbol{E}$ $\boldsymbol{E} \leftarrow \boldsymbol{E} \cup \{(\boldsymbol{g}_k, \boldsymbol{g}_p)\}$ Release lock
Find connected components $\hat{C} = \{\hat{\boldsymbol{C}}_1, \ldots \hat{\boldsymbol{C}}_c\}$ via union-find set;
return $\hat{C}$

3.3. Data point cluster assignment

Following the construction of granular ball connected components, individual data points are assigned to clusters, as illustrated in the data point cluster assignment stage in Fig. 1. We adopt an assignment approach based on the soft affiliation graph between data points and granular balls. Each connected component defines a granular ball cluster, and the cluster membership of a data point is inferred from its affiliation strengths with the granular balls contained in that component.

Specifically, for a data point $\boldsymbol{x}_i$, the granular ball with the highest affiliation strength is first identified, and the data point is then assigned to the cluster corresponding to that granular ball. Formally, this can be written as:

$$y_i = \hat{y}_{\arg\max_{k:\, \boldsymbol{g}_k \in \hat{\boldsymbol{G}}_i} w_{ik}},$$

where w_{ik} represents the Gaussian kernel affiliation strength between $\boldsymbol{x}_i$ and $\boldsymbol{g}_k$, $\hat{\boldsymbol{G}}_i = \{\boldsymbol{g}_k \mid \boldsymbol{x}_i \in \hat{\boldsymbol{X}}_k\}$ denotes the set of granular balls affiliated with $\boldsymbol{x}_i$, $\hat{y}_k$ is the cluster label of granular ball $\boldsymbol{g}_k$, and y_i is the cluster label of $\boldsymbol{x}_i$.

The detailed procedure for assigning cluster labels to data points is outlined in Algorithm 3. This approach ensures that cluster memberships are propagated from granular balls to data points through the connected components, yielding accurate final assignments.

Algorithm 3 Data point cluster assignment.

Input: Data points $\boldsymbol{X} = \{\boldsymbol{x}_i\}_{i=1}^{n}$, granular ball set $\boldsymbol{G} = \{\boldsymbol{g}_k\}_{k=1}^{m}$, connected components $\hat{C} = \{\hat{\boldsymbol{C}}_j\}_{j=1}^{c}$, affiliation sets $\{\hat{\boldsymbol{X}}_k\}_{k=1}^{m}$, soft affiliation graph $\boldsymbol{W}$

Output: Cluster labels $\{y_i\}_{i=1}^{n}$

for $\hat{\boldsymbol{C}}_j \in \hat{C}$ **do**
  Assign unique cluster label $\hat{y}_k \leftarrow j$ to all $\boldsymbol{g}_k \in \hat{\boldsymbol{C}}_j$

for $\boldsymbol{x}_i \in \boldsymbol{X}$ **do**
  $\hat{\boldsymbol{G}}_i \leftarrow \{\boldsymbol{g}_k \mid \boldsymbol{x}_i \in \hat{\boldsymbol{X}}_k\}$
  $k^* \leftarrow \arg\max_{k:\, \boldsymbol{g}_k \in \hat{\boldsymbol{G}}_i} w_{ik}$
  $y_i \leftarrow \hat{y}_{k^*}$

return $\{y_i\}_{i=1}^{n}$

3.4. Complexity analysis

The computational complexity of the proposed clustering method is analyzed by examining its three major procedures involved in structure-aware granular ball generation, connected component construction, and final cluster assignment.

In the granular ball generation stage, the random sampling of t instances from the dataset requires $O(t)$ time. The hierarchical center determination exhibits $O(t \log t)$ complexity through binary space partitioning over the t sampled instances, from which m granular balls are produced (with $m \leq t$). The selection of the top-λ neighboring granular balls for each data point can be performed via partial sorting after brute-force distance evaluation, leading to a time complexity of $O(nm\lambda)$, which simplifies to $O(nm)$ when λ is treated as a constant. In practice, this step can be further accelerated by using KD-tree–based nearest anchor search, which reduces the expected computational cost to approximately $O(n \log m)$. Finally, the computation of granular ball radii requires a weighted distance aggregation over affiliated points, also with cost $O(nm)$. Hence, the overall complexity of granular ball generation is dominated by $O(t \log t + nm)$.

The construction of granular ball connected components requires pairwise similarity checks among the m granular balls, resulting in $O(m^2)$ time complexity. The subsequent union-find operation for component identification requires nearly linear time $O(m)$. Thus, the complexity of this stage is effectively dominated by $O(m^2)$.

In the final cluster assignment stage, each granular ball in a connected component is assigned a unique cluster label in $O(m)$. In the point-to-granule assignment, each data point selects its optimal granular ball by comparing at most λ weights, resulting in an $O(n\lambda)$ total complexity. Given that λ is typically a small constant, this step scales linearly with the dataset size.

By combining the above analysis, the total complexity of the method is $O(t \log t + m^2 + nm)$. Consequently, the method is applicable to large-scale clustering tasks as long as the sampling ratio τ is appropriately chosen.

4. Experiments results

4.1. Setup

To evaluate the performance of the proposed method, we conduct experiments on datasets of various types and compare the results with ten existing clustering methods. All experiments are performed on a Lenovo Y9000P personal computer equipped with a 3rd Gen Intel® Core™ i9-13900HX processor and 16 GB of RAM. The comparison methods GB-USC, U-SPEC, and LGBQPC are executed using MATLAB R2024b, while all other experiments—including the proposed method and remaining baselines—are implemented in Python 3.7. The source code is publicly available.[1]

4.1.1. Datasets

To comprehensively evaluate the clustering performance and stability of the proposed method, we employ a collection of datasets under varying data characteristics. These datasets include 10 synthetic datasets and 7 real-world datasets, covering differences in dimensionality, scale, and number of clusters, as summarized in Table 2.

The synthetic datasets cover a range of scales from small to large. Several two-dimensional datasets with varied cluster structures are included to assess the algorithm's ability to distinguish complex geometric patterns. In addition, million-scale 2D datasets are used to evaluate performance under large-scale scenarios in terms of both clustering quality and computational efficiency.

[1] https://github.com/Du-Team/SAGBC

Table 2
The dataset used in the experiment.

Dataset type	Data name	Instances	Features	Clusters
Synthetic	Wave	12,762	2	4
Synthetic	Chrome	11,093	2	4
Synthetic	T4	7236	2	6
Synthetic	DS9	6847	2	6
Synthetic	DS3	7247	2	7
Synthetic	S1	5000	2	15
Synthetic	TB	1,000,000	2	7
Synthetic	CC	1,000,000	2	3
Synthetic	Z1	1,048,785	2	5
Synthetic	Z2	1,020,000	2	8
Real-world	Htru2	17,989	8	2
Real-world	CS	10,845	28	6
Real-world	Penbased	10,992	16	10
Real-world	Letter	20,000	16	26
Real-world	ORL	100	10,304	10
Real-world	YTF	10,000	10	41
Real-world	CT	581,012	54	7

The real-world datasets include structured data as well as image data from handwritten character and face recognition tasks. These datasets vary in the number of samples and classes, and are primarily used to test the robustness and generalization ability of the proposed method in practical settings.

4.1.2. Comparison methods

To evaluate the proposed method's performance, we compare it with ten baselines, covering diverse technical paradigms such as parallel clustering, spectral clustering, density-based clustering, and granular ball-based clustering.

Among them, SNN-DBSCAN [32] is a clustering method based on shared nearest neighbors, P-Kmeans [17] and RP-DBSCAN [16] are typical parallel clustering methods, representing partition-based and density-based strategies, respectively. GB-USC [29] utilizes granular ball modeling for large-scale clustering, while U-SPEC [18] employs spectral clustering with efficient approximation techniques. Additionally, we select four granular ball variants: GB-DP [25] (a density peak-based method), W-GBC [11] (a weighted clustering method), LGBQPC [24] (a principle of justifiable granularity clustering method), GBDBSCAN [26] (a density-based extension), and GBSC [27] (a spectral clustering variant).

4.1.3. Evaluation metrics

To comprehensively evaluate the performance of each clustering method, we adopt three commonly used clustering evaluation metrics: Clustering Accuracy (ACC) [33], Normalized Mutual Information (NMI) [34], and Purity [35].

For all comparison methods, the results are averaged over multiple runs to mitigate the influence of random factors.

4.2. Comparison experiments

Table 3 compares the performance of different clustering methods across all datasets, where boldface indicates the best result for each metric. Overall, our proposed method achieves superior performance in terms of ACC, NMI, and Purity on the majority of datasets. The visualization of clustering results on synthetic datasets in Fig. 2 further confirms this advantage, particularly showing our method's capability to accurately identify complex cluster structures.

On the synthetic datasets, our method demonstrates top-tier performance. On small to medium-scale datasets, as illustrated in Figs. 2(a)–(f), our method accurately captures challenging patterns, including non-convex-shaped and fuzzy-boundary clusters. For datasets with complex geometric structures (Wave, Chrome, T4, DS9, and DS3), the proposed method achieves near-perfect or perfect scores. While three spectral approaches (GB-USC, U-SPEC, GBSC) and SNN-DBSCAN achieve competitive accuracy in specific scenarios, the proposed SAGBC consistently delivers superior results across most datasets. The S1 dataset presents a different challenge with its partially overlapping clusters. Despite this structural complexity, our method maintains superior performance, delivering the best results among all compared methods.

On the million-scale synthetic datasets, our method maintains superior performance, achieving the best results in terms of ACC, NMI, and Purity. For the TB and CC datasets with highly compact cluster boundaries, our method maintains clear separation between clusters and significantly outperforms all comparison methods. On Z1 and Z2, our method achieves perfect scores (all metrics equal to 1). It shares the best results with RP-DBSCAN, GB-USC, U-SPEC, and LGBQPC on Z1, and with GB-USC, U-SPEC and LGBQPC on Z2. These results demonstrate the robustness of the proposed method under both scale and structural complexity.

Notably, GB-DBSCAN, GBSC, and SNN-DBSCAN fail to process the large-scale datasets due to excessive computational and memory demands, as evidenced by their missing values on all four million-scale datasets in Table 3.

Across all real-world datasets, our method consistently achieves the highest performance in terms of ACC, NMI, and Purity. On Htru2, CS and CT (three structured datasets), our method significantly outperforms all competing methods. For the Penbased and

Table 3
Performance comparison of clustering algorithms on various datasets.

Algorithm	Metric	Wave	Chrome	T4	DS9	DS3	S1	TB	CC	Z1	Z2	Htru2	CS	Penbased	Letter	ORL	YTF	CT
Ours	ACC	**1.000**	**1.000**	**1.000**	**1.000**	**0.999**	**0.993**	**0.997**	**1.000**	**1.000**	**1.000**	**0.964**	**0.434**	**0.827**	**0.311**	**0.968**	**0.620**	**0.346**
	NMI	**1.000**	**0.999**	**1.000**	**1.000**	**0.999**	**0.985**	**0.973**	**0.998**	**1.000**	**1.000**	**0.594**	**0.384**	**0.876**	**0.474**	**0.953**	**0.848**	**0.174**
	Purity	**1.000**	**1.000**	**1.000**	**1.000**	**0.999**	**0.993**	**0.997**	**1.000**	**1.000**	**1.000**	**0.975**	**0.850**	**0.976**	**0.386**	**0.968**	**0.858**	**0.589**
P-Kmeans	ACC	0.666	0.945	0.694	**1.000**	0.951	0.932	0.791	0.335	0.954	0.708	0.814	0.330	0.770	0.177	0.860	0.611	0.301
	NMI	0.460	0.843	0.695	**1.000**	0.927	0.970	0.260	0.000	0.956	0.885	0.230	0.351	0.827	0.281	0.922	0.786	0.153
	Purity	0.666	0.945	0.774	**1.000**	0.951	0.932	0.791	0.529	0.954	0.891	0.908	0.835	0.770	0.177	0.860	0.689	0.499
GB-USC	ACC	**1.000**	**1.000**	**1.000**	**1.000**	0.945	0.931	0.995	**1.000**	**1.000**	**1.000**	0.895	0.401	0.645	0.248	0.910	0.510	0.240
	NMI	**1.000**	**0.999**	**1.000**	**1.000**	0.967	0.974	0.953	**0.998**	**1.000**	**1.000**	0.336	0.347	0.760	0.405	0.921	0.687	0.095
	Purity	**1.000**	**1.000**	**1.000**	**1.000**	0.945	0.931	0.995	**1.000**	**1.000**	**1.000**	0.908	0.773	0.668	0.281	0.910	0.582	0.556
U-SPEC	ACC	**1.000**	**1.000**	**1.000**	**1.000**	0.970	0.927	0.995	**1.000**	**1.000**	**1.000**	0.890	0.401	0.822	0.198	0.850	0.575	0.329
	NMI	**1.000**	**0.999**	**1.000**	**1.000**	0.967	0.972	0.956	**0.998**	**1.000**	**1.000**	0.326	0.347	0.825	0.332	0.928	0.759	0.082
	Purity	**1.000**	**1.000**	**1.000**	**1.000**	0.970	0.927	0.995	**1.000**	**1.000**	**1.000**	0.908	0.773	0.832	0.236	0.900	0.659	0.518
RP-DBSCAN	ACC	0.467	0.647	0.230	0.036	0.040	0.000	0.582	0.322	**1.000**	0.977	0.058	0.110	0.137	0.095	0.231	0.124	0.103
	NMI	0.285	0.386	0.022	0.398	0.413	0.000	0.094	0.000	**1.000**	0.083	0.106	0.236	0.455	0.359	0.430	0.539	0.132
	Purity	0.527	0.647	0.415	0.147	0.150	0.000	0.627	0.485	**1.000**	**1.000**	0.249	0.315	0.371	0.279	0.654	0.370	0.564
GBDP	ACC	0.257	0.321	0.217	0.024	0.022	0.567	0.480	0.334	0.861	0.604	0.146	0.147	0.056	0.042	0.233	0.121	0.293
	NMI	0.000	0.001	0.001	0.021	0.051	0.749	0.000	0.000	0.843	0.674	0.002	0.142	0.006	0.007	0.506	0.662	0.050
	Purity	0.283	0.381	0.399	0.024	0.022	0.567	0.627	0.485	0.907	0.800	0.232	0.212	0.094	0.079	0.711	0.651	0.540
W-GBC	ACC	0.265	0.294	0.339	0.020	0.019	0.308	0.392	0.297	0.379	0.330	0.136	0.115	0.069	0.034	0.100	0.111	0.199
	NMI	0.000	0.000	0.000	0.007	0.017	0.411	0.000	0.000	0.121	0.112	0.001	0.031	0.001	0.001	0.000	0.361	0.009
	Purity	0.283	0.381	0.399	0.020	0.019	0.337	0.627	0.485	0.379	0.330	0.232	0.181	0.089	0.075	0.100	0.226	0.489
LGBQPC	ACC	0.831	**1.000**	0.625	0.622	0.959	0.991	**0.997**	0.605	**1.000**	**1.000**	0.652	0.397	0.815	0.263	0.800	0.577	0.287
	NMI	0.838	**1.000**	0.728	0.678	0.950	0.982	0.968	0.483	**1.000**	**1.000**	0.118	0.326	**0.876**	0.407	0.891	0.755	0.008
	Purity	0.831	**1.000**	0.707	0.622	0.959	**0.993**	**0.997**	0.698	**1.000**	**1.000**	0.908	0.771	0.895	0.290	0.810	0.639	0.491
SNN-DBSCAN	ACC	**1.000**	**1.000**	0.399	0.023	0.019	0.000	–	–	–	–	0.232	0.181	0.106	0.112	0.456	0.251	–
	NMI	**1.000**	**1.000**	0.000	0.024	0.023	0.000	–	–	–	–	0.000	0.000	0.099	0.164	0.000	0.430	–
	Purity	**1.000**	**1.000**	0.399	0.023	0.019	0.000	–	–	–	–	0.232	0.181	0.129	0.154	0.456	0.281	–
GBDBSCAN	ACC	0.998	0.995	0.898	0.910	0.879	0.980	–	–	–	–	0.903	0.390	0.676	0.059	0.100	0.354	–
	NMI	0.996	0.989	0.915	0.935	0.890	0.976	–	–	–	–	0.384	0.340	0.813	0.045	0.667	0.533	–
	Purity	**1.000**	**1.000**	**1.000**	**1.000**	0.945	**0.993**	–	–	–	–	0.954	0.790	0.954	0.059	0.932	0.363	–
GBSC	ACC	0.896	**1.000**	0.375	0.054	0.048	0.020	–	–	–	–	0.090	0.137	0.141	0.103	0.144	0.135	–
	NMI	0.866	**0.999**	0.006	0.194	0.320	0.434	–	–	–	–	0.109	0.214	0.467	0.357	0.534	0.530	–
	Purity	0.896	**1.000**	0.400	0.062	0.093	0.687	–	–	–	–	0.243	0.279	0.360	0.264	0.722	0.356	–

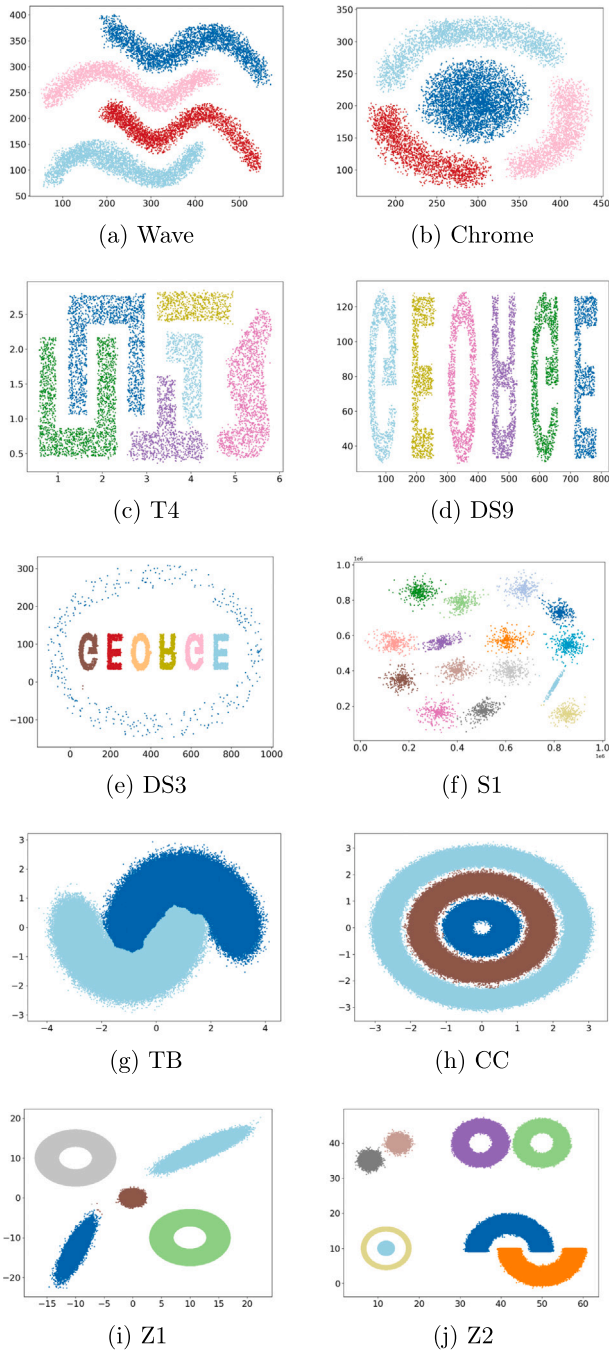


Fig. 2. Clustering of synthetic datasets.

Letter datasets, both designed for handwritten character recognition, our method maintains the best performance across all metrics. On the ORL and YTF face recognition datasets, our method again achieves the best results.

In summary, the proposed method exhibits clear advantages on both synthetic and real-world datasets. This performance stems from the integration of the soft affiliation graph and the dual-constraint connectivity criterion, which together ensure reliable clustering under challenging conditions such as large sample sizes, diverse class distributions, and complex structural patterns. Overall, the experiments confirm the robustness and effectiveness of the method in addressing diverse and large-scale clustering tasks.

4.3. Runtime evaluation

To comprehensively evaluate computational efficiency, we compare the runtime of the proposed method with competitors, as summarized in Table 4 and visualized in Fig. 3.

Table 4
Runtime comparison (in seconds).

Algorithm	Wave	Chrome	T4	DS9	DS3	S1	TB	CC	Z1	Z2	Htru2	CS	Penbased	Letter	ORL	YTF	CT
Ours	**0.330**	**0.320**	**0.321**	**0.283**	**0.335**	**0.301**	**2.253**	**2.021**	**2.023**	**1.940**	**0.333**	**0.292**	**0.315**	**0.334**	0.214	**0.311**	**1.426**
Ours-S	1.479	1.411	1.147	1.160	1.348	1.209	44.032	49.917	45.981	42.518	1.662	1.351	1.350	1.865	**0.011**	1.419	29.061
P-Kmeans	0.523	0.541	0.567	0.492	0.572	0.381	2.782	2.201	5.351	3.126	0.482	0.432	0.662	1.471	0.235	0.842	8.571
GB-USC	0.632	0.513	0.604	0.517	0.628	0.424	3.745	4.350	4.884	5.012	0.759	0.495	0.558	0.647	0.013	0.533	2.238
U-SPEC	11.793	9.931	6.610	6.944	6.909	5.307	72.570	68.440	73.698	83.678	8.991	8.797	8.704	9.784	0.331	9.228	33.445
RP-DBSCAN	1.480	1.519	2.375	1.435	1.476	1.403	176.386	190.632	217.738	229.634	5.357	2.583	2.621	4.592	1.340	1.760	122.349
GBDP	1.491	1.451	1.000	1.000	1.062	0.861	17.923	19.142	20.733	23.665	1.572	1.261	1.281	1.761	0.183	1.242	33.732
W-GBC	4.383	4.042	2.061	1.672	1.691	1.821	356.824	322.512	294.981	376.709	4.571	2.831	2.894	5.801	0.121	2.493	301.813
LGBQPC	0.854	0.766	0.497	0.471	0.566	0.494	57.321	57.555	52.724	48.86	1.187	0.813	0.841	1.508	0.083	0.815	55.615
SNNDBSCAN	2.513	2.194	1.220	1.161	1.212	0.871	–	–	–	–	3.074	1.945	1.919	3.646	0.024	1.697	–
GB-DBSCAN	7.521	6.642	3.851	3.572	3.981	2.392	–	–	–	–	11.523	6.234	6.991	12.061	0.023	5.142	–
GBSC	937.132	704.133	301.312	46.530	310.302	152.460	–	–	–	–	1847.122	685.134	709.001	2402.271	0.071	573.172	–

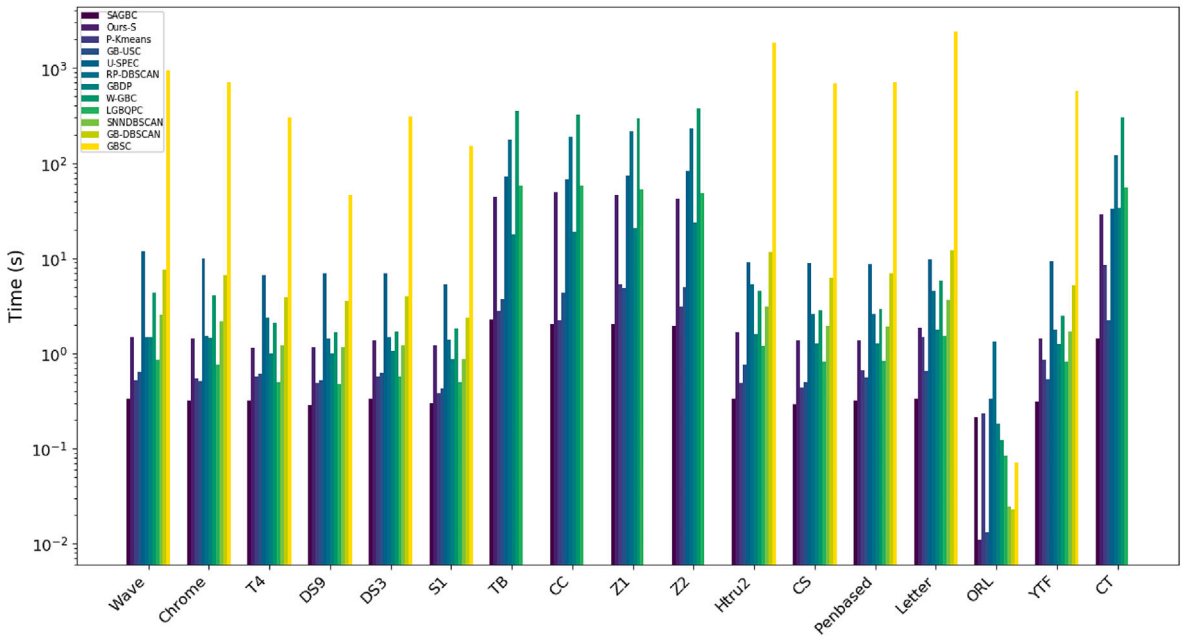


Fig. 3. Runtime comparison.

The proposed method consistently demonstrates superior runtime performance. It achieves the shortest runtime on the majority of datasets, particularly excelling on large-scale datasets. This efficiency stems from its compact representation of data through granular balls and the sampling-driven design that reduces unnecessary computation.

To further validate the contribution of our multi-threaded implementation, we also evaluate a serial variant of our method, denoted as Ours-S. In most cases, Ours-S exhibits longer runtimes than the multi-threaded version, with the performance gap widening on larger datasets. This contrast highlights the effectiveness of multi-threading in improving scalability and computational efficiency. LGBQPC maintains high efficiency on small datasets, but it is slightly inferior to Ours-S on large datasets. A notable exception occurs with the small-scale ORL dataset (100 samples), where our method runs slightly slower than GB-USC and GB-DBSCAN, which can be attributed to the overhead of thread management when dealing with minimal data volumes. In this case, the fastest runtime is achieved by the serial variant of our method.

Among comparison methods, P-Kmeans and GB-USC show competitive efficiency on large datasets. By contrast, U-SPEC incurs a significant time cost due to its internal spectral decomposition, and GBSC consistently records the highest runtime across almost all datasets. Moreover, on the four million-scale datasets (TB, CC, Z1, and Z2), GB-DBSCAN, GBSC, and SNN-DBSCAN fail to complete execution due to excessive computational and memory demands. Their absence in Fig. 3 reflects this inability to handle very large datasets.

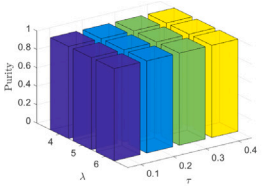
In summary, the proposed method achieves the shortest runtime on the majority of datasets and demonstrates efficient computation.

4.4. Parameter sensitivity analysis

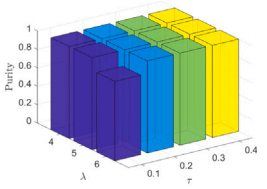
In this study, we evaluate the sensitivity of three primary parameters: the sampling ratio (τ), the number of top-affiliated granular balls (λ), and the search radius scaling factor (γ). Experiments are conducted on two synthetic datasets (DS9 and CC) and two real-world datasets (Htru2 and Penbased) to analyze the impact of clustering metrics and computational overhead on these parameters.

Regarding the joint effect of τ and λ, we vary τ from 0.1 to 0.4 and λ from 4 to 6. To maintain efficiency on large datasets, the number of samples t used for granular ball generation is defined as $t = \min(\text{MAX\_NUM}, n) \times \tau$, where MAX_NUM is 20,000 and n is the total dataset size. The experimental results, shown in Fig. 4, illustrate that the purity metric remains remarkably stable across a wide range of values for both parameters, while runtime gradually increases with larger τ and λ. This suggests that the structure-aware granular ball representation is highly robust in capturing the fundamental topological patterns of the data across varying granularities, whereas the increase in runtime is a direct consequence of the denser affiliation graph and more extensive distance calculations required by larger parameter scales. To ensure a high level of precision while maintaining computational efficiency, we recommend selecting λ from $\{4, 5\}$ and τ within the range of $\{0.2, 0.3\}$.

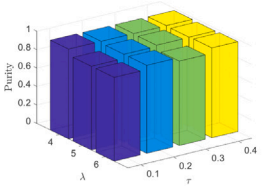
For the search radius scaling factor γ, we vary its value from 1.5 to 3.0, with the results presented in Fig. 5. The method shows robust performance across datasets: clustering accuracy is largely unaffected, and runtime is only slightly influenced. Notably, $\gamma = 2.0$ consistently achieves optimal or near-optimal purity across all evaluated datasets. Therefore, we recommend $\gamma = 2.0$ as a default setting that provides reliable spatial connectivity without introducing additional computational cost.



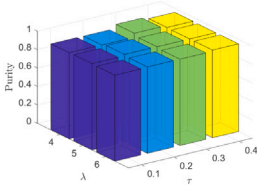
(a) Purity on DS9 of λ and τ



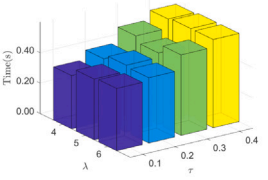
(b) Purity on CC of λ and τ



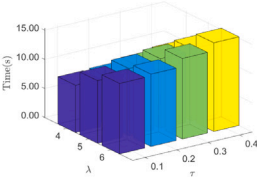
(c) Purity on Htru2 of λ and τ



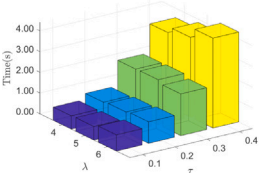
(d) Purity on Penbased of λ and τ



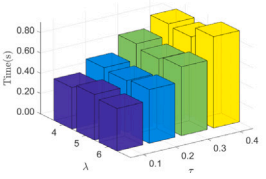
(e) Time on DS9 of λ and τ



(f) Time on CC of λ and τ

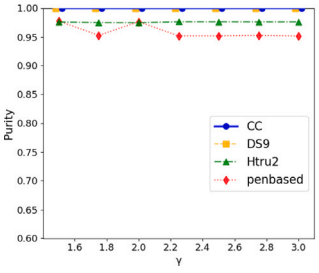


(g) Time on Htru2 of λ and τ

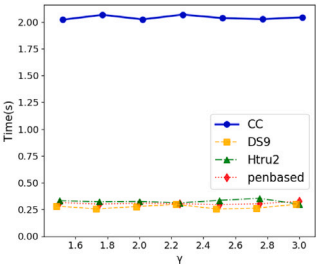


(h) Time on Penbased of λ and τ

Fig. 4. Effect of parameters λ and τ.



(a) Purity on datasets of γ



(b) Time on datasets of γ

Fig. 5. Effect of parameters γ.

4.5. Ablation experiments

To evaluate the contribution of each key component in the proposed framework, we construct four simplified model variants: w/o SR (removing the structure-aware radius), w/o SA (replacing the soft affiliation graph with hard partitioning), w/o STR (removing the structural similarity constraint), and w/o SPA (removing the spatial proximity constraint). Clustering results of the full model and its ablated variants on two synthetic datasets (T4 and TB) and two real-world datasets (YTF and CS) are reported in Table 5, with the best scores highlighted in bold.

Table 5
Ablation experiment results on selected datasets.

Algorithm	Metric	T4	TB	YTF	CS
Ours	ACC	**1.000**	**0.997**	**0.620**	**0.434**
	NMI	**1.000**	**0.973**	**0.848**	**0.384**
	Purity	**1.000**	**0.997**	**0.858**	**0.850**
w/o SR	ACC	0.460	0.501	0.368	0.397
	NMI	0.575	0.000	0.576	0.327
	Purity	0.460	0.501	0.432	0.771
w/o SA	ACC	0.000	0.000	0.000	0.000
	NMI	0.000	0.000	0.000	0.000
	Purity	0.244	0.501	0.069	0.692
w/o STR	ACC	0.107	0.028	0.356	0.054
	NMI	0.472	0.250	0.757	0.259
	Purity	0.828	0.934	0.788	0.803
w/o SPA	ACC	0.911	0.501	0.364	0.400
	NMI	0.954	0.000	0.544	0.345
	Purity	0.911	0.501	0.403	0.771

After removing the structure-aware radius (w/o SR), the model falls back to using a static radius defined by the maximum pairwise distance. This leads to noticeable performance drops, especially on TB, where NMI falls to 0, confirming that a static radius fails to adapt to local density variations.

After removing the soft affiliation graph (w/o SA), each data point is rigidly assigned to only one granular ball. As a consequence, performance collapses across all datasets, with both ACC and NMI reduced to 0, indicating broken connectivity and a loss of boundary flexibility.

The evaluation of the dual-constraint connectivity criterion reveals its critical role in maintaining topological integrity. In the w/o STR variant, granular balls are connected purely based on geometric proximity (spatial-only connectivity), which weakens the graph structure and leads to significantly lower scores on datasets with complex distributions, such as YTF and CS. Conversely, in the w/o SPA variant, connectivity relies solely on structural overlap (structural-only connectivity), which may lead to spurious links between distant but structurally similar regions, thereby distorting the global manifold.

5. Conclusions

This paper proposes a clustering framework based on granular ball modeling that enhances scalability and accuracy in large-scale data clustering. The joint use of a globally structure-aware granular ball set and a Gaussian kernel-based soft affiliation graph enables the framework to capture the comprehensive topological structure of the data efficiently. Furthermore, a dual-constraint connectivity criterion combining spatial proximity and structural similarity effectively constructs connected components of granular balls and uncovers underlying cluster structures.

Extensive evaluations on 17 diverse datasets against 10 baseline algorithms demonstrate that the proposed method achieves superior clustering accuracy and computational efficiency. A systematic ablation study further confirms the indispensable contributions of the four core components to the overall enhancement of clustering precision.

Despite its strengths, the current framework faces challenges in scaling to ultra-large datasets, such as those with hundreds of millions of samples, due to the computational overhead of constructing the full affiliation matrix. To address this limitation, future work will focus on investigating approximation strategies that select representative data points and representative granular balls during the soft affiliation matrix construction.

CRediT authorship contribution statement

Qijia Wang: Writing – original draft, Software, Methodology, Conceptualization. **Mingjing Du:** Writing – original draft, Supervision, Project administration, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is supported by the Qinglan Project of Jiangsu Province of China, the National Natural Science Foundation of China (No. 62006104), Postgraduate Research & Practice Innovation Program of Jiangsu Normal University (No. 2024XKT2626).

Data availability

Data will be made available on request.

References

[1] R. Corizzo, G. Pio, M. Ceci, et al., Dencast: distributed density-based clustering for multi-target regression, J. Big Data 6 (1) (2019) 43.
[2] A. Fahad, N. Alshatri, Z. Tari, et al., A survey of clustering algorithms for big data: taxonomy and empirical analysis, IEEE Trans. Emerg. Topics Comput. 2 (3) (2014) 267–279.
[3] Y. Chen, L. Zhou, S. Pei, et al., Knn-block dbscan: fast clustering for large-scale data, IEEE Trans. Syst, Man, Cybern, Syst. 51 (6) (2019) 3939–3953.
[4] S. Bulusu, V. Gandikota, A. Mazumdar, et al., Robust distributed clustering with redundant data assignment, IEEE Trans. Inf. Theory 71 (4) (2025) 2888–2908.
[5] G. Luo, X. Luo, T.F. Gooch, A parallel dbscan algorithm based on spark, in: 2016 IEEE Int. Conf. Big Data and Cloud Computing, Social Computing and Networking, Sustainable Computing and Communications, IEEE, 2016, pp. 548–553.
[6] W.-Y. Chen, Y. Song, H. Bai, C.-J. Lin, E.Y. Chang, et al., Parallel spectral clustering in distributed systems, IEEE Trans. Pattern Anal. Mach. Intell. 33 (3) (2010) 568–586.
[7] S. Xia, Y. Liu, X. Ding, et al., Granular-ball computing classifiers for efficient, scalable and robust learning, Inf. Sci. 483 (2019) 136–152.
[8] S. Xia, D. Peng, D. Meng, et al., Ball k-means: fast adaptive clustering with no bounds, IEEE Trans. Pattern Anal. Mach. Intell. 44 (1) (2020) 87–99.
[9] S. Xia, X. Dai, G. Wang, et al., An efficient and adaptive granular-ball generation method in classification problem, IEEE Trans. Neural Netw. Learn. Syst. 35 (4) (2022) 5319–5331.
[10] S. Xia, J. Xie, G. Wang, Gbc: An efficient and adaptive clustering algorithm based on granular-ball, arXiv preprint arXiv:2205.14592, 2022.
[11] J. Xie, C. Hua, S. Xia, et al., W-gbc: an adaptive weighted clustering method based on granular-ball structure, in: Proc. IEEE Int. Conf. Data Eng, IEEE, 2024, pp. 914–925.
[12] J. Xie, S. Xia, G. Wang, et al., Gbmst: An efficient minimum spanning tree clustering based on granular-ball computing, arXiv preprint arXiv:2303.01082, 2023.
[13] M. Hefeeda, F. Gao, W. Abd-Almageed, Distributed approximate spectral clustering for large-scale datasets, in: Proc. 21st Int. Symp. High-Performance Parallel and Distributed Computing, 2012, pp. 223–234.
[14] M.A. Ben HajKacem, C.E. Ben N'cir, N. Essoussi, Scalable random sampling k-prototypes using spark, in: Proc. Int. Conf. Big Data Analytics and Knowledge Discovery, Springer, 2018, pp. 317–326.
[15] M. Gowanlock, C.M. Rude, D.M. Blair, J.D. Li, V. Pankratius, A hybrid approach for optimizing parallel clustering throughput using the GPU, IEEE Trans. Parallel Distrib. Syst. 30 (4) (2019) 766–779.
[16] H. Song, J. Lee, Rp-dbscan: a superfast parallel dbscan algorithm based on random partitioning, in: Proc. Int. Conf. Management of Data, ACM, 2018, pp. 1173–1187.
[17] T. Hess, R. Visbord, S. Sabato, Fast distributed k-means with a small number of rounds, in: Proc. Int. Conf. Artificial Intelligence and Statistics, PMLR, 2023, pp. 850–874.
[18] D. Huang, C.-D. Wang, J.-S. Wu, et al., Ultra-scalable spectral clustering and ensemble clustering, IEEE Trans. Knowl. Data Eng. 32 (6) (2019) 1212–1226.
[19] Z. Kang, X. Xie, B. Li, E. Pan, CDC: a simple framework for complex data clustering, IEEE Trans. Neural Networks Learn. Syst. 36 (7) (2025) 13177–13188.
[20] X. Hu, Y. Jiang, W. Pedrycz, Z. Deng, J. Gao, Y. Tang, Automated cluster elimination guided by high-density points, IEEE Trans. Cybern. 55 (4) (2025) 1717–1730.
[21] J. Zhang, R. Fan, H. Tao, J. Jiang, C. Hou, Constrained clustering with weak label prior, Front. Comput. Sci. 18 (3) (2024) 183338.
[22] Q. Xie, Q. Zhang, S. Xia, F. Zhao, C. Wu, G. Wang, et al., GBG++: a fast and stable granular ball generation method for classification, IEEE Trans. Emerg. Top. Comput. Intell. 8 (2) (2024) 2022–2036.
[23] Z. Jia, Z. Zhang, W. Pedrycz, Generation of granular-balls for clustering based on the principle of justifiable granularity, IEEE Trans. Cybern. 55 (4) (2025) 1687–1700.
[24] Z. Jia, Z. Zhang, W. Pedrycz, LGBQPC: local granular-ball quality peaks clustering, arXiv preprint arXiv:2505.11359, 2025.
[25] D. Cheng, Y. Li, S. Xia, et al., A fast granular-ball-based density peaks clustering algorithm for large-scale data, IEEE Trans. Neural Netw. Learn. Syst. 35 (12) (2024) 17202–17215.
[26] D. Cheng, C. Zhang, Y. Li, et al., A fast granular-ball based dbscan clustering algorithm, Inf. Sci. 674 (2024) 120731.
[27] J. Xie, W. Kong, S. Xia, et al., An efficient spectral clustering algorithm based on granular-ball, IEEE Trans. Knowl. Data Eng. 35 (9) (2023) 9743–9753.
[28] J. Xie, M. Dai, S. Xia, et al., An efficient fuzzy stream clustering method based on granular-ball structure, in: Proc. IEEE Int. Conf. Data Eng, IEEE, 2024, pp. 901–913.
[29] D. Cheng, S. Liu, S. Xia, et al., Granular-ball computing-based manifold clustering algorithms for ultra-scalable data, Expert Syst. Appl. 247 (2024) 123313.
[30] A. Quadir, M. Sajid, M. Tanveer, Granular ball twin support vector machine, IEEE Trans. Neural Networks Learn. Syst. 36 (7) (2025) 12444–12453.
[31] A. Quadir, M. Tanveer, Granular ball twin support vector machine with pinball loss function, IEEE Trans. Comput. Soc. Syst. 12 (5) (2025) 3891–3900.
[32] L. Ertöz, M.S. Steinbach, V. Kumar, Finding clusters of different sizes, shapes, and densities in noisy, high dimensional data, in: Proceedings of the Third SIAM International Conference on Data Mining, 2003, pp. 47–58.
[33] J. Xie, R. Girshick, A. Farhadi, Unsupervised deep embedding for clustering analysis, in: Proc. Int. Conf. Machine Learning, PMLR, 2016, pp. 478–487.
[34] A. Strehl, J. Ghosh, Cluster ensembles: a knowledge reuse framework for combining multiple partitions, J. Mach. Learn. Res. 3 (2002) 583–617.
[35] C.D. Manning, P. Raghavan, H. Schütze, Introduction to Information Retrieval, Cambridge Univ. Press, 2008.