

Article

Weighted Deep Embedded Clustering for Robust Representation Learning in Noisy Data

Meaad Altwaimi, Mohamed Maher Ben Ismail * and Ouiem Bchir 

Computer Science Department, College of Computer and Information Sciences, King Saud University, Riyadh 11451, Saudi Arabia; mattuwaimi@ksu.edu.sa (M.A.); obchir@ksu.edu.sa (O.B.)

* Correspondence: mbenismail@ksu.edu.sa

Abstract

Deep clustering methods effectively learn meaningful representations from unlabeled data. Typically, existing approaches treat all samples equally during training, which can lead to unstable clustering in the presence of noise and outliers. In this research, we propose a Reliability-Based Deep Embedded Clustering (RDEC) approach that improves clustering reliability through a novel distance-aware weighting strategy. Specifically, rather than assigning equal importance to all samples, RDEC estimates and exploits the typicality of each instance according to its distance from the cluster centers in the latent space. Accordingly, samples that lie closer to cluster centers are assigned higher weights, while distant, atypical and ambiguous samples are gradually down-weighted using an inverse polynomial function. Moreover, the proposed weighting approach is coupled with a novel Kullback–Leibler divergence objective function to focus on the most representative data instances and guide the clustering process. Furthermore, the resulting latent distances and reliability weights demonstrate that RDEC effectively distinguishes reliable samples from noisy ones, resulting in more stable clustering behavior. RDEC was evaluated on three benchmark image datasets, namely MNIST, USPS, and Fashion-MNIST, under both clean conditions and controlled image corruption settings, including semantic outliers and synthetic perturbations, to assess its robustness during clustering. The experimental results demonstrated that RDEC achieved competitive clustering performance compared with representative clustering methods evaluated under the same experimental setting. In particular, RDEC yielded the best overall performance on the clean USPS dataset, attaining an ACC, NMI, and ARI of 0.8079, 0.7714, and 0.7392, respectively. Moreover, on the clean Fashion-MNIST dataset, RDEC achieved the highest NMI and ARI while maintaining competitive clustering accuracy, demonstrating its effectiveness on a more challenging clustering benchmark. Under these controlled image corruption scenarios, RDEC consistently achieved strong clustering performance across multiple corruption levels on MNIST, USPS, and Fashion-MNIST datasets, demonstrating the effectiveness of the proposed reliability-aware weighting mechanism under diverse image characteristics. The present study focuses on robustness under controlled image-based corruption scenarios, with the evaluation limited to benchmark image datasets.



Academic Editors: Jonathan Cepeda-Negrete and Qianmu Li

Received: 30 June 2026

Revised: 29 July 2026

Accepted: 31 July 2026

Published: 5 August 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and

conditions of the [Creative Commons](#)

[Attribution \(CC BY\)](#) license.

Keywords: deep embedded clustering; robust deep clustering; reliability-aware learning; sample weighting; representation learning; image corruption

1. Introduction

The rapid growth of digital technologies has led to an unprecedented increase in the volume and complexity of data generated across various domains. Extracting meaningful information from these large datasets has become a fundamental challenge in machine learning and data mining [1]. Among the different machine learning paradigms, unsupervised learning plays a particularly important role when labeled data are unavailable or expensive to obtain [1]. Clustering techniques are typically designed to group similar samples into homogeneous categories while separating dissimilar ones without relying on predefined labels [2]. Conventional clustering algorithms, such as K-means [3], have been widely applied in numerous applications, including image analysis, medical diagnosis, document organization, and pattern recognition [2]. However, such methods operate directly on the available feature space and typically rely on handcrafted features and distance-based similarity measures [2,4]. Although effective for relatively simple datasets, their performance often deteriorates when dealing with high-dimensional and nonlinear data, where manually designed features may fail to capture the underlying structure of the data [4]. This limitation has motivated the use of representation learning techniques capable of automatically discovering more informative features in complex data [5].

Recent advances in deep learning underscore the efficacy of representation learning in automatically extracting features from complex datasets [6]. In particular, representation learning techniques have shown that the quality of learned features plays a crucial role in the success of downstream machine learning tasks [7]. Autoencoders have emerged as one of the most widely adopted approaches for unsupervised representation learning because they can learn compact latent representations while preserving important structural information within the data [5,8].

Building upon these advances, deep clustering methods integrate representation learning with clustering optimization, enabling clustering to be performed in a learned latent space that better reflects the intrinsic characteristics of the data [9,10]. By jointly learning feature representations and cluster assignments, deep clustering has demonstrated promising performance across a variety of domains, including computer vision, medical imaging, and pattern recognition [9]. Consequently, deep clustering has become a prominent research area aimed at addressing the inherent limitations of conventional clustering methods.

Despite the substantial progress in deep clustering, the model performance remains susceptible to noisy and ambiguous instances which distort the learned latent representations and interfere with the clustering optimization process. This leads to unstable cluster assignments and a degraded clustering performance [11,12]. In other words, noisy, corrupted, or atypical samples obscure the underlying data structure, thereby degrading clustering results [11,12].

To enhance robustness, various strategies have been proposed across both traditional and deep clustering paradigms. Existing approaches include explicit outlier handling, sparse and robust optimization techniques, model-based noise estimation, reconstruction constraints, adversarial learning, and robust deep clustering frameworks designed to improve clustering stability under challenging conditions [11,13–15]. Nevertheless, recent deep clustering methods [10] optimize a shared objective across all samples and do not adapt the contribution of individual samples according to their estimated reliability during training. As a result, noisy and ambiguous samples may persistently degrade both the learned representations and subsequent clustering assignments.

In this work, we introduce Reliability-Based Deep Embedded Clustering (RDEC), a novel reliability-aware clustering framework that leverages latent-space distances to handle data imperfections. In particular, we adapt an inverse polynomial function to dynamically down-weight noisy samples and minimize their negative impact. This sample

reliability is directly incorporated into the clustering objective by formulating a weighted KL-divergence objective function. To demonstrate the effectiveness of our approach, we evaluate RDEC on benchmark image datasets under clean conditions and controlled image corruption scenarios, including semantic outliers, synthetic perturbations, and class imbalance, demonstrating improved clustering robustness over existing methods.

2. Related Work

Traditional clustering methods have been extensively studied for unsupervised data analysis. One of the earliest and most widely used approaches is K-means, proposed by MacQueen [3], which partitions data into clusters by minimizing within-cluster variance. Other popular methods include hierarchical clustering [16], which constructs a nested cluster structure, and Gaussian Mixture Models (GMMs) [17], which represent data as a mixture of probabilistic distributions. Comprehensive reviews of clustering algorithms can be found in the works of Xu and Wunsch [4] and Jain [2]. Although these methods have been successfully applied in many domains, they operate directly on the available feature space. Consequently, their performance is strongly influenced by the quality of the input representation. For complex datasets such as images and other high-dimensional data, manually designed features may not adequately capture the underlying structure of the data. This limitation has motivated the development of deep clustering approaches that learn feature representations and clustering assignments simultaneously [2,5].

The success of deep learning has led to the emergence of deep clustering methods that combine representation learning with clustering optimization. Unlike traditional clustering algorithms, which operate directly on handcrafted features, deep clustering methods learn latent representations that better capture the underlying structure of the data. Autoencoders have become particularly popular for this purpose because they can compress high-dimensional data into compact representations while preserving important information [5]. The Deep Embedded Clustering approach proposed in [10] first learns a low-dimensional latent representation through autoencoder pretraining and then refines cluster assignments by minimizing the Kullback–Leibler (KL) divergence between a soft assignment distribution and an auxiliary target distribution. The method demonstrated significant improvements over traditional clustering approaches. Although DEC achieves strong clustering performance, the reconstruction objective is discarded during the clustering refinement stage, which may result in the loss of local structural information. To address this limitation, the authors in [18] introduced Improved Deep Embedded Clustering (IDEC) to preserve the reconstruction loss during clustering optimization. Specifically, IDEC has yielded more stable and accurate clustering results by jointly maintaining local structure and refining cluster assignments. Recently, a Deep Clustering via Joint Convolutional Autoencoder Embedding and Relative Entropy Minimization, which combines convolutional feature extraction with clustering optimization, was outlined in [19] to improve the clustering quality. Other studies have explored alternative learning strategies to improve representation quality and clustering performance. For example, Adversarial Deep Embedded Clustering (ADEC) [11] adopted an adversarial learning paradigm to address feature randomness and feature drift during clustering optimization.

More recent studies have further extended deep clustering by addressing practical challenges such as automatic cluster-number estimation and flexible cluster discovery. DeepDPM [20] integrates deep representation learning with nonparametric clustering to infer the number of clusters during training, while DiP-DECK [21] combines deep embedding learning with a statistical dip test to estimate cluster structure without requiring prior knowledge of K .

Recent surveys indicate that deep clustering has evolved into a rapidly growing research area with applications in computer vision, text analysis, medical imaging, and many other domains [9]. Despite the progress achieved so far, several challenges remain. Many deep clustering methods are sensitive to initialization, cluster distributions, and data quality. Furthermore, most existing approaches optimize all samples using the same objective function, regardless of the reliability of individual samples. As a result, noisy, ambiguous, or outlier samples may negatively influence the learned representations and clustering performance. These limitations have motivated increasing interest in robust and noise-aware clustering methods, which are discussed in the following subsection.

The sensitivity of clustering algorithms to noisy, corrupted, and outlier samples has motivated extensive research on robust clustering methods. Early studies focused on improving clustering robustness through explicit outlier handling and robust optimization strategies. In [13], a robust clustering with outlier-sparsity regularization was presented to model outliers as sparse deviations from cluster structures, allowing unreliable observations to be identified and their influence reduced during optimization. Similarly, the robust and sparse clustering approaches in [22,23] introduced sparsity constraints, trimming mechanisms, and adaptive regularization techniques to reduce the impact of anomalous samples and improve clustering stability in noisy environments. Although these methods have demonstrated improved robustness compared with conventional clustering algorithms, they generally operate in the original feature space and may therefore struggle when dealing with complex high-dimensional data. Another related direction is possibilistic clustering, which relaxes the probabilistic membership constraints used in conventional fuzzy clustering. The Possibilistic C-Means (PCM) [24] algorithm assigns typicality values each that reflect how representative sample is of a cluster, allowing atypical observations to exert less influence on the clustering process. Although PCM does not incorporate deep representation learning, it introduced the important idea that samples may contribute differently according to their degree of representativeness, a concept that is related to modern reliability-aware clustering approaches. Another line of research aimed at improving robustness through model-based clustering methods that explicitly incorporate a noise component during parameter estimation. Specifically, the researchers in [14] proposed the Robust Improper Maximum Likelihood Estimator (RIMLE), which models outliers using an improper constant-density component within a Gaussian mixture framework. They further introduced the Optimally Tuned Robust Improper Maximum Likelihood Estimator (OTRIMLE), which automatically selects the noise-component level through an optimization criterion that balances cluster fitting and noise detection.

Adversarial Learning for Robust Deep Clustering [12] was presented to improve stability by enforcing resistance against latent-space perturbations and adversarial disturbances. Similarly, Duet Robust Deep Subspace Clustering (DRDSC) [15] was introduced to improve robustness by explicitly modeling corruption within both the reconstruction and latent self-expression processes, thereby optimizing clustering performance under noisy conditions. Together, these works demonstrate that deep clustering frameworks can be substantially improved by integrating dedicated architectural mechanisms. Recently, researchers have explored strategies that adjust the contribution of individual samples according to their estimated quality or outlier likelihood. Deep Clustering based Fair Outlier Detection (DCFOD) utilizes distances between latent representations and cluster centroids to estimate outlier scores, enhancing feature learning via fairness-aware outlier detection. Unlike traditional outlier-removal approaches, DCFOD demonstrates that latent-space information can successfully differentiate reliable and unreliable samples. However, its primary objective remains outlier detection rather than explicit clustering optimization.

A recent adaptive hybrid clustering framework [9] combines autoencoder-based feature learning with Mahalanobis-distance outlier detection and iterative sample reweighting to reduce the influence of noisy observations. Although this approach improves robustness by identifying and down-weighting anomalies, its weighting process remains dependent on a separate outlier-detection stage and discrete decision criteria.

Despite these advancements in model-based robustness and adaptive sample selection, most existing methods still rely on explicit outlier removal, adversarial regularization, or explicit corruption modeling. Comparatively few studies incorporate continuous sample reliability estimates directly into the clustering objective function while preserving the full dataset.

3. Reliability-Based Deep Embedded Clustering

In this section, we outline the Reliability-Based Deep Embedded Clustering (RDEC) approach, which jointly learns latent representations and cluster assignments by leveraging an autoencoder-based embedding and refining clusters through soft assignments. To mitigate the impact of data imperfections, RDEC introduces a distance-aware weighting mechanism that emphasizes samples near cluster centers while down-weighting distant ones, which are typically noisy or ambiguous. Consequently, samples closer to cluster centers are considered more informative and receive greater importance during optimization, whereas distant or ambiguous samples are progressively down-weighted using an inverse polynomial function. This weighting mechanism ensures that unreliable samples exert minimal influence on the optimization process, leading to more stable and accurate cluster formation. Finally, we incorporate these weights into a weighted Kullback–Leibler (KL) divergence objective function, enabling the clustering process to focus on representative samples while reducing the influence of noisy or outlier observations. The overall RDEC framework is depicted in Figure 1.

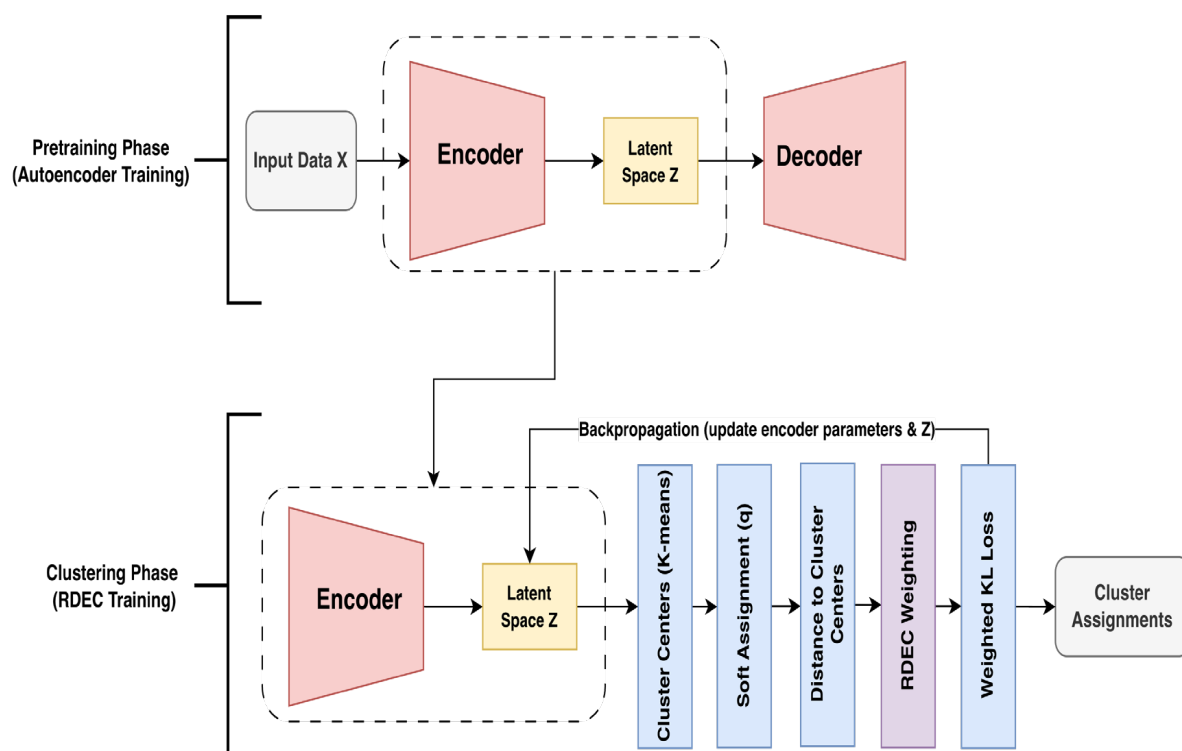


Figure 1. Overview of the proposed Reliability-Based Deep Embedded Clustering (RDEC) approach.

As it can be seen, RDEC operates in two stages: (i) an autoencoder pretraining phase intended to learn latent representations and (ii) a clustering phase where the obtained embeddings are iteratively refined according to the optimization process. In addition, the soft assignments are computed in the latent space. Accordingly, the reliability aware mechanism assigns higher weights to the typical samples that are closer to the cluster centers. On the other hand, the influence of noisy samples and/or outliers that are distant from the cluster centers is reduced. In addition, a novel weighted KL-divergence loss is designed to simultaneously optimize both the encoder and the cluster centers during backpropagation.

Given a dataset, a nonlinear mapping is first learned as follows:

$$z_i = f_\theta(x_i) \quad (1)$$

where f_θ is the encoder of a stacked denoising autoencoder and $z_i \in \mathbb{R}^d$ is the corresponding latent representation. The autoencoder is pretrained and fine-tuned to capture meaningful structure in the data before clustering [5,8]. The cluster centers, $\{\mu_j\}_{j=1}^K$, are initialized using K-means [3] algorithm in the latent space. Accordingly, the soft assignment of a given sample, z_i , to the cluster is computed using

$$q_{ij} = \frac{(1 + \|z_i - \mu_j\|^2)^{-1}}{\sum_{j'} (1 + \|z_i - \mu_{j'}\|^2)^{-1}} \quad (2)$$

In addition, an auxiliary target distribution is defined as follows to sharpen cluster assignments:

$$p_{ij} = \frac{q_{ij}^2 / \sum_i q_{ij}}{\sum_{j'} (q_{ij'}^2 / \sum_i q_{ij'})} \quad (3)$$

In order to improve the clustering robustness to noise, a distance-based weighting mechanism is introduced to reflect the reliability of each sample with respect to the learned cluster structure. Specifically, the distance between each embedding and its nearest cluster center is computed as

$$d_i = \min_j \|z_i - \mu_j\| \quad (4)$$

Moreover, in order to ensure stability across batches, the obtained distances are normalized using

$$d_i^{\text{norm}} = \frac{d_i}{\sigma} \quad (5)$$

where σ is a global scaling factor estimated via an exponential moving average as follows:

$$\sigma \leftarrow \alpha\sigma + (1 - \alpha)\text{mean}(d_i) \quad (6)$$

Accordingly, an inverse polynomial weighting function is used to incorporate the proposed distance-aware reliability into the optimization process. More specifically, the proposed reliability weights are obtained using

$$w_i = \frac{1}{1 + (d_i^{\text{norm}})^p} \quad (7)$$

where $p > 0$ controls the decay rate. This inverse polynomial weighting function assigns higher weights to samples close to cluster centers (typical samples) and progressively reduces the influence of distant samples, which are more likely to be noisy or ambiguous. One should also mention that the weights are constrained within a bounded range

$w_i \in [w_{\min}, 1]$ in order to ensure stability and prevent vanishing gradients during optimization. This formulation ensures that reliable/typical samples contribute more strongly to the optimization process, while reducing the impact of uncertain samples.

The considered inverse polynomial weighting function provides a gradual and controllable reduction in sample importance as the distance from the cluster center increases. Specifically, the inverse polynomial formulation preserves a meaningful contribution from distant samples that still encapsulate useful clustering information while continuing to reduce the influence of highly distant and potentially unreliable instances. This behavior is consistent with the objective of the proposed framework, which aims at retaining all samples during training while adaptively limiting the impact of noisy or ambiguous observations.

The proposed weighting strategy involves an inherent trade-off between clustering robustness and the preservation of atypical samples. By design, RDEC reduces the influence of distant or ambiguous instances during optimization to improve the stability of the learned latent representations and cluster centers. Consequently, the framework is intended to prioritize representative samples for clustering rather than emphasize rare or anomalous observations. Importantly, down-weighting does not remove these samples from the training process; instead, they continue to contribute to the optimization with reduced influence through the bounded weighting function. As a result, atypical samples remain represented in the learned latent space but are not explicitly identified or optimized as anomalies. Detecting or characterizing rare events constitutes a different learning objective that typically requires dedicated anomaly detection or outlier analysis techniques. Therefore, the primary goal of RDEC is to improve clustering robustness and representation learning under noisy conditions, whereas extending the framework to jointly support anomaly detection represents an interesting direction for future research.

RDEC optimizes the Kullback–Leibler (KL) divergence between the target distribution P and the soft assignment Q as follows:

$$L_{\text{RDEC}} = \frac{1}{n} \sum_{i=1}^n w_i \cdot \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}} \quad (8)$$

where n denotes the total number of samples, i indexes data points, and j indexes clusters. The weight assigned to sample i , denoted by w_i , reflects its reliability based on its proximity to the cluster centers. On the other hand, the soft assignment, q_{ij} , represents the probability of assigning sample i to cluster j , computed using a Student's t -distribution, while p_{ij} denotes the corresponding target distribution that emphasizes high-confidence assignments. The logarithmic term corresponds to the Kullback–Leibler divergence between the target distribution, P , and the soft assignment distribution, Q . In addition, the training proceeds until the change in cluster assignments falls below a predefined threshold δ .

The complete training procedure of the proposed RDEC framework is summarized in Algorithm 1. One should note that the model is optimized using a mini-batch gradient-based optimization procedure, where latent embeddings, cluster assignments, and distance-based reliability weights are updated iteratively. At each iteration, the encoder and cluster centers are jointly refined through backpropagation by minimizing the proposed weighted objective function. The training process continues until convergence, which is defined by a small change in cluster assignments.

Algorithm 1: RDEC (Robust Deep Embedded Clustering)**Input:** Dataset $\mathcal{X}$, number of clusters K , weighting parameter p , convergence threshold δ **Output:** Cluster assignments y , cluster centers μ

Pretrain the autoencoder to learn an initial latent representation

Initialize cluster centers μ using K-means on the latent spaceInitialize the scaling factor σ

Repeat:

Shuffle the dataset and divide it into mini-batches

For each mini-batch $\mathcal{B}$:Compute latent embeddings z_i using (1)Compute soft assignments q_{ij} using (2) and target distribution p_{ij} using (3)Compute distances d_i using (4) and update scaling factor σ using (5)Compute normalized distances $\tilde{d}_i$ using (6)Compute weights w_i using (7)

Update model parameters by minimizing the loss in (8)

Until $\Delta < \delta$

4. Experiments

In this research, three benchmark image datasets, namely, MNIST [25], USPS [26] and Fashion-MNIST [27], were used to assess the proposed Reliability-based Deep Embedded Clustering (RDEC) approach. The experimental evaluation investigates the clustering performance and robustness of RDEC under controlled image corruption scenarios, including semantic outliers, synthetic perturbations, and class imbalance. In particular, MNIST [25] dataset consists of grayscale handwritten digit images of size 28×28 , which were reshaped into 784-dimensional vectors. The USPS [26] dataset contains grayscale digit images of size 16×16 , represented as 256-dimensional numerical vectors. In addition, the Fashion-MNIST [27] dataset comprises grayscale images of clothing articles from ten categories, each with a resolution of 28×28 pixels. Compared with handwritten digit datasets, Fashion-MNIST exhibits greater intra-class variability and higher visual similarity between classes, making it a more challenging benchmark for clustering evaluation. Moreover, RDEC was benchmarked against multiple clustering approaches, including K-means [3], AE + K-means, DEC [10], and IDEC [18]. One should note that AE + K-means deploys K-means in a latent feature space learned using a pretrained autoencoder. Furthermore, the clustering performance was assessed using Hungarian matching-based Accuracy (ACC), along with Normalized Mutual Information (NMI) and Adjusted Rand Index (ARI). In addition to these standard metrics, we conducted a detailed analysis of latent distances and sample weights to gain deeper insight into the behavior of the proposed weighting mechanism.

The selected baselines include conventional clustering (K-means), latent representation learning followed by clustering (AE + K-means), and representative deep embedded clustering methods (DEC and IDEC). These methods share comparable clustering objectives and evaluation protocols, thereby providing a consistent benchmark for assessing the effectiveness of the proposed reliability-aware weighting mechanism. Since the main contribution of this work is the introduction of a reliability-aware sample weighting strategy within a deep embedded clustering framework, the experimental evaluation focuses on methods with comparable clustering formulations, allowing the contribution of the proposed weighting mechanism to be assessed under consistent experimental conditions. Although several more recent clustering-related approaches have been reported in the literature, many are designed for different methodological objectives, including adversarial representation learning, deep subspace clustering, and fairness-aware outlier detection. Be-

cause these methods rely on substantially different optimization strategies and architectural assumptions, direct quantitative comparison would not isolate the specific contribution of the proposed reliability-aware weighting mechanism.

In the following experiments, RDEC encoder was implemented as a stacked denoising autoencoder (SDAE) [28]. As shown in Figure 2, the SDAE architecture can be summarized as [784, 500, 500, 2000, 10] for MNIST [25] and Fashion-MNIST [27], and [256, 500, 500, 2000, 10] for USPS [26]. Accordingly, the encoder maps the input data into a 10-dimensional latent space, while the decoder reconstructs its input using a symmetric structure.

According to the architecture reported in Figure 2, the model was trained in three stages. First, the autoencoder was pretrained for 60 epochs to learn meaningful latent representations. Then, a fine-tuning of 100 epochs was conducted to further refine the learned features. Finally, the clustering phase was performed for 20 epochs. Note that a batch size of 256 was used across all experiments.

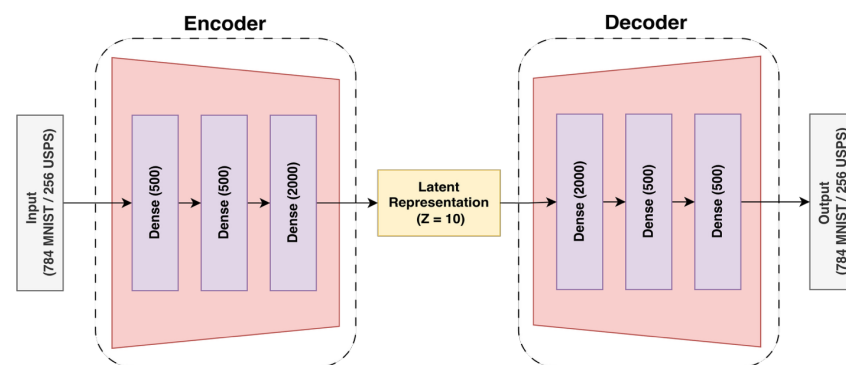


Figure 2. Architecture of the RDEC's stacked denoising autoencoder.

To analyze the behavior of the proposed weighting mechanism, we investigate the impact of the parameter p in (7), which controls the decay rate and determines how strongly distant samples are down-weighted during training. For the noisy setting, an outlier ratio of 10% was deployed.

For the MNIST [25] dataset, the results under clean conditions, shown in Figure 3, show that higher values of p tend to improve clustering performance, indicating that stronger penalization of distant samples enhances representation learning.

However, under noisy conditions with 10% outliers, as reported in Figure 4, moderate values of p yield better performance, suggesting that excessive penalization may reduce robustness in the presence of noise.

For the USPS dataset, a different trend is observed. Under clean conditions as depicted in Figure 5, smaller values of p achieve better performance, reflecting the more compact structure of the latent space.

As shown in Figure 6, this behavior remains consistent under noisy conditions with 10% outliers where lower values of p provide more stable clustering results.

Overall, these observations indicate that the optimal value of p depends on both the dataset characteristics and the noise level. Nevertheless, the proposed RDEC framework demonstrates stable performance across a wide range of values, highlighting its robustness to this hyperparameter.

To ensure a fair and consistent comparison across all subsequent experiments, we fix p to 2 as a default setting. This choice provides a balanced trade-off between robustness and performance across different datasets while avoiding dataset-specific tuning. For optimization, stochastic gradient descent (SGD) with momentum was employed for MNIST [25] and Fashion-MNIST [27], while the Adam optimizer was used for USPS [26] dataset. The corruption level in the denoising autoencoder was set to 0.2. In RDEC, the weighting func-

tion is defined using an inverse polynomial formulation, where the parameter p controls the decay rate, as discussed in the preceding analysis. In addition, the scaling factor was dynamically estimated as the mean distance between embeddings and their nearest cluster centers and updated using an exponential moving average.

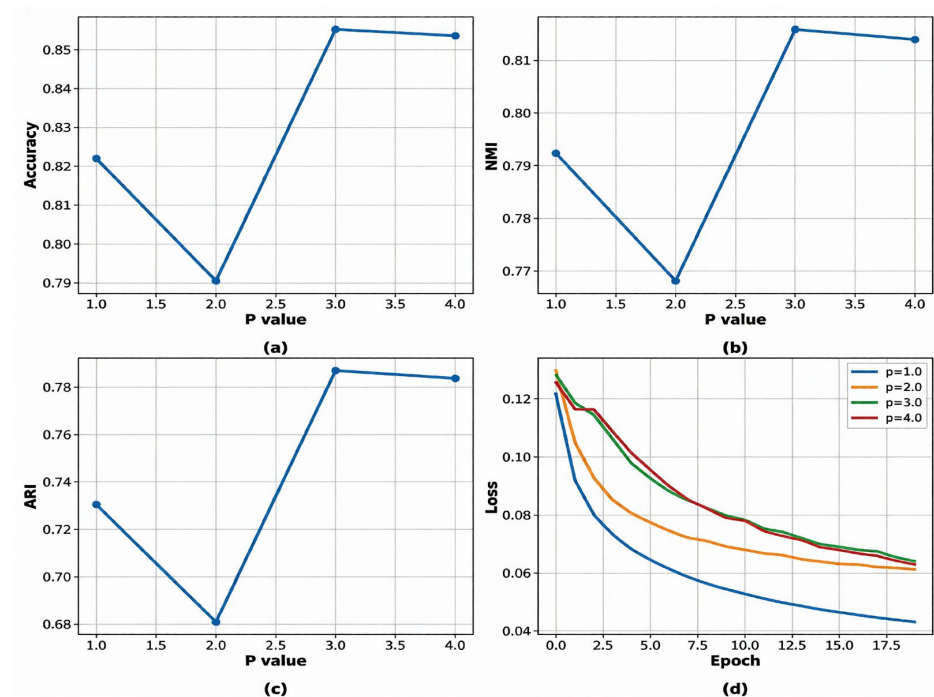


Figure 3. Effect of the weighting parameter p on the clustering performance for Clean MNIST: (a) Accuracy (ACC), (b) Normalized Mutual Information (NMI), (c) Adjusted Rand Index (ARI), and (d) Training loss across epochs.

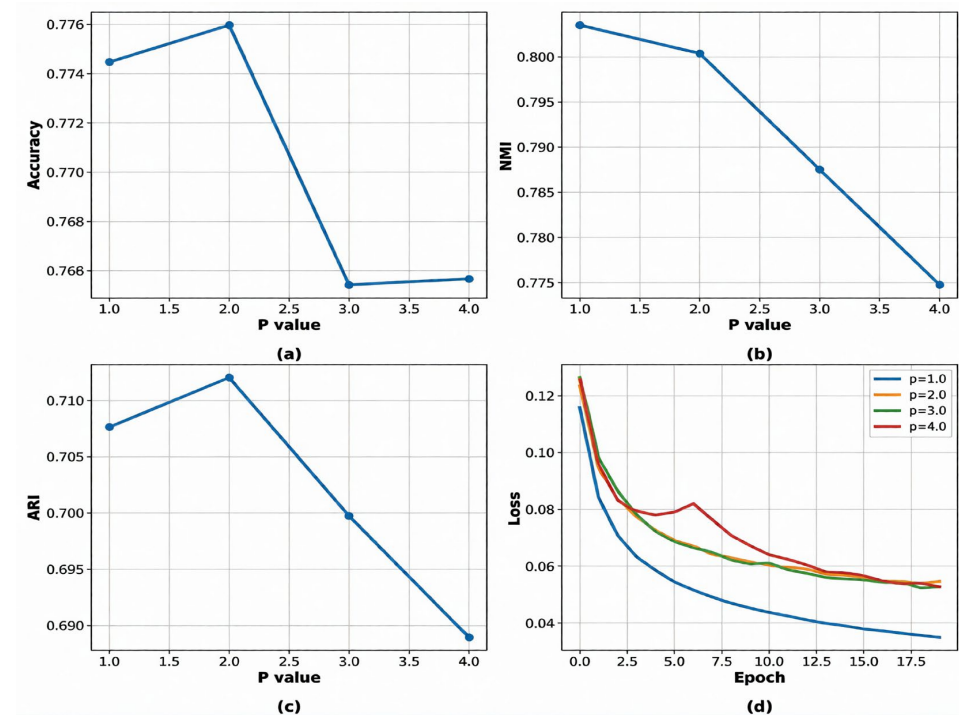


Figure 4. Effect of the weighting parameter p on the clustering performance for Noisy MNIST: (a) Accuracy (ACC), (b) Normalized Mutual Information (NMI), (c) Adjusted Rand Index (ARI), and (d) Training loss across epochs.

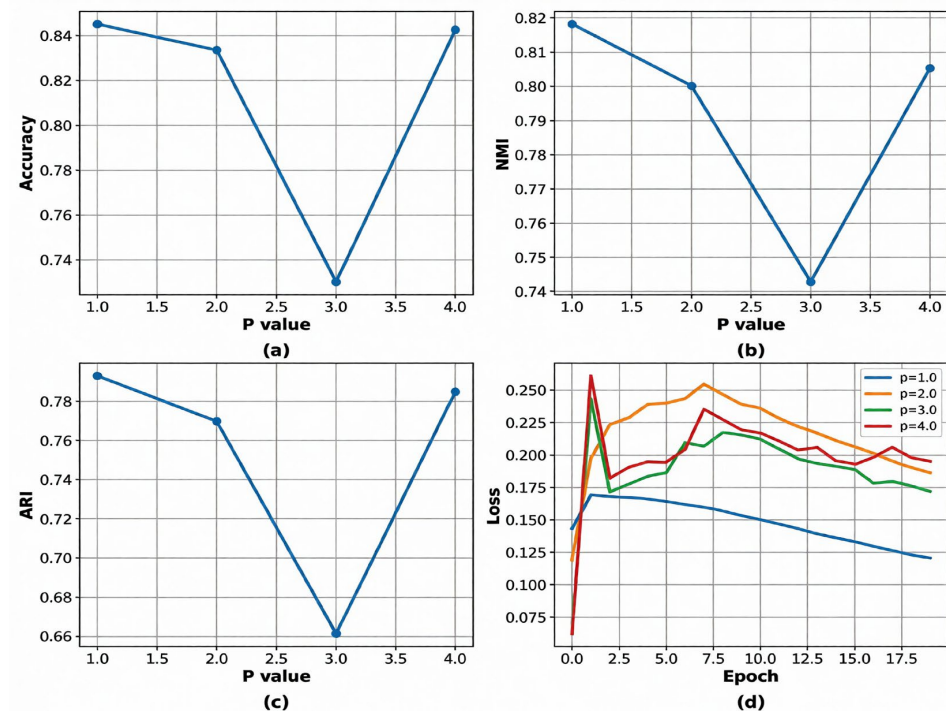


Figure 5. Effect of the weighting parameter p on the clustering performance for Clean USPS: (a) Accuracy (ACC), (b) Normalized Mutual Information (NMI), (c) Adjusted Rand Index (ARI), and (d) Training loss across epochs.

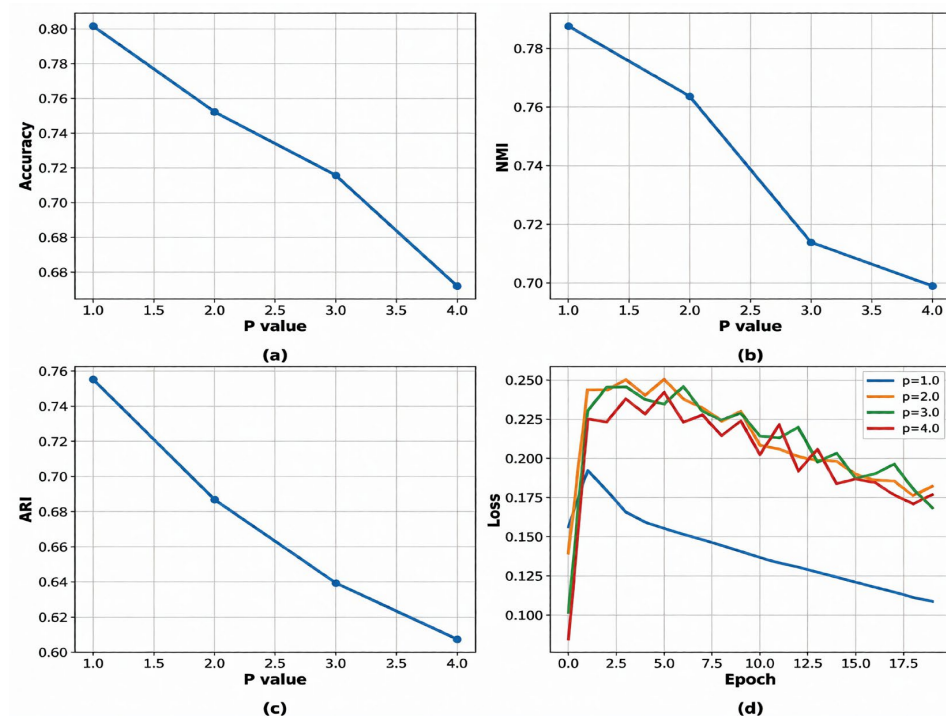


Figure 6. Effect of the weighting parameter p on the clustering performance for Noisy USPS: (a) Accuracy (ACC), (b) Normalized Mutual Information (NMI), (c) Adjusted Rand Index (ARI), and (d) Training loss across epochs.

4.1. Clustering Performance Achieved Using Clean Data

To evaluate the baseline clustering performance, we first assess all methods on the original datasets with no injected noise. MNIST, USPS, and Fashion-MNIST datasets are reported in Table 1, Table 2 and Table 3 respectively.

Table 1. Clustering Performance achieved using clean MNIST.

	ACC	NMI	ARI
K-means [3]	0.5131	0.4906	0.3607
AE + K-means	0.7760	0.7004	0.6368
IDEC [18]	0.7940	0.7293	0.6725
RDEC	0.7920	0.7691	0.6823

Table 1 presents the clustering performance achieved using the clean MNIST [25] dataset. Compared with classical K-means, both AE + K-means, IDEC, and the proposed RDEC yielded substantial improvements across all evaluation metrics, highlighting the effectiveness of deep representation learning for clustering. While IDEC achieved the highest ACC score, the top NMI and ARI values were obtained using RDEC. This indicates stronger cluster consistency and better alignment between the predicted clusters and ground-truth labels. These results suggest that the proposed distance-aware weighting mechanism maintains competitive clustering performance even under clean conditions.

Table 2. Clustering performance achieved using clean USPS.

	ACC	NMI	ARI
K-means [3]	0.6721	0.6135	0.5361
AE + K-means	0.6889	0.6714	0.5660
IDEC [18]	0.7055	0.6773	0.6028
RDEC	0.8079	0.7714	0.7392

Table 2 reports the clustering performance obtained using the USPS [26] dataset. Consistently with the MNIST results, AE + K-means, IDEC along with the proposed RDEC outperformed significantly the classical K-means across all evaluation metrics, demonstrating the effectiveness of deep representation learning for clustering. Among all methods, RDEC achieved the highest ACC, NMI, and ARI scores, indicating stronger cluster quality and better agreement between predicted clusters and ground-truth labels. Compared with IDEC, the proposed RDEC shows substantial improvements across all evaluation metrics. This proves that the introduced distance-aware weighting mechanism succeeded to enhance the clustering effectiveness while preserving stable performance under clean conditions.

Table 3. Clustering performance achieved using clean Fashion-MNIST.

	ACC	NMI	ARI
K-means [3]	0.472	0.5117	0.3476
AE + K-means	0.4811	0.5423	0.3701
IDEC [18]	0.5002	0.5471	0.3737
RDEC	0.4828	0.5643	0.3828

Table 3 reports the clustering performance achieved using the clean Fashion-MNIST [27] dataset. Compared with the handwritten digit datasets, Fashion-MNIST represents a more challenging clustering benchmark because of the greater visual similarity between classes and the higher intra-class variability. As a result, all clustering methods achieved lower performance than that obtained on MNIST and USPS. Nevertheless, the proposed RDEC achieved the highest NMI and ARI scores among all competing methods, indicating improved cluster quality and better agreement between the learned cluster assignments and the ground-truth labels. Although IDEC obtained a slightly higher ACC, the proposed

RDEC maintained competitive accuracy while providing more discriminative latent representations. These results further demonstrate that the proposed distance-aware weighting mechanism remains effective even when applied to more complex image datasets.

To gain deeper insight into the clustering behavior, we further analyze the per-class accuracy of each method.

For the MNIST [25] dataset, a closer examination of the cluster-level results in Figure 7 shows that the clustering behavior varies noticeably across different clusters. Compared with classical K-means, the deep clustering methods generally yield substantially improved performance across most clusters. Particularly, the proposed RDEC demonstrates strong performance for the clusters C0, C2, C3, C6, C7, and C9. On the other hand, AE + K-means performed the best in clusters C1 and C5, while IDEC achieved the strongest performance in cluster C8. Interestingly, the cluster C4 appears to be more challenging for the deep clustering methods, with K-means slightly outperforming the competing approaches. These results suggest that the clustering effectiveness depends on the characteristics of individual clusters and that the proposed reliability-aware weighting mechanism is particularly beneficial for several complex cluster structures, leading to more consistent performance across diverse data distributions.

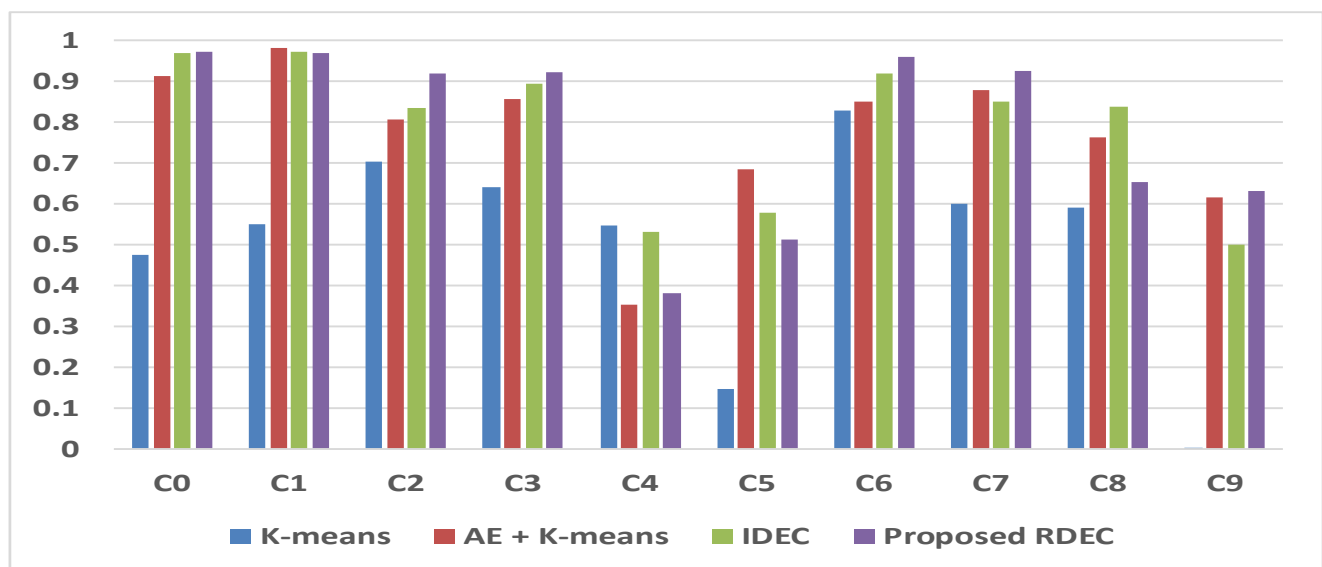


Figure 7. Per-Class Accuracy on Clean MNIST.

Extending this analysis to the USPS [26] dataset, a broadly similar pattern can be observed in Figure 8. Compared with classical K-means, an improved performance across most clusters was achieved using the deep clustering methods, showcasing the benefit of representation learning for the clustering task. The proposed RDEC yielded particularly strong results in clusters C0, C2, C3, C5, C6, and C7. This confirmed the efficiency of the introduced distance-aware weighting mechanism in enhancing the clustering discrimination for several cluster structures. On the other hand, IDEC achieved a competitive performance in cluster C4, while AE + K-means performed strongly in C1 and C9. One should also note that the clustering behavior was not consistent across all clusters, as some clusters proved to be more challenging than others. Overall, these results further suggest that clustering effectiveness depends not only on the clustering method itself but also on the underlying characteristics of each cluster.

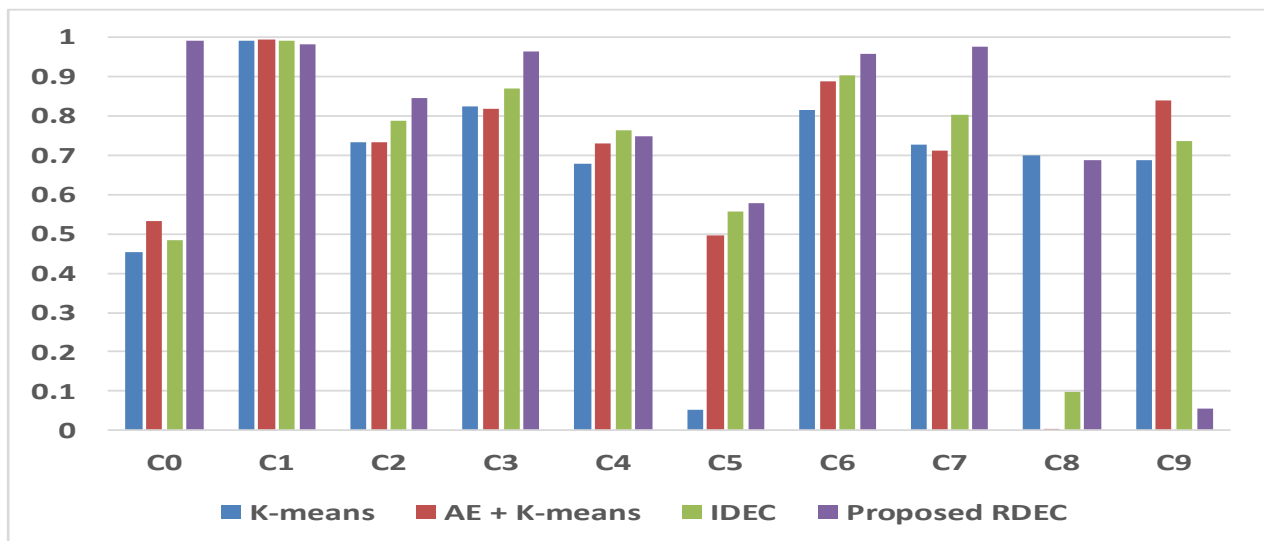


Figure 8. Per-Class Accuracy on Clean USPS.

Figure 9 presents the per-class clustering accuracy obtained on the clean Fashion-MNIST [27] dataset. Similar to the previous analyses, the deep clustering methods generally outperform the classical K-means approach across most classes, demonstrating the benefit of learning latent representations. The proposed RDEC achieves the highest accuracy for classes C0, C1, C2, and C8, whereas IDEC performs best on classes C3, C4, C6, and C9. In contrast, AE + K-means yields the highest accuracy for classes C5 and C7. Although no single method consistently achieves the highest accuracy across all classes, the proposed RDEC maintains competitive performance on multiple categories while achieving the highest overall NMI and ARI reported in Table 3. These results further demonstrate the effectiveness of the proposed reliability-aware weighting mechanism in learning robust and discriminative latent representations.

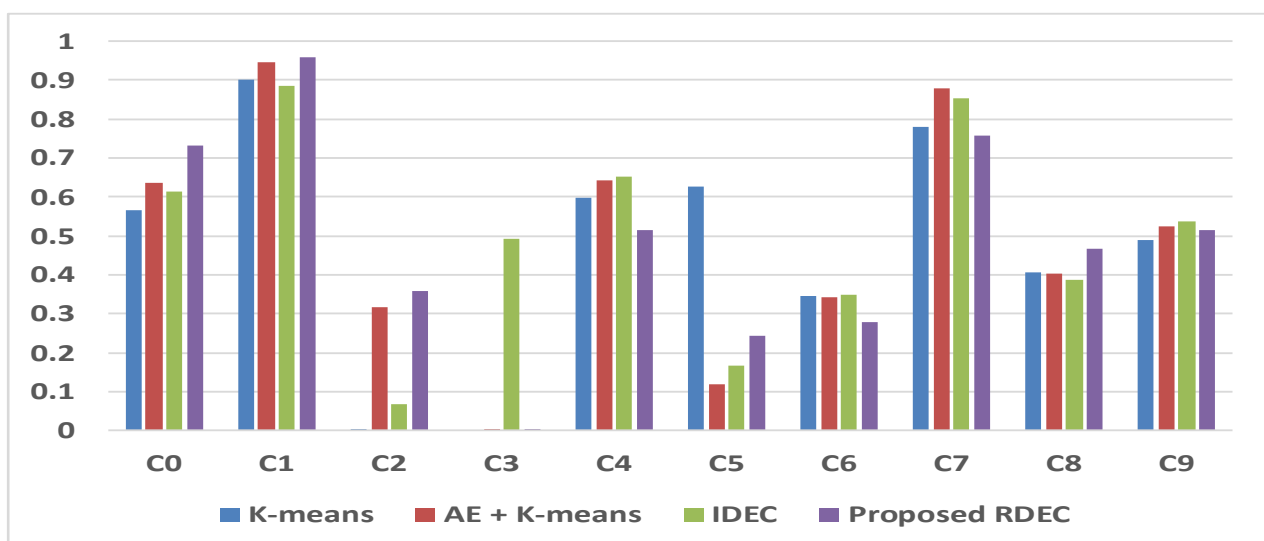


Figure 9. Per-Class Accuracy on Clean Fashion-MNIST.

4.2. Clustering Performance Achieved Using Noisy Data

In order to evaluate the robustness of RDEC under controlled image corruption scenarios, controlled corruption levels were introduced into the benchmark image datasets. For MNIST [25], semantic outliers were generated by incorporating samples from the

Fashion-MNIST dataset [27], introducing structurally different patterns that do not belong to the original digit classes. In contrast, for USPS [26], noise was introduced through synthetic perturbations, including random noise injection and geometric transformations, resulting in more challenging and less structured outliers. For the Fashion-MNIST [27] dataset, semantic outliers were generated by incorporating handwritten digit samples from the MNIST dataset [25], creating visually distinct samples that do not belong to the original clothing categories. Moreover, the noise ratio was varied from 1% to 30%, where the noise ratio denotes the proportion of injected outliers relative to the total dataset size. To provide a visual intuition of these differences, representative examples of inlier and outlier samples are shown in Figure 10.

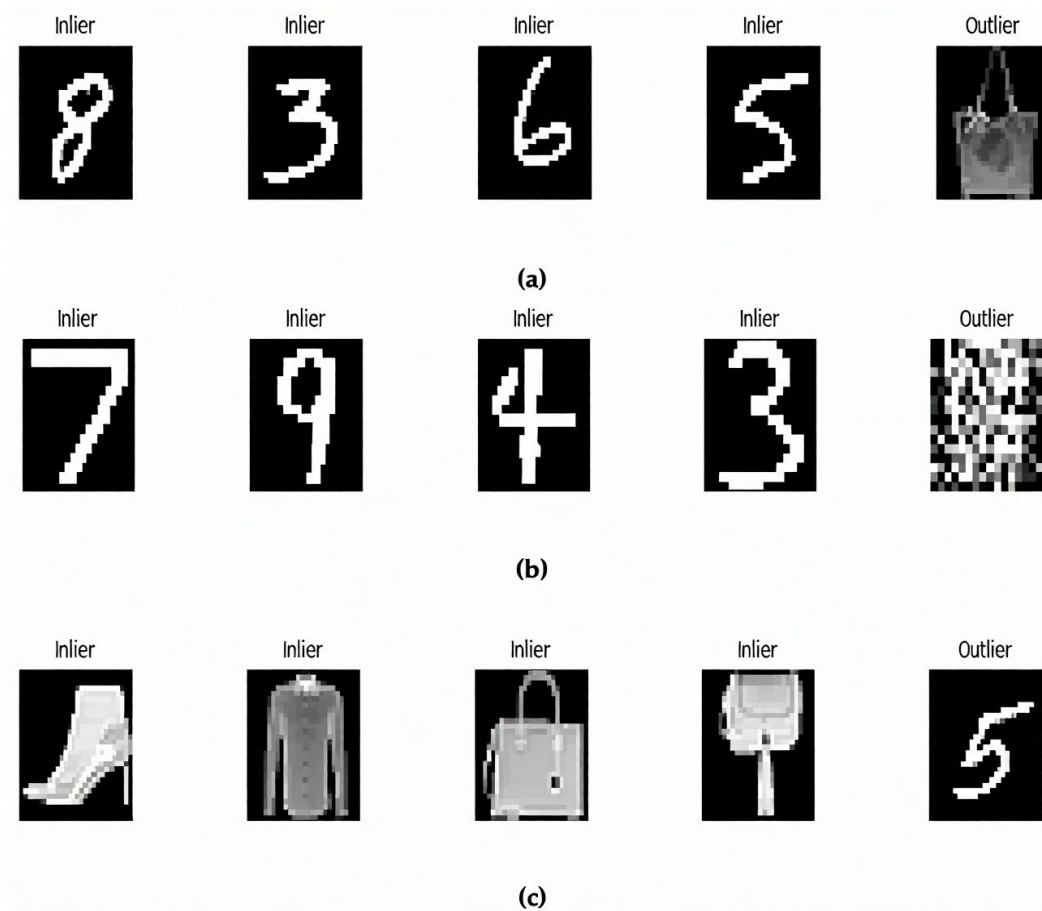


Figure 10. Samples of inlier and outlier data used in the noisy experiments: (a) MNIST [25] inlier samples together with semantic outliers from the Fashion-MNIST [27] dataset, (b) USPS [26] inliers samples along with synthetically corrupted outliers generated using random perturbations and geometric transformations, and (c) Fashion-MNIST [27] inlier samples together with semantic outliers from the MNIST [25] dataset.

The clustering performance of the different methods under varying noise levels is illustrated in Figure 11, Figure 12 and Figure 13 for MNIST, USPS and Fashion-MNIST respectively.

The results on MNIST [25] under semantic noise are shown in Figure 11. As it can be seen, at low noise levels (1–2%), DEC and IDEC achieve the highest performance across all metrics, indicating strong clustering capability when the data remains mostly clean. However, as the noise level increases, a gradual decline can be observed across all methods. In this setting, the proposed RDEC becomes more competitive, particularly in the moderate noise range (5–10%), where it achieves comparable or slightly improved results in terms of NMI and ARI. This suggests that the weighting mechanism helps reduce the influence of

mismatched samples introduced from Fashion-MNIST. At higher noise levels (20–30%), the performance of all methods degrades more noticeably, although RDEC maintains relatively stable behavior compared to several baselines.

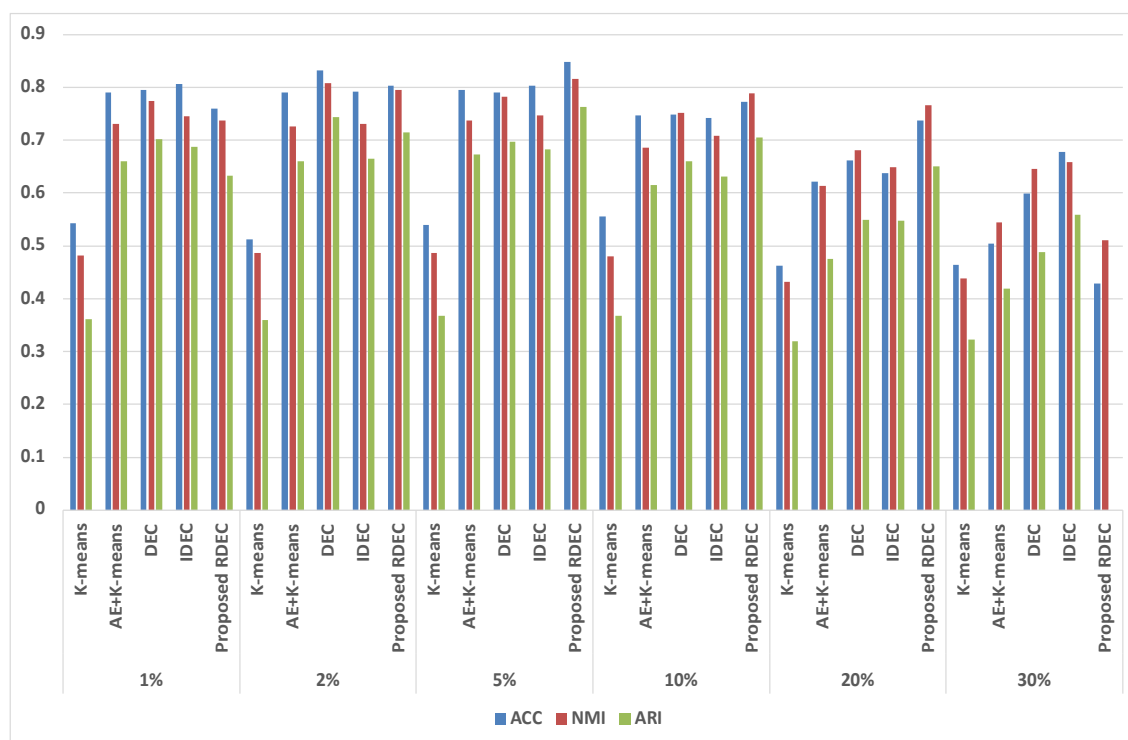


Figure 11. Clustering performance obtained using MNIST under varying levels of semantic noise (1–30%).

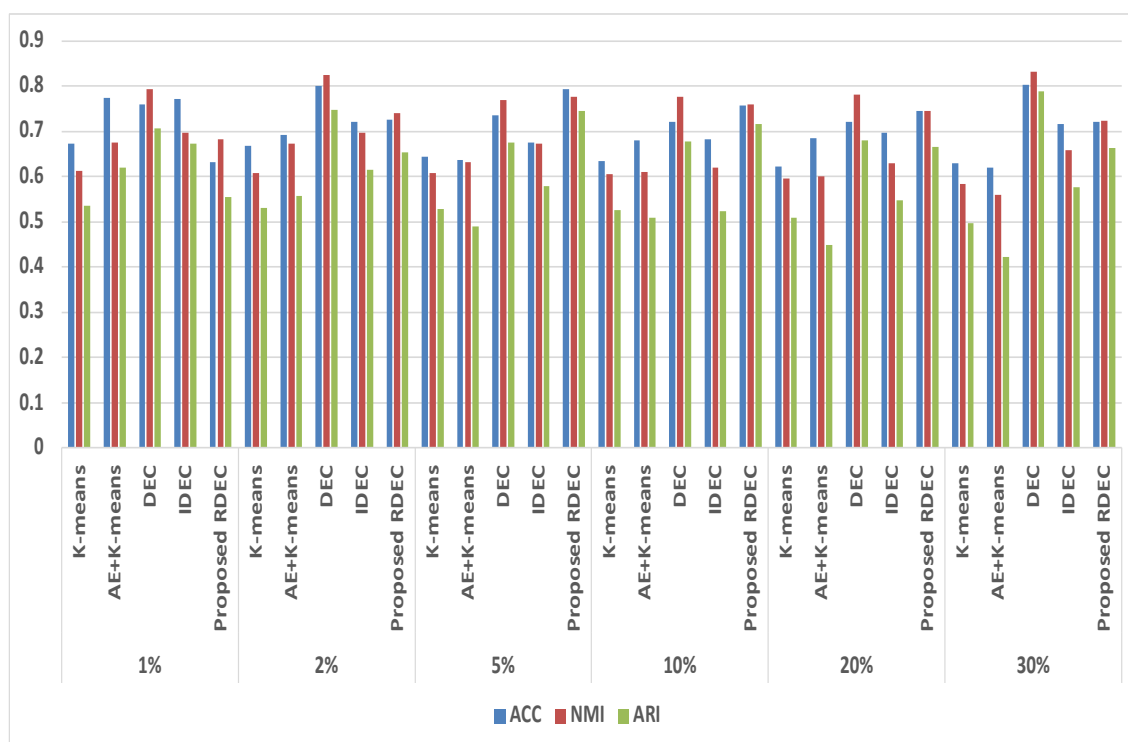


Figure 12. Clustering performance achieved using USPS under varying levels of synthetic noise (1–30%).

A similar pattern can be observed for the USPS [26] dataset. As illustrated in Figure 12, while DEC performs strongly at low noise levels, its advantage gradually diminishes as the level of synthetic noise increases. In contrast, RDEC shows more stable behavior under these conditions, particularly in the mid-noise range (5–20%), where it remains competitive with or outperforms several baseline methods. This improvement is particularly evident under synthetic perturbations, which introduce less structured and more challenging distortions compared to semantic noise. At higher noise levels, all methods show a noticeable decline, but RDEC still preserves a reasonable level of clustering quality.

Figure 13 presents the clustering performance obtained using the Fashion-MNIST [27] dataset under varying levels of semantic noise. Similar to the MNIST and USPS experiments, DEC achieves the strongest performance at very low noise levels (1–5%), where the latent representations remain only slightly affected by the injected outliers. As the corruption level increases, however, the proposed RDEC becomes increasingly competitive. In particular, at 10% and 20% noise, RDEC achieves the highest overall clustering performance across the evaluation metrics, while at 30% noise it maintains the highest NMI and ARI values despite the substantial degradation observed for all methods. These results suggest that the proposed reliability-aware weighting mechanism becomes increasingly beneficial as the influence of noisy samples grows, allowing the clustering process to preserve more discriminative latent representations under challenging conditions.

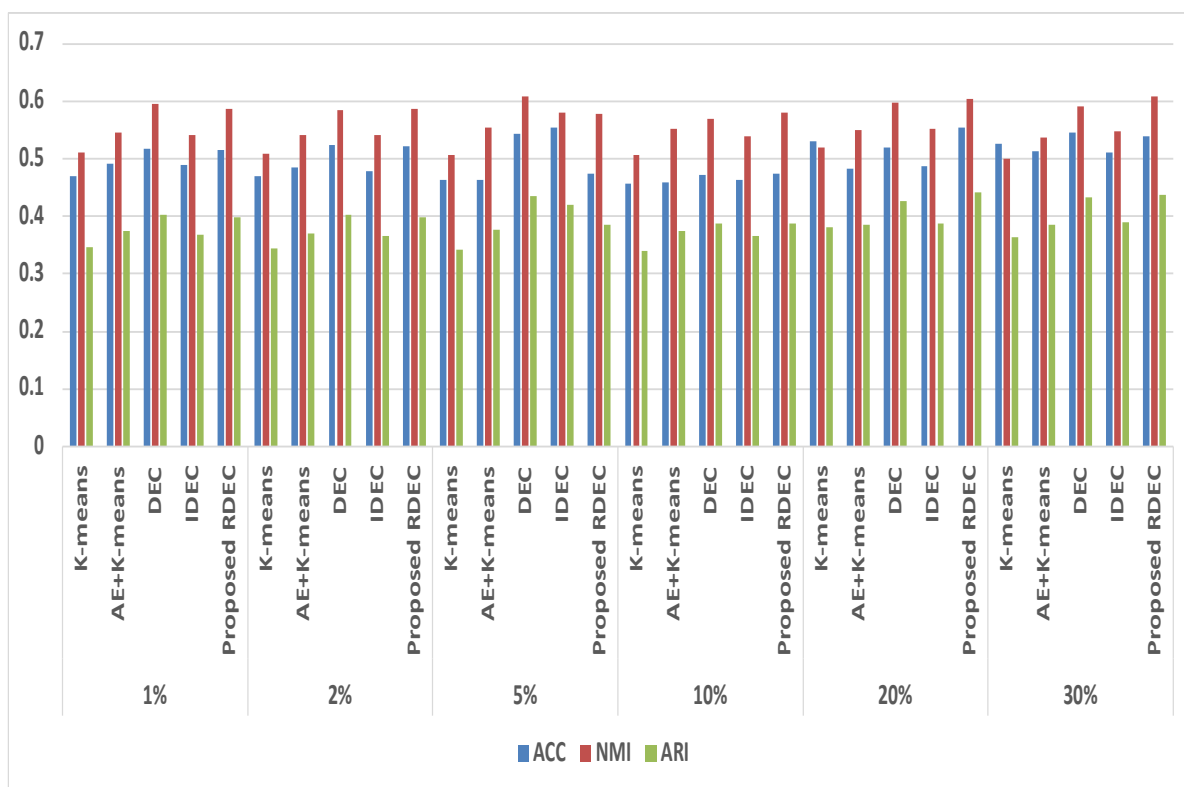


Figure 13. Clustering performance achieved using Fashion-MNIST under varying levels of semantic noise (1–30%).

A broader observation across all three benchmark datasets is that the clustering performance of every method gradually deteriorates as the corruption level increases. Under low-noise conditions, DEC generally provides the strongest performance because the latent representations remain only minimally affected by corrupted samples. However, as the proportion of noisy samples increases, this advantage becomes less consistent. In contrast, the proposed RDEC exhibits a more stable degradation pattern and becomes

increasingly competitive under moderate-to-high noise levels. This behavior is particularly evident for Fashion-MNIST and is also observed across several corruption levels on MNIST and USPS. These findings indicate that incorporating reliability-aware sample weighting helps reduce the influence of corrupted observations and improves the robustness of the clustering process under progressively more challenging conditions.

4.3. Clustering Performance Achieved Using Imbalanced Data

To study the effect of class imbalance, we create modified versions of the MNIST dataset using different retention rates. A minimum retention rate $r_{\min}$ is used to determine how underrepresented the smallest class is allowed to be. Sampling probabilities are assigned across classes, ranging from $r_{\min}$ for the least frequent class up to 1 for the most frequent one. In this way, smaller values of $r_{\min}$ lead to more severe imbalance in the data. The clustering performance under these settings is reported in Table 4. As it can be seen, as the retention rate increases, all methods benefit from a more balanced data distribution and yield an improved clustering accuracy. One should also note that under severe imbalance conditions (low $r_{\min}$), the performance of most baseline methods decreased noticeably. In contrast, the proposed RDEC maintained a more stable performance and yielded the highest accuracy at $r_{\min} = 0.1$, indicating stronger robustness to class imbalance. In addition, as the data distribution becomes more balanced, the performance gap between methods decreases gradually with IDEC and RDEC achieving competitive results at higher retention rates. Overall, the proposed RDEC remains highly competitive while providing improved stability under more challenging imbalanced scenarios.

Table 4. Clustering accuracy achieved using imbalanced MNIST under different retention rates ($r_{\min}$).

	$r_{\min} = 0.1$	$r_{\min} = 0.3$	$r_{\min} = 0.5$	$r_{\min} = 0.7$	$r_{\min} = 0.9$
K-means [3]	0.4587	0.4966	0.5185	0.5408	0.5388
AE + K-means	0.6534	0.7317	0.7451	0.7780	0.8004
IDEC [18]	0.6621	0.7423	0.7545	0.7882	0.8123
RDEC	0.7161	0.7553	0.7877	0.7697	0.8290

4.4. Number of Clusters Investigation

Since the true number of clusters is often unknown in practice, selecting a suitable value is essential for achieving reliable results. Accordingly, the clustering performance was assessed using MNIST [25] dataset for different numbers of clusters. In particular, we evaluated the proposed model using two standard performance measures. The first one is the Normalized Mutual Information (NMI), defined as

$$\text{NMI}(l, c) = \frac{I(l, c)}{\frac{1}{2}[H(l) + H(c)]}, \quad (9)$$

where I denotes the mutual information between predicted labels and ground truth, and H denotes entropy. The second metric is the generalizability, G , defined as the ratio between training and validation loss:

$$G = \frac{L_{\text{train}}}{L_{\text{validation}}} \quad (10)$$

One should note that a smaller value of G indicates better generalization, while larger values suggest potential overfitting. Figure 14 illustrates how NMI and generalizability measures vary with the number of clusters. As k increases, NMI improves at first and reaches its highest value around $k = 10$ – 11 , suggesting that the model is able to capture the underlying structure of the data at this point. When k becomes larger than this range, NMI gradually declines, which indicates that the data is being split into too many clusters. On the other hand, generalizability shows a steady upward trend as k increases. This

is expected, since a larger number of clusters gives the model more flexibility to fit the training data. However, this does not necessarily lead to better clustering performance.

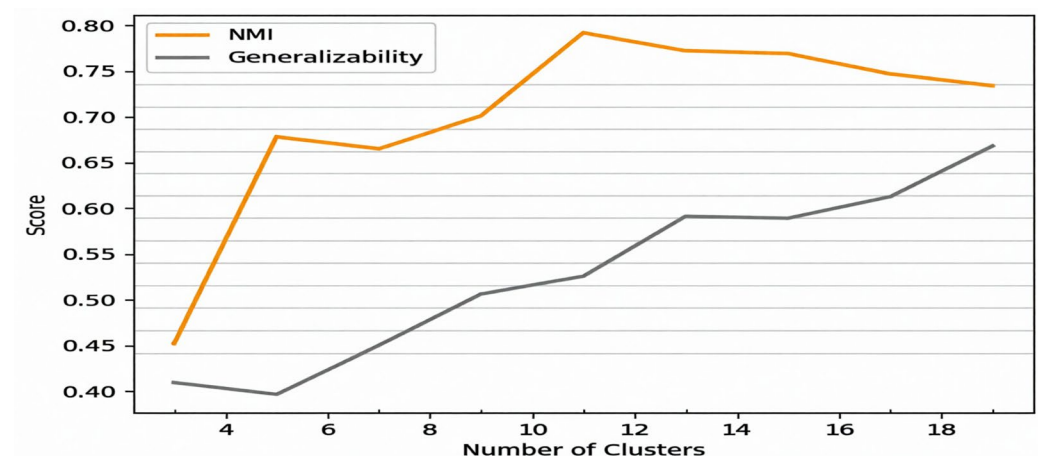


Figure 14. Normalized Mutual Information (NMI) and Generalizability (G) obtained using different number of centroids, k , for MNIST [25] dataset.

In summary, these observations suggest that the best choice of k lies near the peak of NMI, which is close to the true number of classes in the dataset. This also shows that NMI is a more reliable indicator for selecting the number of clusters, while generalizability helps provide additional insight into how the model behaves as k changes.

To further investigate the behavior of the proposed weighting mechanism under noisy conditions, we analyze the average latent-space distances and the corresponding average weights assigned to inlier and outlier samples. The difference between the average inlier and outlier weights is quantified using

$$\text{Weight Gap} = w_{in} - w_{out} \quad (11)$$

on the other hand, the relative separation between outlier and inlier samples in the latent space is measured as follows:

$$\text{Separation Ratio} = \frac{d_{out}}{d_{in}} \quad (12)$$

These measures provide additional insight into how effectively the proposed method distinguishes reliable samples from noisy or anomalous ones. The resulting statistics for the noisy MNIST [25] and USPS [26] and Fashion-MNIST [27] datasets are reported in Tables 5–7 respectively.

Table 5. Average distances and average weights for inlier and outlier samples from MNIST dataset.

	Inliers Distance	Outliers Distance	Inliers Weight	Outliers Weight	Weight Gap	Separation Ratio
1%	13.935	56.2604	0.8054	0.2067	0.5987	4.0373
2%	10.2602	51.1844	0.8204	0.1652	0.6553	4.9886
5%	6.3484	54.8967	0.8887	0.1008	0.7879	8.6474
10%	7.8626	97.342	0.9028	0.0966	0.8061	12.3804
20%	1.3761	11.9411	0.8721	0.2959	0.5762	8.6774
30%	2.7232	182.5608	0.9954	0.5294	0.4660	67.0392

The difference in weights between inliers and outliers (weight gap), together with the ratio of their distances (separation ratio), provides a useful indication of how well the

model separates reliable samples from noisy ones. Overall, inliers have smaller distances to cluster centers and thus higher weights, whereas outliers have larger distances and lower weights. This demonstrates that the proposed distance-aware weighting mechanism effectively prioritizes reliable training samples.

Table 6. Average distances and average weights for inlier and outlier samples from USPS dataset.

	Inliers Distance	Outliers Distance	Inliers Weight	Outliers Weight	Weight Gap	Separation Ratio
1%	6.9648	59.653	0.8264	0.0652	0.7611	8.5649
2%	4.0013	0.2134	0.9187	0.9965	−0.0778	0.0533
5%	4.9251	100.4759	0.9173	0.0868	0.8304	20.401
10%	2.4889	30.9099	0.8921	0.1225	0.7697	12.419
20%	0.4422	5.8208	0.9382	0.2033	0.7350	13.1633
30%	12.7976	0.4439	0.8791	0.9942	−0.1151	0.0347

Table 7. Average distances and average weights for inlier and outlier samples from the Fashion-MNIST dataset.

	Inliers Distance	Outliers Distance	Inliers Weight	Outliers Weight	Weight Gap	Separation Ratio
1%	10.2276	10.6337	0.744	0.556	0.188	1.0397
2%	9.4408	4.5771	0.7184	0.8104	−0.092	0.4848
5%	8.1181	45.3713	0.7779	0.0884	0.6895	5.5889
10%	4.9156	59.7306	0.8328	0.0679	0.7649	12.1513
20%	2.8117	181.0745	0.9904	0.0767	0.9137	64.4013
30%	2.1052	19.2059	0.9051	0.5004	0.4047	9.123

The statistics reported in Tables 5–7 provide additional insight into the behavior of the proposed reliability-aware weighting mechanism under noisy conditions across the MNIST, USPS, and Fashion-MNIST datasets. Overall, the results consistently show that inlier samples tend to remain closer to their corresponding cluster centers than outlier samples. Consequently, inliers receive higher reliability weights, whereas distant samples are assigned lower weights, reducing their influence during the clustering optimization process.

For the MNIST dataset (Table 5), this separation becomes particularly evident at moderate and high noise levels. For example, at a 10% noise ratio, the average inlier distance remains substantially smaller than the corresponding outlier distance, resulting in a large weight gap and a high separation ratio. Similar behavior is observed at the remaining noise levels, indicating that the proposed weighting mechanism effectively distinguishes reliable samples from noisy observations as the corruption level increases.

A similar trend is observed for the USPS dataset (Table 6), where positive weight gaps and large separation ratios are obtained for most noise levels, demonstrating effective discrimination between inlier and outlier samples. Nevertheless, at the 2% and 30% noise ratios, negative weight gaps are observed because the injected perturbations occasionally produce outlier samples that remain closer to the learned cluster centers than some inlier samples. As a result, these outliers receive relatively larger weights. However, these isolated cases do not noticeably affect the overall clustering performance, since the proposed RDEC continues to achieve competitive clustering results across the evaluated noise levels.

The Fashion-MNIST results presented in Table 7 further confirm the effectiveness of the proposed weighting strategy. Despite the greater complexity of this dataset, positive weight gaps are obtained for almost all corruption levels, while the separation ratio generally increases as the distance between inlier and outlier samples becomes more pronounced. The only exception occurs at the 2% noise level, where a small negative weight gap is observed

because a limited number of injected digit images remain close to the learned clothing clusters. Nevertheless, as the corruption level increases, the distance-aware weighting mechanism consistently assigns larger weights to representative clothing samples and lower weights to corrupted observations.

Overall, the consistent trends observed across Tables 5–7 demonstrate that the proposed distance-aware weighting mechanism effectively prioritizes reliable samples while reducing the contribution of noisy or atypical observations. These findings explain the improved robustness of RDEC under controlled image corruption scenarios and support the clustering results reported in the previous subsection.

To further illustrate this behavior, we present qualitative examples from the MNIST, USPS, and Fashion-MNIST datasets at a representative noise level (10%), corresponding to the setting where the proposed method achieves strong separation between inliers and outliers. As shown in Figure 15, these examples demonstrate how the proposed weighting mechanism effectively distinguishes reliable samples from noisy or ambiguous instances across different image datasets.

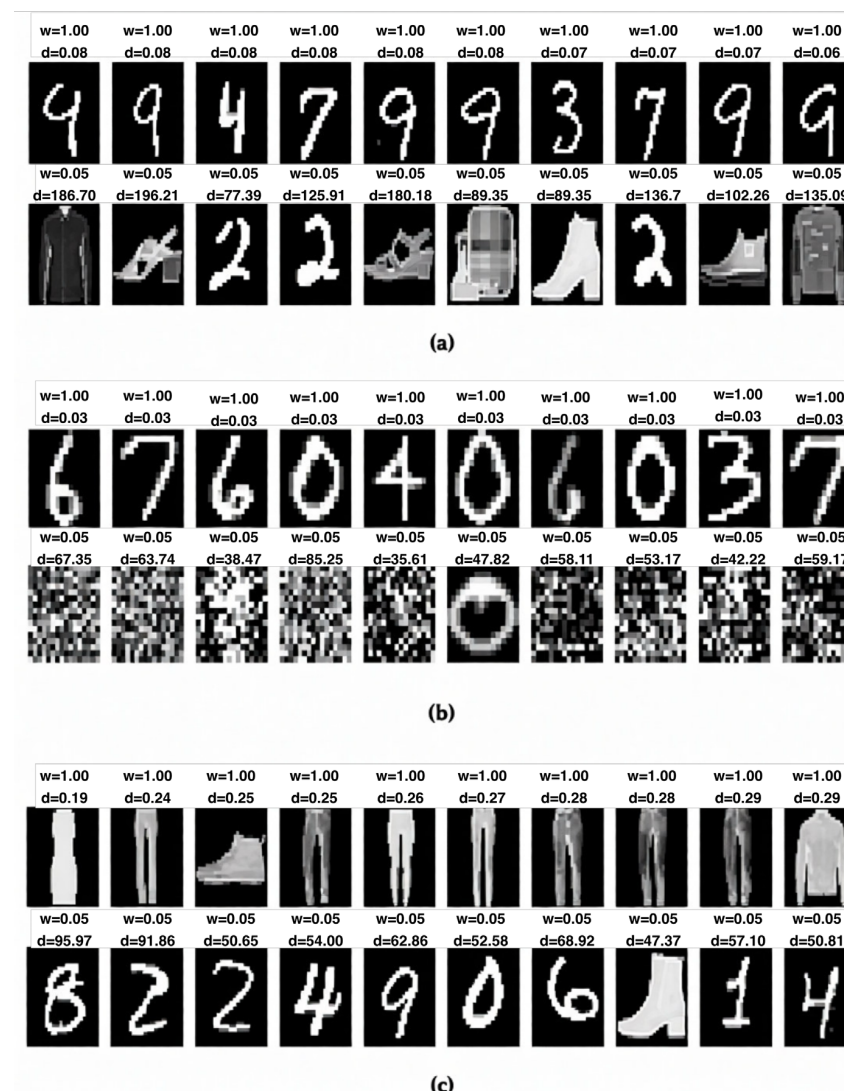


Figure 15. Illustrative samples from: (a) MNIST (10% noise). Top row shows the top-10 highest-weight samples, while the bottom row shows the top-10 lowest-weight samples. (b) USPS (10% noise). Top row shows the top-10 highest-weight samples, while the bottom row shows the top-10 lowest-weight samples. (c) Fashion-MNIST (10% noise). Top row shows the top-10 highest-weight samples, while the bottom row shows the top-10 lowest-weight samples.

Figure 15a shows examples from the MNIST [25] dataset that help illustrate how the weighting mechanism behaves in the presence of noise. The samples with the highest weights (top row) are generally clean and well-formed digits, and they tend to lie close to the corresponding cluster centers. By contrast, the lowest-weight samples (bottom row) include more ambiguous digits as well as semantic outliers introduced from Fashion-MNIST [27]. These examples are noticeably less consistent in structure and appear farther from the cluster centers, which explains why they receive lower weights. Overall, the model appears to place more emphasis on reliable samples while reducing the influence of noisier ones.

A similar pattern can be seen in Figure 15b for the USPS [26] dataset. The high-weight samples again correspond to clearer and more recognizable digits, whereas the low-weight samples include heavily corrupted inputs produced through synthetic perturbations. Compared to the MNIST case, these distortions are less structured and often harder to interpret visually. This makes the separation between reliable and unreliable samples more challenging, yet the model still assigns lower weights to the corrupted inputs. This behavior further supports the idea that the weighting mechanism is able to adapt to different types of noise.

Figure 15c presents the corresponding examples for the Fashion-MNIST [27] dataset. Similar to the previous datasets, the highest-weight samples (top row) correspond to representative clothing items that remain close to their corresponding cluster centers. In contrast, the lowest-weight samples (bottom row) include semantic outliers introduced from the MNIST dataset [25], together with a few more ambiguous clothing items, which generally appear farther from the learned cluster centers and consequently receive lower reliability weights. These qualitative examples further demonstrate that the proposed weighting mechanism effectively distinguishes representative inlier samples from semantically inconsistent observations across different image datasets.

Overall, the results indicate that the proposed RDEC improves robustness to noise while maintaining competitive performance on clean data. The distance-aware weighting strategy encourages the model to focus on more reliable samples while reducing the influence of noisy or semantically inconsistent ones, leading to more stable clustering behavior. At the same time, when the noise level becomes very high, the structure of the latent space becomes less distinct, which limits the effectiveness of the weighting mechanism.

4.5. Internal Cluster Validity Analysis

To further assess the quality of the learned latent representations, we complement the external evaluation based on ACC, NMI, and ARI with three widely used internal clustering validity metrics: the Silhouette Coefficient (SI), Calinski–Harabasz score (CH), and Davies–Bouldin Index (DBI). Unlike external metrics, these measures evaluate cluster compactness and separation directly from the learned latent space without relying on class labels. Table 8 reports the internal clustering performance of the proposed RDEC under clean conditions and a representative noisy setting (20% noise) across the three benchmark datasets. Together with the external evaluation, these metrics provide a more comprehensive assessment of the quality and robustness of the learned latent representations.

Table 8 shows that the proposed RDEC preserves high-quality latent representations under both clean and noisy conditions across the three benchmark datasets. For MNIST and USPS, the noisy setting (20%) yields higher SI and CH values together with lower DBI values than the corresponding clean setting, indicating that the learned latent representations remain compact and well separated despite the presence of noise.

For Fashion-MNIST, SI and CH also increase under the noisy setting, suggesting improved cluster compactness and overall latent-space organization. However, the DBI increases from 0.5027 to 0.8579, indicating that the separation between some neighbor-

ing clusters becomes less distinct. Nevertheless, the relatively high SI and CH values demonstrate that the latent representations remain well structured under noisy conditions.

Table 8. Internal cluster validity metrics of the proposed RDEC under clean and representative noisy conditions. Higher SI ($\uparrow$) and CH ($\uparrow$) indicate better clustering quality, whereas lower DBI ($\downarrow$) indicates better cluster separation.

Dataset	Noise Level	SI $\uparrow$	CH $\uparrow$	DBI $\downarrow$
MNIST	0%	0.7993	39,454.1836	0.4195
MNIST	20%	0.8788	308,691.2812	0.1616
USPS	0%	0.7905	3162.2183	0.5327
USPS	20%	0.9394	17,054.6875	0.4530
Fashion-MNIST	0%	0.6592	56,494.6758	0.5027
Fashion-MNIST	20%	0.8317	241,093.2812	0.8579

Overall, the internal validity analysis is consistent with the external evaluation presented in the previous subsection. Across all three datasets, RDEC maintains strong latent-space quality while achieving competitive ACC, NMI, and ARI scores, indicating that the proposed reliability-aware weighting mechanism effectively preserves the discriminative structure of the learned representations under noisy conditions.

5. Conclusions

This research introduced Reliability-Based Deep Embedded Clustering (RDEC), a deep clustering framework that incorporates sample reliability into the clustering process. The proposed method estimates the reliability of the data samples according to the distances between their latent embeddings and the nearest cluster centers. Moreover, these reliability estimates are integrated into a weighting mechanism that ensures that unreliable samples exert minimal influence on the optimization process, leading to more stable and accurate clustering performance. Moreover, a novel Kullback–Leibler divergence objective function is formulated to guide the clustering process and focus on the more representative data instances. In other words, typical data samples contribute more strongly during optimization, while the influence of distant and potentially unreliable samples is reduced. To better understand the effect of the proposed weighting mechanism, additional analyses of the latent-space distances, reliability weights, weight gaps, separation ratios, class-level clustering performance, and qualitative examples revealed a clear relationship between sample reliability and the assigned weights, further confirming the effectiveness of RDEC.

The experiments conducted using MNIST, USPS, and Fashion-MNIST datasets showed that RDEC achieves competitive clustering performance compared with the representative clustering baselines considered in this study while maintaining stable performance under controlled image corruption scenarios involving semantic outliers, synthetic perturbations, and class imbalance. Specifically, the results suggest that incorporating reliability information into the clustering objective can improve robustness without removing samples from the training process. In particular, the proposed approach achieved the best overall clustering performance on the clean USPS dataset and attained the highest NMI and ARI on the clean Fashion-MNIST dataset while maintaining competitive clustering accuracy. Moreover, the proposed weighting mechanism demonstrated consistent and robust behavior across multiple corruption levels on all three benchmark datasets. The present evaluation is limited to benchmark image datasets under controlled image corruption scenarios, and further validation on naturally occurring real-world noisy datasets remains an important direction for future work.

Future work will focus on several directions to further enhance the proposed framework. First, more sophisticated reliability estimation strategies will be investigated, includ-

ing adaptive and learnable weighting mechanisms that can dynamically adjust sample importance during training. Second, the proposed framework will be extended to other deep clustering architectures and evaluated on more diverse real-world datasets exhibiting varying levels of noise, class imbalance, and distribution complexity. Third, future studies will explore integrating reliability estimation with cluster-number determination methods to improve performance when the number of clusters is unknown. In addition, the applicability of the proposed approach will be examined in real-world domains such as medical imaging, biomedical data analysis, and other high-dimensional datasets. Finally, future work will investigate incorporating additional information beyond centroid distance, such as local neighborhood relationships, to provide a more comprehensive estimate of sample reliability, further improve clustering robustness, and explore how the proposed reliability-aware weighting mechanism can be extended to jointly support robust clustering and anomaly detection by explicitly characterizing atypical samples.

Author Contributions: Conceptualization, M.A., M.M.B.I. and O.B.; methodology, M.A., M.M.B.I. and O.B.; software, M.A.; validation, M.A., M.M.B.I. and O.B.; formal analysis, M.A.; investigation, M.A., M.M.B.I. and O.B.; resources, M.A., M.M.B.I. and O.B.; data curation, M.A.; writing—original draft preparation, M.A.; writing—review and editing, M.A., M.M.B.I. and O.B.; visualization, M.A.; supervision, M.M.B.I. and O.B.; project administration, M.M.B.I. and O.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets used in this study are publicly available. MNIST, USPS and Fashion-MNIST can be obtained from their respective public repositories.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ACC	Accuracy
AE	Autoencoder
ADEC	Adversarial Deep Embedded Clustering
ARI	Adjusted Rand Index
DCFOD	Deep Clustering based Fair Outlier Detection
DEC	Deep Embedded Clustering
DRDSC	Duet Robust Deep Subspace Clustering
EMA	Exponential Moving Average
GMM	Gaussian Mixture Model
IDEC	Improved Deep Embedded Clustering
KL	Kullback–Leibler Divergence
K-means	K-means Clustering
MNIST	Modified National Institute of Standards and Technology Database
NMI	Normalized Mutual Information
OTRIMLE	Optimally Tuned Robust Improper Maximum Likelihood Estimator
PCM	Possibilistic C-Means
RDEC	Reliability-based Deep Embedded Clustering
RIMLE	Robust Improper Maximum Likelihood Estimator
SDAE	Stacked Denoising Autoencoder
SGD	Stochastic Gradient Descent
USPS	United States Postal Service Handwritten Digit Dataset

References

1. Han, J.; Pei, J.; Tong, H. *Data Mining: Concepts and Techniques*; Morgan Kaufmann: Burlington, MA, USA, 2022.
2. Jain, A.K. Data clustering: 50 years beyond K-means. *Pattern Recognit. Lett.* **2010**, *31*, 651–666. [\[CrossRef\]](#)
3. MacQueen, J. Some methods for classification and analysis of multivariate observations. In *Proceedings of the 5-th Berkeley Symposium on Mathematical Statistics and Probability*; University of California Press: Oakland, CA, USA, 1967.
4. Xu, R.; Wunsch, D. Survey of clustering algorithms. *IEEE Trans. Neural Netw.* **2005**, *16*, 645–678. [\[CrossRef\]](#) [\[PubMed\]](#)
5. Hinton, G.E.; Salakhutdinov, R.R. Reducing the dimensionality of data with neural networks. *Science* **2006**, *313*, 504–507. [\[CrossRef\]](#) [\[PubMed\]](#)
6. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [\[CrossRef\]](#) [\[PubMed\]](#)
7. Bengio, Y.; Courville, A.; Vincent, P. Representation learning: A review and new perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 1798–1828. [\[CrossRef\]](#) [\[PubMed\]](#)
8. Baldi, P. Autoencoders, unsupervised learning, and deep architectures. In *Proceedings of the ICML Workshop on Unsupervised and Transfer Learning, Bellevue, WA, USA, 2 July 2011*; JMLR Workshop and Conference Proceedings; PMLR: Bellevue, WA, USA, 2011; pp. 37–49.
9. Ren, Y.; Pu, J.; Yang, Z.; Xu, J.; Li, G.; Pu, X.; Yu, P.S.; He, L. Deep clustering: A comprehensive survey. *IEEE Trans. Neural Netw. Learn. Syst.* **2024**, *36*, 5858–5878.
10. Xie, J.; Girshick, R.; Farhadi, A. Unsupervised deep embedding for clustering analysis. In *International Conference on Machine Learning*; PMLR: New York, NY, USA, 2016; pp. 478–487.
11. Mrabah, N.; Bouguessa, M.; Ksantini, R. Adversarial deep embedded clustering: On a better trade-off between feature randomness and feature drift. *IEEE Trans. Knowl. Data Eng.* **2020**, *34*, 1603–1617. [\[CrossRef\]](#)
12. Yang, X.; Deng, C.; Wei, K.; Yan, J.; Liu, W. Adversarial learning for robust deep clustering. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 9098–9108.
13. Forero, P.A.; Kekatos, V.; Giannakis, G.B. Robust clustering using outlier-sparsity regularization. *IEEE Trans. Signal Process.* **2012**, *60*, 4163–4177. [\[CrossRef\]](#)
14. Coretto, P.; Hennig, C. Robust improper maximum likelihood: Tuning, computation, and a comparison with other methods for robust Gaussian clustering. *J. Am. Stat. Assoc.* **2016**, *111*, 1648–1659. [\[CrossRef\]](#)
15. Jiang, Y.; Xu, Q.; Yang, Z.; Cao, X.; Huang, Q. Duet robust deep subspace clustering. In *Proceedings of the 27th ACM International Conference on Multimedia*, New York, NY, USA, 15 October 2019; pp. 1596–1604.
16. Johnson, S.C. Hierarchical clustering schemes. *Psychometrika* **1967**, *32*, 241–254. [\[CrossRef\]](#) [\[PubMed\]](#)
17. Reynolds, D.A. Gaussian mixture models. *Encycl. Biom.* **2009**, *741*, 3.
18. Guo, X.; Gao, L.; Liu, X.; Yin, J. Improved deep embedded clustering with local structure preservation. *Int. Jt. Conf. Artif. Intell.* **2017**, *17*, 1753–1759. [\[CrossRef\]](#) [\[PubMed\]](#)
19. Guo, X.; Liu, X.; Zhu, E.; Yin, J. Deep clustering with convolutional autoencoders. In *International Conference on Neural Information Processing*; Springer: Cham, Switzerland, 2017; pp. 373–382.
20. Ronen, M.; Finder, S.E.; Freifeld, O. Deepdpm: Deep clustering with an unknown number of clusters. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Virtual*, 19–25 June 2021; pp. 9861–9870.
21. Leiber, C.; Bauer, L.G.; Schelling, B.; Böhm, C.; Plant, C. Dip-based deep embedded clustering with k-estimation. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, Singapore, 14–18 August 2021; pp. 903–913.
22. Kondo, Y.; Salibian-Barrera, M.; Zamar, R. A robust and sparse K-means clustering algorithm. *arXiv* **2012**, arXiv:1201.6082.
23. Li, H.; Sugawara, S.; Katayama, S. Adaptively Robust and Sparse K-means Clustering. *arXiv* **2024**, arXiv:2407.06945.
24. Krishnapuram, R.; Keller, J.M. A possibilistic approach to clustering. *IEEE Trans. Fuzzy Syst.* **2002**, *1*, 98–110.
25. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE International Symposium on Information Theory*, Lausanne, Switzerland, 30 June–5 July 2002; Volume 86, pp. 2278–2324.
26. Hull, J.J. A database for handwritten text recognition research. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *16*, 550–554.
27. Xiao, H.; Rasul, K.; Vollgraf, R. Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms. *arXiv* **2017**, arXiv:1708.07747.
28. Vincent, P.; Larochelle, H.; Lajoie, I.; Bengio, Y.; Manzagol, P.-A.; Bottou, L. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *J. Mach. Learn. Res.* **2010**, *11*, 3371–3408.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.