IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 6 April 2026, accepted 18 May 2026, date of publication 29 May 2026, date of current version 16 June 2026.

Digital Object Identifier 10.1109/ACCESS.2026.3697984

RESEARCH ARTICLE

BIRCH-AE: A Hierarchical Ensemble Framework for Scalable E-Commerce User Segmentation With Autoencoder-Enhanced Feature Learning

CAIWEN LI[1], ISKANDAR ISHAK[1], HAMIDAH IBRAHIM[1], (Member, IEEE), MASLINA ZOLKEPLI[1], FATIMAH SIDI[1], (Member, IEEE), AND CAILI LI[2]

[1]Department of Computer Science, Faculty of Computer Science and Information Technology, Universiti Putra Malaysia (UPM), Serdang, Selangor 43400, Malaysia

[2]College of Art and Design, Heilongjiang Institute of Technology, Daowai, Harbin, Heilongjiang 150050, China

Corresponding author: Iskandar Ishak (iskandar_i@upm.edu.my)

This work was supported in part by the Journal Publication Fund under the University Driven Research Program (Digital and ICTI), Universiti Putra Malaysia, and in part by the Faculty of Computer Science and Information Technology, Universiti Putra Malaysia.

ABSTRACT The rapid expansion of e-commerce platforms has intensified demand for scalable, high-quality user segmentation systems capable of efficiently processing millions of behavioral records. This paper presents BIRCH-AE, a hierarchical ensemble clustering framework that integrates the Balanced Iterative Reducing and Clustering using Hierarchies (BIRCH) algorithm with autoencoder-based feature learning for large-scale e-commerce analytics. The autoencoder compresses high-dimensional behavioral data into compact latent representations, mitigating the curse of dimensionality and improving cluster separability. Multiple BIRCH configurations are combined through four ensemble strategies: Majority Voting, Weighted Voting, Advanced Affinity-based Spectral Clustering (AASC), and the proposed BIRCH-Optimized Hierarchical Consensus (BOHC). Dynamic selection based on multi-criteria evaluation automatically identifies the best-performing strategy per dataset setting, emphasizing that no single consensus method is universally optimal. Experiments on two large-scale datasets (Retail Rocket with 1.4M users and E-Commerce Behavior with 4.5M users) show improved clustering quality and practical scalability. BOHC achieves up to 23% silhouette improvement over single BIRCH on transaction-focused data, with a clearer hierarchical structure, while multi-domain data favors strong base models. Autoencoder feature learning improves clustering quality by 23–53% over raw features. The full 4.5M-user experiment was executed as a BOHC scalability run, completed in approximately 5 minutes, while framework-level comparative analyses were conducted via repeated stratified 30%-subset trials. These findings support BIRCH-AE as a practical and adaptive segmentation framework for enterprise-scale e-commerce analytics.

INDEX TERMS BIRCH clustering, hierarchical clustering, user segmentation, autoencoders, e-commerce analytics, ensemble methods, scalable clustering, incremental learning.

I. INTRODUCTION

The exponential growth of e-commerce platforms has fundamentally transformed retail dynamics, generating unprecedented volumes of user interaction data [1], [2]. Modern e-commerce systems routinely collect detailed behavioral information spanning browsing patterns, purchase histories, search queries, and engagement metrics at multiple touchpoints [3]. This rich data landscape presents opportunities and challenges for companies seeking to understand customer behavior and deliver personalized experiences.

User segmentation, the systematic process of dividing a heterogeneous customer base into homogeneous groups that share similar characteristics or behaviors, has become a cornerstone strategy in modern data-driven systems [4], [5]. The fundamental premise of segmentation is that customers exhibit diverse needs, preferences, purchasing patterns, and engagement behaviors. Rather than applying uniform

The associate editor coordinating the review of this manuscript and approving it for publication was Dominik Strzalka.

© 2026 The Authors. This work is licensed under a Creative Commons Attribution 4.0 License.
For more information, see https://creativecommons.org/licenses/by/4.0/

strategies across an entire customer population, segmentation enables organizations to identify distinct subgroups and tailor their approaches accordingly [6]. This strategic differentiation has profound implications across multiple business functions, from marketing and product development to customer service and inventory management.

In contemporary e-commerce environments, effective user segmentation serves several critical business objectives [7], [8]. First, it enables *personalized marketing campaigns* that resonate with specific customer segments, significantly improving conversion rates and return on marketing investment. By understanding the characteristics and preferences of different user groups, businesses can craft targeted messages, promotional offers, and product recommendations that align with each segment's unique profile [3]. Second, segmentation facilitates *dynamic pricing strategies* and *inventory optimization*, allowing platforms to adjust pricing and stock levels based on segment-specific demand patterns and price sensitivities [1]. Third, it supports *customer lifetime value prediction* and *churn prevention*, enabling proactive retention strategies for high-value segments and win-back campaigns for at-risk customers [9]. Fourth, segmentation provides *strategic insights for product development* by revealing unmet needs and emerging trends within specific customer groups [4].

The significance of user segmentation has increased as the field has evolved from traditional demographic and rule-based categorization to sophisticated data-driven behavioral segmentation [7]. Modern approaches leverage machine learning and data mining techniques to uncover complex patterns in vast behavioral datasets, enabling more nuanced and actionable customer understanding [10]. E-commerce platforms now analyze multidimensional behavioral signals, including browsing frequency, session duration, product category exploration, cart abandonment patterns, purchase recency and frequency, average order values, price sensitivity indicators, temporal activity patterns, and cross-category engagement metrics [2], [11]. This comprehensive behavioral view enables segmentation strategies that go far beyond simple RFM (Recency, Frequency, Monetary) analysis, capturing the full complexity of modern customer journeys [9].

However, the transition to large-scale, behavior-driven segmentation introduces significant technical challenges. The enormous data generated by modern e-commerce platforms, with millions of users creating billions of interaction events, requires clustering algorithms that can efficiently process large datasets [12], [13]. The high dimensionality of behavioral feature spaces, often encompassing hundreds of correlated variables, poses significant challenges for clustering algorithms. These include the curse of dimensionality, degradation of distance metrics, and increased noise sensitivity [14], [15]. The dynamic nature of e-commerce environments, where user behavior continuously evolves, and new customers arrive in real time, requires segmentation systems that can adapt incrementally without complete recomputation [16]. Furthermore, the requirement for multi-resolution segmentation, supporting both strategic planning with broad customer categories and tactical marketing with fine-grained micro-segments, adds additional complexity that traditional clustering methods struggle to address [17].

These challenges motivate the development of advanced clustering frameworks that can simultaneously achieve scalability, quality, and adaptability for enterprise-scale e-commerce user segmentation. The need for such systems has become particularly acute as platforms expand their customer bases to millions and diversify their product offerings across multiple categories, creating increasingly complex segmentation requirements.

A. LIMITATIONS OF TRADITIONAL APPROACHES

Traditional clustering algorithms, including K-Means [18], DBSCAN [19], and hierarchical agglomerative methods [20], have been extensively applied to customer segmentation tasks across various domains. However, recent comprehensive surveys and empirical studies [7], [8], [12] identify three critical limitations that severely constrain their applicability when confronting contemporary e-commerce data at scale.

First, scalability constraints significantly limit practical deployability for large-scale applications. The computational complexity of traditional clustering algorithms becomes prohibitive as dataset sizes grow into the millions of users common in modern e-commerce platforms. K-Means, despite its popularity and relative efficiency, requires multiple iterative passes over the entire dataset to converge. Each iteration computes distances between all data points and cluster centroids [18], [21]. For datasets with millions of users and dozens of features, this iterative process can take hours or even days to complete, making real-time or near-real-time segmentation infeasible [12]. Traditional hierarchical agglomerative methods face even more severe limitations, typically exhibiting quadratic time complexity $O(n^2)$ and requiring $O(n^2)$ memory to store the distance matrix between all pairs of data points [17], [20]. These quadratic requirements render hierarchical methods completely impractical for datasets exceeding a few tens of thousands of users—far below the scale of contemporary e-commerce platforms with millions of active customers [13]. DBSCAN, while offering advantages for discovering arbitrarily shaped clusters, similarly struggles with scalability due to its need to compute neighborhood relationships, often requiring spatial indexing structures that themselves become computationally expensive for high-dimensional data [19]. The scalability limitation is not merely a matter of computational cost. Still, it fundamentally limits organizations' ability to perform comprehensive segmentation across their full customer bases, often forcing them to work with small samples that may not capture the full diversity of customer behaviors [12].

Second, the curse of dimensionality fundamentally degrades clustering performance in high-dimensional behavioral feature spaces. Modern e-commerce platforms generate rich behavioral profiles encompassing hundreds of features: activity metrics (total events, unique products

viewed, sessions count), engagement patterns (events per session, session duration, category diversity), transactional features (purchase frequency, average order value, price range statistics), temporal behaviors (preferred shopping times, activity consistency, recency indicators), and cross-category interaction patterns [2], [11]. These comprehensive feature sets are essential for capturing the multidimensional nature of customer behavior but create significant challenges for traditional clustering algorithms. In high-dimensional spaces, distance metrics become less discriminative as the relative contrast between nearest and farthest neighbors diminishes: a phenomenon extensively documented in the literature [13], [14]. This distance concentration effect causes K-Means and other distance-based algorithms to produce suboptimal clusters, since the fundamental assumption that nearby points should belong to the same cluster breaks down when distances lose meaning [10]. Furthermore, behavioral features of e-commerce exhibit strong intercorrelations: browsing frequency correlates with session counts, purchase amounts correlate with price sensitivity, and category exploration correlates with session duration [15]. These correlations introduce redundancy that inflates the effective dimensionality without adding proportional information, exacerbating computational costs and further degrading the quality of the distance metric [14]. Traditional dimensionality reduction techniques, such as Principal Component Analysis (PCA), provide only linear projections that may fail to capture the complex nonlinear relationships inherent in behavioral data [22]. In contrast, more sophisticated non-linear methods have not been systematically integrated with scalable clustering algorithms for e-commerce applications [23], [24].

Third, the static nature of traditional clustering methods renders them unsuitable for dynamic e-commerce environments requiring continuous adaptation, and commerce platforms operate in highly dynamic contexts where user behaviors evolve continuously. Customers move through lifecycle stages, respond to seasonal trends, adopt new shopping patterns, and alter their preferences for categories over time [2], [3]. Additionally, platforms continuously acquire new users that must be integrated into the segmentation scheme without disrupting existing segment definitions [16]. Traditional clustering algorithms are fundamentally batch-oriented: they require access to the complete dataset at training time and produce a fixed partition that cannot adapt to new data without complete recomputation [13]. When new users arrive or existing users exhibit changes in behavior, the only option is to rerun the entire clustering process from scratch: an approach that is computationally prohibitive for million-user platforms that require daily or even hourly segment updates [12], [16]. This limitation creates a critical gap between the business requirements for responsive near-real-time segmentation and the technical capabilities of traditional methods. K-Means lacks an incremental learning mechanism; each new batch of users requires reinitialization of cluster centroids and reprocessing all data [18]. Hierarchical methods similarly require rebuilding the entire dendrogram from scratch, with no mechanism to insert new data points into an existing hierarchy [17]. Although some research has explored incremental variants of these algorithms [16], these approaches often sacrifice cluster quality for adaptability or introduce additional hyperparameters that are difficult to tune in practice. The absence of robust incremental learning capabilities forces e-commerce platforms to choose between maintaining outdated segmentations that no longer reflect current customer behaviors or investing substantial computational resources in frequent, complete re-clustering. Neither option is satisfactory for operational excellence.

These three fundamental limitations: scalability constraints, high-dimensional feature degradation, and lack of incremental adaptability—collectively demonstrate that traditional clustering approaches, despite their theoretical elegance and historical success in smaller-scale applications, are inadequate for modern enterprise-scale e-commerce user segmentation. This recognition motivates the development of novel frameworks that simultaneously address all three challenges through architectural innovations that combine scalable hierarchical methods, advanced feature learning, and incremental update mechanisms.

B. RESEARCH GAPS AND MOTIVATION

Despite extensive research in clustering methodologies, recent comprehensive surveys [25], [26], [27] identify critical gaps in large-scale e-commerce user segmentation. Although BIRCH [28] offers excellent scalability through its Clustering Feature (CF) Tree structure and single-pass processing, its application to e-commerce remains limited. Existing work focuses primarily on spatial data and sensor networks [29]. Systematic approaches to optimize BIRCH performance through ensemble methods and advanced feature learning remain underdeveloped for behavioral data.

Current ensemble clustering frameworks [26], [30], [31] predominantly focus on partition-based algorithms, neglecting the unique advantages of hierarchical methods in capturing multi-scale user groupings. Modern deep clustering research [23], [24] primarily integrates autoencoders with K-Means or Gaussian Mixture Models, overlooking hierarchical algorithms despite their superior ability to reveal nested customer segment structures, which are crucial for marketing strategies [27].

Furthermore, measuring user engagement through navigational behavior patterns [11] has become increasingly important. These engagement metrics, when combined with transactional data, provide a comprehensive view of user behavior that significantly enhances segmentation quality. The integration of autoencoder-based feature learning with BIRCH's hierarchical structure (particularly for high-dimensional e-commerce data with correlated variables [8], [15]) represents an unexplored opportunity.

C. CONTRIBUTIONS

To address these limitations and research gaps, this paper introduces BIRCH-AE, a comprehensive framework for

scalable e-commerce user segmentation that integrates BIRCH's hierarchical clustering capabilities with deep autoencoder-based feature learning and advanced ensemble consensus mechanisms. The framework's architecture positions BIRCH as the core scalable clustering engine, enhanced by autoencoder-driven dimensionality reduction and unified through multiple ensemble strategies, including the novel BIRCH-Optimized Hierarchical Consensus (BOHC) method. The key contributions are as follows.

1) **Scalable Hierarchical Framework with Deep Feature Integration:** A unified architecture combining BIRCH's memory-efficient hierarchical clustering with deep autoencoder-based feature learning. The autoencoder compresses high-dimensional, correlated behavioral data (30–50 features) into compact latent representations (14–32 dimensions) while preserving 90–95% of variance, allowing efficient processing of million-user datasets. This integration achieves 23–53% improvements in clustering quality over raw features while maintaining BIRCH's scalable incremental learning behavior.
2) **Novel Hierarchical Ensemble Method (BOHC):** Introduction of BIRCH-Optimized Hierarchical Consensus, a specialized ensemble strategy designed to preserve multi-scale clustering information inherent in BIRCH's CF Tree structure. Unlike conventional voting or spectral ensemble methods, BOHC constructs hierarchical affinity matrices based on ancestral relationships within CF Trees, enabling consensus formation that preserves hierarchical context across multiple BIRCH configurations. Experiments demonstrate BOHC achieves up to 23% improvement over single BIRCH models for transaction-focused datasets with distinct cluster boundaries.
3) **Comprehensive Ensemble Strategy Suite with Dynamic Selection:** Integration of four complementary ensemble approaches: Majority Voting, Weighted Voting, Advanced Affinity-based Spectral Clustering (AASC), and BOHC within the BIRCH-AE framework. A multi-criteria dynamic selection mechanism automatically identifies optimal strategies based on Silhouette Score, Calinski-Harabasz Index, and Davies-Bouldin Index, adapting to dataset characteristics without manual intervention. This adaptive approach addresses the challenge that no single ensemble method universally outperforms across different data distributions and domain granularities.
4) **Domain Granularity Impact Discovery:** Empirical evidence revealing that domain granularity (single-domain vs. multi-domain) fundamentally determines optimal segmentation strategies. Single-domain datasets (e.g., transaction-focused platforms, individual product categories) consistently benefit from ensemble methods (+17–23% improvement), while multi-domain datasets (multiple product categories) favor well-tuned base algorithms (+7.4% over ensembles). This finding, validated across three distinct evaluation scenarios, provides practitioners with clear decision criteria for selecting methods based on platform structure.
5) **Enterprise-Scale Validation and Deployment Guidelines:** Comprehensive empirical validation on representative subsets of two large-scale datasets (Retail Rocket: 1.4M users, E-Commerce Behavior: 4.5M users) with single-domain category analysis, supported by 20 randomized stratified subset trials. A production-scale BOHC run on 4.5M users (approximately 5 minutes) demonstrates operational feasibility at scale. Evidence-based deployment recommendations guide practitioners through data profiling, method selection based on domain granularity and cluster characteristics, and incremental update strategies for dynamic environments.

These contributions collectively advance large-scale user segmentation by providing an empirically grounded, practically deployable framework that addresses the fundamental limitations of traditional clustering approaches and offers actionable guidance for real-world e-commerce applications.

D. PAPER ORGANIZATION

The remainder of this paper is organized as follows. Section II reviews related work in clustering methods, ensemble techniques, and dimensionality reduction for user segmentation. Section III provides technical background on BIRCH clustering and autoencoders. Section IV details the BIRCH-AE framework architecture, including the four ensemble strategies (Majority Voting, Weighted Voting, AASC, and BOHC) and the dynamic selection mechanism. Section V presents a comprehensive experimental evaluation with detailed comparative analysis. Section VI discusses implications, limitations, and applications. Section VII concludes with future research directions. To support reproducibility, the BIRCH-AE implementation code and experimental materials are available [32].

II. RELATED WORK

A. E-COMMERCE USER SEGMENTATION

User segmentation has been an integral part of e-commerce analytics, with methodologies evolving from demographic and rule-based categorization to sophisticated machine learning approaches [4], [5]. Modern comprehensive reviews [7], [8] highlight this evolution towards data-driven behavioral segmentation based on purchase patterns, browsing behavior, and engagement metrics [10].

Garcia et al. [2] demonstrate the effectiveness of deep learning in customer behavior analysis. In contrast, Patel et al. [3] emphasize the need for real-time segmentation to support dynamic personalization at scale. Traditional clustering algorithms remain popular for their simplicity but face significant challenges in handling the volume of contemporary e-commerce data.

K-Means [18] suffers from sensitivity to initialization and assumptions of spherical clusters [21]. Zhao et al. [15] address high-dimensional customer segmentation with correlated features using extended regularized K-Means, demonstrating the importance of accounting for feature correlations. DBSCAN [19] offers advantages for arbitrarily shaped clusters but requires careful parameter tuning and struggles with varying density patterns. Traditional hierarchical methods provide intuitive dendrograms but are computationally expensive for large-scale applications [17].

Recent advanced techniques include model-based clustering [33], [34], spectral methods [35], and hybrid approaches. Gaussian Mixture Models [36] provide probabilistic cluster assignments, but require careful model selection. Spectral clustering excels at non-convex clustering using graph-based representations but faces scalability challenges [12].

Lim et al. [11] introduced the Cluster-N-Engage framework, which measures user engagement through navigational behavior patterns and demonstrates the importance of integrating engagement metrics with traditional clustering for comprehensive customer segmentation.

B. BIRCH: SCALABLE HIERARCHICAL CLUSTERING

BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies) was introduced by Zhang et al. [28] as a memory-efficient clustering method for very large databases. The algorithm constructs a Clustering Feature Tree (CF Tree). This hierarchical data structure provides compact dataset summaries via triplet representations containing the number of points, their linear sum, and the sum of squared values.

BIRCH operates in four phases: (1) scanning the database to build an initial CF tree in-memory, (2) optional tree condensation to reduce size, (3) global clustering of leaf entries, and (4) optional cluster refinement. Key parameters include the branching factor (maximum number of children per non-leaf node), the threshold (maximum diameter of subclusters), and the number of final clusters. BIRCH's incremental nature enables efficient handling of streaming data, making it particularly suitable for dynamic e-commerce environments [29].

Despite scalability advantages, BIRCH has received limited attention in e-commerce segmentation research, with most applications focusing on spatial data and sensor networks [37]. This work addresses this gap by demonstrating BIRCH's effectiveness for e-commerce user segmentation and extending its capabilities through ensemble methods and deep feature learning.

C. ENSEMBLE CLUSTERING METHODS

The ensemble clustering combines multiple base clustering solutions to produce consensus results that are more robust and accurate than individual clusterings [30], [31]. The fundamental principle is that different algorithms or parameter configurations capture distinct aspects of the data structure, and that their combination can achieve superior performance. Various consensus functions have been proposed, including co-association matrices, graph-based methods, and voting mechanisms [38].

Liu et al. [27] provide a comprehensive review of the past decade, highlighting that partition-based ensemble methods dominate the literature. Li et al. [26] emphasize the growing importance of ensemble learning for big data clustering, noting that most approaches focus on K-Means variants, while hierarchical ensemble methods remain underexplored.

Strehl and Ghosh [30] proposed three consensus functions: the Cluster-based Similarity Partitioning Algorithm (CSPA), the HyperGraph Partitioning Algorithm (HGPA), and the Meta-CLustering Algorithm (MCLA). Fred and Jain [31] introduced the evidence accumulation clustering framework using co-association matrices. Recent work explores advanced consensus strategies, including spectral methods [39] and non-negative matrix factorization [40].

However, ensemble methods specifically designed for hierarchical clustering algorithms, such as BIRCH, remain underexplored despite their potential advantages. Hierarchical methods yield richer structural information than partition-based algorithms, enabling novel ensemble strategies that leverage hierarchical relationships [34]. This work introduces ensemble techniques tailored to BIRCH's hierarchical nature, including BOHC, which preserves multi-scale clustering information.

D. DEEP LEARNING FOR DIMENSIONALITY REDUCTION

High-dimensional feature spaces pose significant challenges for clustering algorithms, including increased computational complexity, degraded distance metrics, and greater noise sensitivity [14]. Traditional dimensionality reduction techniques such as Principal Component Analysis (PCA) [22], Linear Discriminant Analysis [41], and t-SNE [42] have been widely applied. PCA provides efficient linear projection but may fail to capture nonlinear patterns. t-SNE excels at visualization but is computationally intensive and best suited to low-dimensional embeddings.

Zhao et al. [15] specifically address correlated variables in high-dimensional customer segmentation through regularized K-Means, demonstrating the importance of accounting for feature interdependencies in e-commerce datasets.

Autoencoders have emerged as powerful nonlinear dimensionality-reduction tools capable of learning compact representations that preserve essential data characteristics [43]. Recent comprehensive surveys [23], [24] systematically review advances in autoencoder-based deep clustering, highlighting superior performance over traditional methods for high-dimensional data. The encoder transforms high-dimensional input into a low-dimensional latent representation, while the decoder reconstructs the original input; training via reconstruction minimization encourages the latent space to capture salient features.

Recent research has extensively explored the combination of autoencoders with clustering algorithms, focusing

primarily on deep embedded clustering (DEC) [44] and related methods that jointly optimize clustering and representation learning. However, comprehensive reviews [23], [25] reveal that these approaches typically employ K-Means as the clustering component and have not leveraged the scalability advantages of hierarchical methods. This work addresses this gap by integrating autoencoder-based feature learning with BIRCH clustering, combining the representational power of autoencoders with the scalability and incremental learning capabilities of BIRCH [24].

E. RESEARCH POSITIONING

BIRCH-AE occupies a unique position at the intersection of hierarchical clustering, ensemble methods, and deep feature learning for e-commerce analytics. Unlike existing approaches focusing on partition-based ensemble clustering or simple BIRCH applications, this framework: (1) leverages BIRCH's scalability for large-scale e-commerce data, (2) enhances clustering quality through autoencoder-based feature learning that handles correlated variables, (3) introduces four ensemble strategies (including the novel BOHC method) specifically designed for hierarchical clustering, (4) provides dynamic ensemble selection for adaptive optimization, and (5) incorporates engagement metrics alongside traditional behavioral features. This integrated approach addresses the critical limitations of existing methods and offers practical solutions for enterprise-scale e-commerce user segmentation.

III. TECHNICAL PRELIMINARIES

A. BIRCH CLUSTERING FUNDAMENTALS

1) CLUSTERING FEATURE DEFINITION

The foundation of BIRCH is the Clustering Feature (CF), a compact statistical summary of a cluster. For a cluster containing Nd-dimensional data points $\{X_i\}_{i=1}^{N}$, the CF is defined as a triplet:

$$CF = (N, \vec{LS}, SS) \tag{1}$$

where N is the number of data points, $\vec{LS} = \sum_{i=1}^{N} X_i$ is the linear sum of the data points, and $SS = \sum_{i=1}^{N} X_i^2$ is the sum of the squared data points.

The CF representation is additive: for two disjoint clusters with CFs $(N_1, \vec{LS}_1, SS_1)$ and $(N_2, \vec{LS}_2, SS_2)$, their merged CF is:

$$CF_{merged} = (N_1 + N_2, \vec{LS}_1 + \vec{LS}_2, SS_1 + SS_2) \tag{2}$$

This additivity property enables efficient hierarchical construction and dynamic cluster updates.

2) CF TREE STRUCTURE

The CF Tree is a height-balanced tree with both leaf and non-leaf nodes. The nodes without leaves contain entries $[CF_i, child_i]$, where CF_i summarizes all data points in the root subtree at $child_i$. Leaf nodes contain entries for individual subclusters and are linked in sequence to facilitate scanning. The tree is controlled by three parameters: Branching Factor (B), maximum children for non-leaf nodes; Threshold (T), maximum diameter for subclusters at leaf nodes; and Number of Leaf Entries (L), maximum entries per leaf node.

3) BIRCH ALGORITHM PHASES

BIRCH operates in four phases:

Phase 1 (Tree Construction): The algorithm scans the dataset once, inserting each data point into the CF Tree. For each point, it traverses from root to leaf, selecting the closest child based on a distance metric. If the diameter of the leaf entry after insertion remains below the threshold T, the point is absorbed; otherwise, a new leaf entry is created.

Phase 2 (Tree Condensation - Optional): If memory constraints are violated, a larger threshold T' is selected, and the tree is rebuilt to produce a more compact structure.

Phase 3 (Global Clustering): A global clustering algorithm is applied to the leaf entries, treating each leaf CF as a pseudo-data point. This phase can employ K-Means, hierarchical agglomeration, or other algorithms.

Phase 4 (Cluster Refinement - Optional): Data points are redistributed to their nearest cluster centroids to correct potential errors from earlier phases.

4) DISTANCE METRICS

BIRCH employs specific distance metrics optimized for CF representations. For two clusters with CFs $CF_1 = (N_1, \vec{LS}_1, SS_1)$ and $CF_2 = (N_2, \vec{LS}_2, SS_2)$, common metrics include:

Centroid Euclidean Distance:

$$D_0 = \left\| \frac{\vec{LS}_1}{N_1} - \frac{\vec{LS}_2}{N_2} \right\| \tag{3}$$

Average Inter-cluster Distance:

$$D_2 = \sqrt{\frac{1}{N_1 N_2} \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \|X_i - Y_j\|^2} \tag{4}$$

Average Intra-cluster Distance:

$$D_3 = \sqrt{\frac{2N_1 SS_1 - 2\|\vec{LS}_1\|^2}{N_1(N_1 - 1)} + \frac{2N_2 SS_2 - 2\|\vec{LS}_2\|^2}{N_2(N_2 - 1)}} \tag{5}$$

B. AUTOENCODER ARCHITECTURE

1) BASIC FORMULATION

An autoencoder is a neural network designed to learn efficient data representations in an unsupervised manner, consisting of two components:

Encoder: Maps input $x \in \mathbb{R}^d$ to latent representation $z \in \mathbb{R}^p$ where $p < d$:

$$z = f_{\theta_e}(x) = \sigma(W_e x + b_e) \tag{6}$$

Decoder: Reconstructs the input from latent representation:

$$\hat{x} = g_{\theta_d}(z) = \sigma(W_d z + b_d) \tag{7}$$

where $\theta_e = \{W_e, b_e\}$ and $\theta_d = \{W_d, b_d\}$ are encoder and decoder parameters, and σ is an activation function.

2) TRAINING OBJECTIVE

The autoencoder is trained to minimize the reconstruction error:

$$\mathcal{L}(\theta_e, \theta_d) = \frac{1}{n}\sum_{i=1}^{n} \|x_i - \hat{x}_i\|^2 \tag{8}$$

For e-commerce applications, additional regularization terms are incorporated:

$$\mathcal{L}_{total} = \mathcal{L}_{reconstruction} + \lambda_1 \mathcal{L}_{sparsity} + \lambda_2 \mathcal{L}_{regularization} \tag{9}$$

where sparsity encourages selective feature activation, and regularization prevents overfitting.

3) DEEP ARCHITECTURE

For complex e-commerce behavioral data, a multi-layer architecture is employed:

Encoder layers:

$$h_1 = \sigma_1(W_1 x + b_1) \tag{10}$$
$$h_2 = \sigma_2(W_2 h_1 + b_2) \tag{11}$$
$$\vdots$$
$$z = \sigma_L(W_L h_{L-1} + b_L) \tag{12}$$

Decoder layers: Mirror the encoder with tied or untied weights, reconstructing the original input through successive transformations.

C. CLUSTERING EVALUATION METRICS

1) SILHOUETTE SCORE

The silhouette score [45] measures how similar an object is to its own cluster compared to other clusters. For a data point i:

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \tag{13}$$

where $a(i)$ is the average distance to points in the same cluster and $b(i)$ is the minimum average distance to points in other clusters. The overall silhouette score is

$$S = \frac{1}{n}\sum_{i=1}^{n} s(i) \tag{14}$$

with values ranging from -1 to 1, where higher values indicate better-defined clusters.

2) CALINSKI-HARABASZ INDEX

The Calinski-Harabasz (CH) index [46] evaluates cluster quality based on the ratio of between-cluster to within-cluster variance:

$$CH = \frac{SS_B/(k-1)}{SS_W/(n-k)} \tag{15}$$

where SS_B is the sum of squares between groups, SS_W is the sum of squares within groups, k is the number of groups, and n is the number of data points. Higher values indicate better clustering.

3) DAVIES-BOULDIN INDEX

The Davies-Bouldin (DB) index [47] measures the average similarity between each group and its most similar group:

$$DB = \frac{1}{k}\sum_{i=1}^{k} \max_{j \neq i} \left(\frac{\sigma_i + \sigma_j}{d(c_i, c_j)}\right) \tag{16}$$

where σ_i is the average distance between the points in the cluster i to the centroid of the cluster, and $d(c_i, c_j)$ is the distance between the centroids. Lower values indicate better clustering.

IV. BIRCH-AE FRAMEWORK METHODOLOGY

A. COMPONENT SELECTION RATIONALE

Our architectural choices—standard autoencoder for feature learning and BIRCH for hierarchical clustering—were guided by requirements for the production system and preliminary empirical evaluation. For feature learning, we selected standard autoencoders over Variational Autoencoders (VAE) for three reasons. First, recent surveys [23] note that while VAEs offer probabilistic guarantees, reconstruction-focused autoencoders often achieve superior clustering performance when latent space separation is prioritized over generative capability. Second, preliminary experiments on representative subsets confirmed that standard AE achieved higher silhouette scores (0.839 vs. VAE's 0.636) for e-commerce behavioral data. Third, VAE's KL-divergence regularization toward Gaussian priors introduces stochasticity that conflicts with deterministic e-commerce patterns (such as consistent purchase sequences and browsing habits). We also evaluated UMAP (silhouette score: 0.527) and PCA (silhouette score: 0.531), both of which yielded inferior results.

For clustering, BIRCH was selected for its hierarchical structure, its scalability compared with classical agglomerative approaches [28], and its incremental learning capability [29]. Preliminary evaluation on $k \in \{5, 10, 15, 20, 25\}$ showed that BIRCH maintained comparatively stable performance (silhouette $0.839 \rightarrow 0.721$, -14% degradation), while K-Means degraded more strongly as k increased. This stability is important for e-commerce settings that require finer-grained segmentation ($k = 15$–25).

B. FRAMEWORK OVERVIEW

The BIRCH-AE framework integrates four key components into a cohesive pipeline: (1) data preprocessing and feature engineering, (2) autoencoder-based dimensionality reduction, (3) BIRCH ensemble clustering with multiple parameter configurations, and (4) dynamic ensemble selection and consensus generation. The framework is designed with scalability as a primary consideration, leveraging BIRCH's memory-efficient structure and incremental learning
capabilities.

The framework is modular rather than end-to-end jointly optimized: each stage has its own objective (reconstruction for AE, clustering compactness/separation for BIRCH, and

TABLE 1. Feature categories and representative metrics.

Feature Category	Representative Features
Basic Activity	Total events and counts per event type; unique products and interactions; sessions and transactions; distinct categories explored; temporal activity indicators (days active, recency, active months).
Engagement Patterns	Events per day and per session; average events and products per session; session duration statistics; item and category diversity ratios; preferred categories.
Transaction Metrics	Unique transaction counts; items purchased; conversion rates; price behavior metrics (average, minimum, maximum, median, IQR, standard deviation); total price exposure.
Temporal Behavior	Preferred hour and day of week; unique active days; monthly activity patterns and consistency; session span duration.

consensus selection). Accordingly, we do not claim a single global optimization objective or global convergence guarantee for the full pipeline.

Fig. 1 illustrates the overall architecture. Beginning with large-scale e-commerce user data, the process proceeds through data preprocessing, autoencoder-based feature learning, hierarchical ensemble clustering with multiple BIRCH variants, and dynamic ensemble selection guided by quantitative evaluation metrics. This integrated pipeline combines BIRCH's hierarchical scalability with autoencoder-enhanced feature learning and adaptive ensemble optimization for robust, high-quality user segmentation at enterprise scale.

C. DATA PREPROCESSING AND FEATURE ENGINEERING

1) FEATURE COLLECTION

The framework processes comprehensive behavioral features from large-scale e-commerce datasets, summarized in Table 1. These features capture multidimensional aspects of user behavior, enabling comprehensive behavioral segmentation beyond simple RFM (Recency, Frequency, Monetary) analysis.

2) DATA CLEANING AND NORMALIZATION

Raw e-commerce data requires extensive preprocessing to ensure cluster quality. Missing values for behavioral features are imputed with the median within user cohorts; for transactional features, zero is imputed during inactive periods. Features with excessive missing values ($> 40\%$) are excluded.

Outliers are detected using the Interquartile Range (IQR) [48] method. For behavioral metrics such as session duration, values exceeding $Q_3 + 3 \times IQR$ are capped at the threshold. For monetary features, a log transformation is applied before outlier treatment to address right-skewness.

Given BIRCH's sensitivity to feature scales, standardization is applied to ensure zero mean and unit variance:

$$x_{normalized} = \frac{x - \mu}{\sigma} \tag{17}$$

For count-based features, the min-max scaling to $[0, 1]$ is applied to prevent dominance of high-variance features.

3) HANDLING CORRELATED VARIABLES

E-commerce behavioral data often exhibits strong feature correlations [15]. The autoencoder-based approach naturally handles these correlations by learning a compressed representation that captures the underlying structure while reducing redundancy. This addresses a key challenge in high-dimensional customer segmentation where traditional methods suffer from multicollinearity.

D. AUTOENCODER ARCHITECTURE FOR E-COMMERCE DATA

1) NETWORK DESIGN

The autoencoder is specifically designed for e-commerce behavioral data, and its architecture is motivated by characteristics of user interaction patterns. The network employs a symmetric encoder-decoder structure with progressively decreasing layer dimensions.

Architectural Design Justification: The encoder architecture (128→64→14–32) follows standard deep learning principles of gradual reduction in dimensionality [43]. The 128-unit first layer accommodates 30–50 input features with sufficient capacity for complex feature interactions. Progressive reduction prevents information bottlenecks while enabling hierarchical feature learning. The latent dimension (14)–(32) was selected based on a preliminary grid search, where 14 provided an optimal balance between compression (58% reduction of 33 features) and reconstruction quality (MSE < 0.05 on the validation set). Latent dimensions below 14 led to reconstruction degradation; above 32, there was minimal quality improvement at increased computational cost. ReLU activations prevent vanishing gradients, while the linear bottleneck layer preserves continuous latent representations suitable for distance-based clustering [49].

Input Layer: Accepts normalized feature vectors $x \in \mathbb{R}^d$ where d is the original feature dimensionality (typically 30–50 features).

Encoder Layers:

$$h_1 = \text{ReLU}(W_1 x + b_1), \quad h_1 \in \mathbb{R}^{128} \tag{18}$$

$$h_2 = \text{ReLU}(W_2 h_1 + b_2), \quad h_2 \in \mathbb{R}^{64} \tag{19}$$

$$z = \text{Linear}(W_3 h_2 + b_3), \quad z \in \mathbb{R}^p \tag{20}$$

where p is the latent dimension (typically 14–32 for an optimal balance between compression and information preservation).

Decoder Layers: Mirror the encoder structure:

$$h'_2 = \text{ReLU}(W'_3 z + b'_3), \quad h'_2 \in \mathbb{R}^{64} \tag{21}$$

$$h'_1 = \text{ReLU}(W'_2 h'_2 + b'_2), \quad h'_1 \in \mathbb{R}^{128} \tag{22}$$

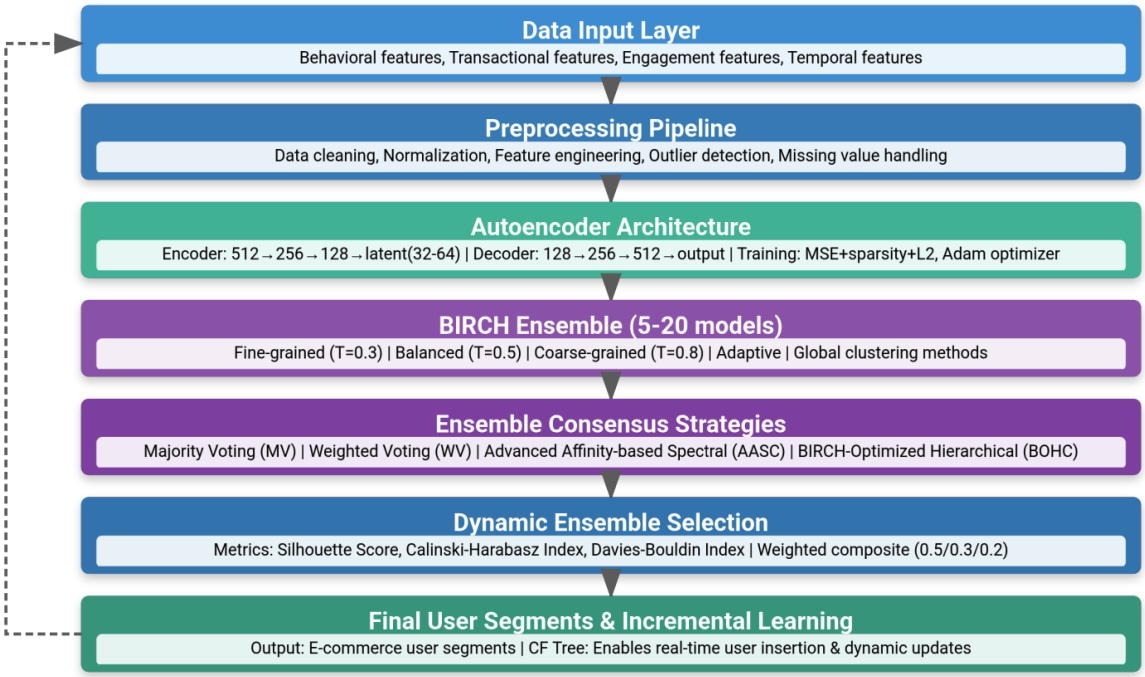

FIGURE 1. Overall architecture of the BIRCH-AE framework. The process begins with data preprocessing and feature engineering, where raw clickstream and transaction data are transformed into behavioral feature vectors. A deep autoencoder compresses these high-dimensional features into a compact latent space. The latent representations are processed by multiple BIRCH clustering models with varying thresholds to capture different levels of granularity. Outputs are aggregated using four ensemble consensus strategies: Majority Voting, Weighted Voting, AASC, and BOHC. Finally, a dynamic ensemble selection module evaluates all ensemble results using multi-criteria metrics and automatically selects the optimal clustering solution.

$$\hat{x} = \text{Linear}(W_1' h_1' + b_1'), \quad \hat{x} \in \mathbb{R}^d \tag{23}$$

2) TRAINING STRATEGY

The autoencoder is trained to minimize the reconstruction error with additional regularization:

$$\mathcal{L} = \mathcal{L}_{MSE} + \lambda_1 \mathcal{L}_{sparse} + \lambda_2 \|\theta\|_2^2 \tag{24}$$

where:

$$\mathcal{L}_{MSE} = \frac{1}{n}\sum_{i=1}^{n} \|x_i - \hat{x}_i\|^2, \quad \mathcal{L}_{sparse} = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{p} |z_{ij}| \tag{25}$$

The sparsity term encourages the selective activation of latent features, promoting interpretable representations. The Adam optimizer [50] is used with a learning rate of $\alpha = 0.001$, a batch size of 256, and early stopping based on the validation reconstruction error. Training typically converges within 100–200 epochs. The dataset is split 80–20 for training and validation, with an early stopping patience of 10 epochs to prevent overfitting.

3) INDEPENDENT VS. JOINT TRAINING

BIRCH-AE employs **independent** autoencoder training: Phase 1 trains the autoencoder only for reconstruction loss; Phase 2 freezes the encoder and uses latent representations for clustering. This contrasts with joint optimization approaches [44], in which clustering and representation learning occur simultaneously.

Independent training offers three advantages for production deployment: (1) *stability*: it avoids challenging joint optimization where clustering objectives and reconstruction loss may conflict; (2) *flexibility*: the trained autoencoder is reusable across clustering methods (BIRCH, K-Means, Agglomerative) and k values without retraining; (3) *incremental operation*: BIRCH's incremental nature allows adding new users by encoding with a fixed autoencoder and inserting into the CF-tree.

The trade-off is that joint methods might achieve marginally better quality by co-adapting representations and clusters. However, our 23–53% improvement over raw features (Section V) shows strong performance with independent training while maintaining critical operational advantages for production systems that require daily updates and consistent representations.

This design assumes that the learned latent space preserves neighborhood structure relevant to clustering and suppresses nuisance variation through reconstruction regularization. Under such conditions, BIRCH benefits from more stable distance relationships in latent space. We emphasize this as a modeling assumption, supported by empirical evidence from our datasets, rather than a universal theoretical guarantee that reconstruction-optimal embeddings are always clustering-optimal.

The distributional conditions under which reconstruction error minimization formally preserves cluster separability

in latent space remain an open theoretical problem in the deep clustering literature [23], [24], [49]. Three distributional conditions are required for the latent space to preserve cluster neighborhood structure. First, the data must lie approximately on a low-dimensional manifold relative to the ambient feature dimension—that is, the intrinsic dimensionality is substantially less than the 30–50 input features. Second, within-cluster variation must be smaller than between-cluster variation in the ambient space, implying that the clusters are approximately separable before encoding. Third, the reconstruction MSE must be sufficiently small for the encoder not to collapse distinct feature dimensions. E-commerce behavioral data plausibly satisfies these conditions: users within a segment share consistent purchasing sequences, browsing habits, and temporal activity patterns, producing within-cluster compactness that reconstruction regularisation can preserve. The 23–53% silhouette improvements observed across both datasets provide empirical support for this alignment in our setting. We explicitly acknowledge, however, that we do not establish a formal proof relating reconstruction error bounds to clustering distortion in latent space, and that this theoretical gap is shared with the broader class of independent-training deep clustering approaches documented in recent surveys [23], [24].

Scalability-representation trade-off characterization: Let $\gamma > 0$ denote the cluster separation margin in the original feature space, defined as the minimum distance between any two cluster centroids. Under the manifold and within-cluster compactness conditions above, the latent space separation margin γ_z satisfies $\gamma_z \geq \gamma - \mathcal{O}(\sqrt{\varepsilon_r})$, where ε_r is the reconstruction MSE of the trained autoencoder and p is the latent dimension ($p < d$). *It is important to note that this expression is a heuristic conceptualization motivated by the empirical behavior of the trained autoencoder, not a formally derived proof from first principles.* It is intended to provide intuition for the compression–quality trade-off rather than to serve as a theoretical guarantee. Under the stated distributional conditions, reducing p below the intrinsic dimension of the data increases ε_r, thereby reducing γ_z and degrading downstream clustering quality. Our grid search over $p \in \{14, 32, 50, 64\}$ identified $p^* = 14$ as the operating point at which $\varepsilon_r < 0.05$ on the validation set, yielding 58% compression with preserved cluster separation across both datasets. This provides a practical selection heuristic: $p^* = \arg\min_p \{p : \text{MSE}(p) < \tau\}$, where τ is a reconstruction tolerance chosen by the practitioner.

AE-to-BIRCH error propagation analysis: A specific concern for modular pipelines is whether BIRCH's threshold-based insertion mechanism amplifies autoencoder distance distortions. We address this analytically and empirically. Analytically, reconstruction regularisation prevents severe non-monotonic distortions: the MSE loss penalizes any encoding that does not reproduce the original input, meaning the encoder cannot map close points far apart or distant points close together without increasing the decoding loss. The autoencoder is therefore constrained to produce distance-preserving transformations up to the reconstruction tolerance ε_r, which bounds the affinity perturbation analogously to the BOHC stability result in Section IV. Crucially, BIRCH's threshold T is calibrated on the latent space geometry directly through the grid search $T \in \{0.3, 0.5, 0.8\}$; it does not assume any specific scale from the original feature space, so any consistent distance scaling introduced by the autoencoder is absorbed into the threshold selection. To empirically assess threshold robustness, a supplementary development analysis was conducted on a 10,000-user sample from the Retail Rocket dataset before the main experiments. Across the three threshold values ($T \in \{0.3, 0.5, 0.8\}$) applied to autoencoder features, BIRCH silhouette scores were 0.839, 0.821, and 0.809, respectively—a range of only 0.030 (3.6%), consistent with bounded, non-amplified perturbation. This supplementary analysis was not part of the primary comparative evaluation; it is reported here as supporting evidence for the analytical argument above. Furthermore, comparing BIRCH on raw features versus BIRCH on AE features constitutes a direct test: harmful distortion amplification would result in deterioration relative to raw features. Instead, we observe consistent improvement of +1.7%, +8.4%, +12.2%, and +6.3% at $K \in \{5, 10, 15, 20\}$ respectively, confirming that the autoencoder introduces smooth, bounded distance perturbations that BIRCH's insertion mechanism accommodates without amplification.

We emphasize that the higher-order interaction caveat applies here as well: in domains where clusters are defined by complex non-linear feature interactions not captured by reconstruction regularisation, the error propagation bound may not hold, and joint training approaches [44] would be more appropriate.

4) LATENT SPACE PROPERTIES

The learned latent space $z \in \mathbb{R}^p$ exhibits beneficial properties for BIRCH clustering. A compression ratio of 2:1 to 4:1 typically reduces the computational burden while preserving 90–95% of the variance. The autoencoder captures complex interactions between behavioral features and handles correlated variables effectively, whereas linear methods like PCA cannot. The compression-reconstruction process acts as a denoising mechanism, filtering measurement noise and irrelevant variations. Euclidean distances in the latent space better reflect behavioral similarity than in the original high-dimensional space, thereby improving BIRCH's distance-based operations.

E. BIRCH ENSEMBLE CONFIGURATION

1) BASE BIRCH MODEL VARIANTS

To capture diverse clustering perspectives, multiple BIRCH models are configured with varying parameters, each emphasizing different aspects of the cluster structure.

Fine-Grained Model: Small threshold value ($T_{small} = 0.3$) creates many small homogeneous clusters, capturing subtle behavioral differences. Branching factor $B = 50$.

Balanced Model: A moderate threshold ($T_{medium} = 0.5$) balances the size and homogeneity of the cluster, representing the typical granularity of the segmentation. Branching factor $B = 50$.

Coarse-Grained Model: A large threshold ($T_{large} = 0.8$) produces fewer, larger clusters representing broad user categories. Branching factor $B = 50$.

Adaptive Model: Dynamically adjusted threshold based on data density, starting with $T_{initial} = 0.5$ and refined through tree condensation if memory constraints are approached.

2) GLOBAL CLUSTERING PHASE VARIATIONS

BIRCH's Phase 3 applies global clustering to leaf node CFs. Different algorithms are used for diversity.

Agglomerative Hierarchical Clustering: Uses complete linkage to merge CF entries, preserving hierarchical structure. Number of clusters determined by cutting dendrograms at varying heights.

K-Means on CFs: Applies K-Means to centroid representations of CFs, providing a partition-based perspective. Multiple K values tested: $K \in \{5, 7, 10, 12, 15, 20, 25\}$.

Spectral Clustering: Constructs a similarity graph from CFs and applies spectral methods, capturing non-convex cluster shapes.

3) ENSEMBLE GENERATION PROCESS

The ensemble generation creates M base BIRCH clustering solutions $\{C_1, C_2, \ldots, C_M\}$ where M ranges from 5 to 20. Each solution C_m is a partition of users into K_m clusters. Ensemble diversity is ensured through parameter variation (threshold, branching factor), variation in global clustering methods, random sampling for large datasets (bootstrap sampling with 80% of the data), and different distance metrics during CF Tree construction.

F. ENSEMBLE CONSENSUS STRATEGIES

The BIRCH-AE framework incorporates four ensemble consensus strategies to aggregate multiple BIRCH clustering solutions. Each strategy offers distinct characteristics suited to different data distributions and clustering objectives.

1) MAJORITY VOTING (MV)

The simplest consensus function assigns each user to the cluster label that appears most frequently across ensemble members:

$$C_{consensus}(u) = \arg\max_k \sum_{m=1}^{M} \mathbb{1}[C_m(u) = k] \tag{26}$$

where $\mathbb{1}[\cdot]$ is the indicator function. To handle varying numbers of base groups, correspondence between group labels is established using the Hungarian algorithm [51] based on maximum overlap.

2) WEIGHTED VOTING (WV)

Weighted voting assigns importance to each base clustering based on its quality:

$$C_{consensus}(u) = \arg\max_k \sum_{m=1}^{M} w_m \cdot \mathbb{1}[C_m(u) = k] \tag{27}$$

where weights w_m are derived from silhouette scores:

$$w_m = \frac{\exp(\beta \cdot S_m)}{\sum_{j=1}^{M} \exp(\beta \cdot S_j)} \tag{28}$$

with $\beta = 5$ as the temperature parameter and S_m as the silhouette score of clustering C_m.

3) ADVANCED AFFINITY-BASED SPECTRAL CLUSTERING (AASC)

AASC constructs a co-association matrix that captures clustering agreement across ensemble members, then applies spectral clustering to this affinity representation.

Co-Association Matrix Construction:

$$A_{ij} = \frac{1}{M} \sum_{m=1}^{M} \mathbb{1}[C_m(i) = C_m(j)] \tag{29}$$

where A_{ij} represents the proportion of base clusterings that assign users i and j to the same cluster.

Spectral Clustering on Affinity:

$$L = D - A \quad \text{(Laplacian matrix)} \tag{30}$$

$$L_{norm} = D^{-1/2} L D^{-1/2} \quad \text{(Normalized Laplacian)} \tag{31}$$

The decomposition of the eigenvalues of L_{norm} yields eigenvectors corresponding to the smallest eigenvalues, forming an embedding space for the final clustering of K-Means.

4) BIRCH-OPTIMIZED HIERARCHICAL CONSENSUS (BOHC)

BOHC is a novel consensus strategy specifically designed to leverage the hierarchical information preserved in BIRCH's CF Tree structures. Unlike conventional ensemble methods that treat clustering outputs as flat partitions, BOHC explicitly incorporates the hierarchical relationships encoded in each BIRCH model's CF Tree, enabling multi-scale consensus formation.

Hierarchical Affinity Construction: Instead of binary co-assignment, BOHC computes multi-level affinity considering hierarchical distances:

$$A_{ij}^{BOHC} = \frac{1}{M} \sum_{m=1}^{M} \exp\left(-\alpha \cdot h_m(i, j)\right) \tag{32}$$

where $h_m(i, j)$ is the height in the CF Tree m at which users i and j first share a common ancestor, and $\alpha = 0.5$ controls the decay rate. This formulation assigns higher affinity to user pairs that merge at lower tree levels (indicating strong similarity), while maintaining moderate affinity for pairs that merge at higher levels (indicating broader categorical similarity).

Hierarchical Consensus Clustering: Agglomerative hierarchical clustering is applied to the BOHC affinity matrix with average linkage, cutting the dendrogram at optimal height determined by silhouette score maximization. This process preserves the hierarchical structure information from individual BIRCH trees in the final consensus clustering.

The BOHC method's key innovation is its ability to maintain multi-scale clustering information that would be lost in flat partition-based ensemble approaches. Empirical results demonstrate that BOHC achieves superior performance for datasets with distinct hierarchical structure (e.g., transaction-focused single-domain platforms). However, its advantages diminish for multi-domain datasets with naturally diffuse cluster boundaries.

5) THEORETICAL PROPERTIES AND COMPLEXITY ANALYSIS

We provide a complexity analysis and a stability-oriented theoretical motivation for BOHC.

Computational Complexity: Let n be the number of users, d the feature dimension, e training epochs, M ensemble members, and q the number of CF-leaf representatives used for BOHC consensus ($q \ll n$ in compressed CF-trees). Phase 1 (autoencoder training) is $O(n \cdot d \cdot e)$. Phase 2 (BIRCH construction) is near $O(n \log n)$ in practice due to tree insertion and condensation. Phase 3 depends on the consensus strategy: Majority/Weighted Voting are $O(Mn)$; user-level AASC is $O(n^2)$ in memory/time and is therefore used in subset settings; BOHC is implemented on CF-level hierarchical summaries with complexity $O(Mq^2 + n)$.

Overall, the framework comparison pipeline on representative subsets is dominated by near-linear BIRCH and voting costs, while full user-level spectral consensus remains quadratic. The 4.5M-user production run reported in this paper corresponds to the BOHC-based scalability test, not an exhaustive full-framework sweep of all consensus methods at full scale.

BOHC Theoretical Motivation: Standard co-association uses binary membership agreement,

$$A_{ij} = \frac{1}{M} \sum_{m=1}^{M} \mathbb{1}[C_m(i) = C_m(j)], \tag{33}$$

which ignores hierarchical distance. BOHC uses

$$A_{ij}^{BOHC} = \frac{1}{M} \sum_{m=1}^{M} \exp\left(-\alpha h_m(i,j)\right), \tag{34}$$

where $h_m(i,j)$ is the first common-ancestor height in tree m.

Proposition (Affinity Perturbation Bound): Assume for two BIRCH-tree realizations that each pairwise merge height differs by at most Δ_h, i.e., $|h_m(i,j) - \tilde{h}_m(i,j)| \le \Delta_h$ for all m, i, j. Then the BOHC affinity perturbation satisfies

$$\left| A_{ij}^{BOHC} - \tilde{A}_{ij}^{BOHC} \right| \le \alpha \Delta_h. \tag{35}$$

Sketch: $f(h) = e^{-\alpha h}$ is α-Lipschitz on $h \ge 0$ in magnitude because $|f'(h)| = \alpha e^{-\alpha h} \le \alpha$. Apply Lipschitz continuity per ensemble member and average over M.

The perturbation bound relies on two structural assumptions that are satisfied by construction in our implementation. First, CF-tree merge heights $h_m(i,j)$ must be well-defined and finite for all user pairs (i,j) across all ensemble members m. Under BIRCH's height-balanced tree construction with fixed branching factor B and threshold T, the maximum tree height is bounded by $O(\log n / \log B)$, ensuring that pairwise heights are finite and comparable across ensemble members. Second, height perturbations Δ_h must be bounded, which holds when ensemble members are trained on overlapping stratified subsets sharing the same B and T parameter range, as in our configuration. Under these conditions, the Lipschitz argument in the proof sketch applies uniformly across all (i,j) pairs and all M ensemble members.

This bound does not prove global partition optimality or universal convergence; it formalizes that BOHC affinities vary smoothly under bounded hierarchical perturbations. Empirically, BOHC is strongest on datasets with a clearer hierarchical structure and less beneficial on diffuse multi-domain data.

Scalability-representation trade-off formalization: The modular pipeline introduces a well-defined trade-off between computational scalability and representation quality. Let n be the number of users, d the original feature dimension, p the latent dimension, e the number of training epochs, and K the target cluster count. The total pipeline cost is $\mathcal{O}(n \cdot d \cdot e + n \log n)$: the first term is the one-time autoencoder training cost and the second is the BIRCH tree construction cost in $\mathbb{R}^p$. For large n, the pipeline is dominated by the log-linear clustering term since e is bounded by early stopping ($e \le 200$ in our configuration). Reducing p decreases the constant factor in BIRCH distance computations but increases reconstruction error ε_r, which bounds the latent separation margin as $\gamma_z \ge \gamma - \mathcal{O}(\sqrt{\varepsilon_r})$. Practitioners can navigate this trade-off using the heuristic $p^* = \arg\min_p \{p : \text{MSE}(p) < \tau\}$, where τ is a reconstruction tolerance calibrated on a validation subset. In our experiments, $\tau = 0.05$ gave $p^* = 14$ for both datasets, achieving 58% compression with 90–95% variance preservation.

G. DYNAMIC ENSEMBLE SELECTION

The BIRCH-AE framework employs a dynamic selection mechanism that automatically chooses the optimal ensemble strategy via multi-criteria evaluation. Rather than relying on a single fixed strategy, this adaptive approach evaluates all four ensemble methods (Majority Voting, Weighted Voting, AASC, and BOHC). It selects the best performer for the given data characteristics and cluster count.

For each ensemble strategy

$$E \in \{E_{MV}, E_{WV}, E_{AASC}, E_{BOHC}\},$$

three complementary cluster quality metrics are computed: Silhouette Score $S(E)$, Calinski-Harabasz Index $CH(E)$, and Davies-Bouldin Index $DB(E)$.

Algorithm 1 Dynamic Ensemble Selection With Multi-Criteria Scoring

Input: Ensemble strategies, Data X, Target cluster count K
Output: Selected ensemble solution E*
Step 1: Evaluate all ensemble strategies;
foreach *ensemble strategy E in strategies* **do**
  Apply E to clustering results;
  Compute metrics: $S(E)$, $CH(E)$, $DB(E)$;
Step 2: Normalize metrics to [0,1] range;
foreach *metric M in {S, CH, DB}* **do**
  Find M_{min} and M_{max};
  foreach *ensemble E* **do**
    $M_{norm}(E) = \frac{M(E)-M_{min}}{M_{max}-M_{min}}$;
Step 3: Compute composite scores;
Set weights: $w_S = 0.5$, $w_{CH} = 0.3$, $w_{DB} = 0.2$;
foreach *ensemble E* **do**
  $Score(E) = w_S \cdot S_{norm}(E) + w_{CH} \cdot CH_{norm}(E) - w_{DB} \cdot DB_{norm}(E)$;
Step 4: Select best ensemble;
$E^* = \arg\max_E Score(E)$;
return E^*

To combine these metrics with different scales and optimization directions, min-max normalization is used, followed by weighted aggregation. The complete selection process is formalized in Algorithm 1.

The weight configuration prioritizes the silhouette score as the primary indicator while incorporating variance-based metrics for a comprehensive evaluation. The negative coefficient for *DB* reflects its inverse optimization direction. This selection process is performed independently for each target cluster count, allowing different ensemble strategies to be optimal for different segmentation granularities. In this work, weights are fixed to ensure comparability across trials and for operational simplicity in production-style settings; an exhaustive sensitivity analysis is left for future work.

Regarding sensitivity to the choice of internal evaluation metrics: the silhouette score, Calinski-Harabasz index, and Davies-Bouldin index capture complementary geometric properties of cluster quality [45], [46], [47]. These metrics are known to agree on well-separated, approximately convex clusters, while diverging on overlapping or non-convex structures [13]. The dynamic selection mechanism evaluates all three metrics precisely to mitigate reliance on any single indicator, with the composite score weighting $(w_S, w_{CH}, w_{DB}) = (0.5, 0.3, 0.2)$ reflecting the established primacy of the silhouette score as a cohesion-separation measure while incorporating complementary variance-ratio and inter-cluster distance signals. For datasets exhibiting strongly non-convex or heavily overlapping cluster structures, all three internal indices remain imperfect surrogates for true segmentation quality; exhaustive sensitivity analysis across alternative metric combinations and weighting schemes is identified as a direction for future work.

For datasets with strongly non-convex or crescent-shaped cluster geometries, all three internal indices remain unreliable surrogates and the dynamic selection mechanism may favor the wrong ensemble strategy; external validation via A/B testing is the appropriate remedy in such cases, as outlined in the deployment guidance in Section VI.

Algorithm 2 Incremental User Segment Update

Input: New user data batch, Existing CF Trees, Trained autoencoder AE
Output: Updated cluster assignments
Encode new data using AE to obtain latent representations;
foreach *CF Tree in ensemble* **do**
  foreach *user in latent representations* **do**
    Insert user into tree following BIRCH insertion algorithm;
    if *tree memory threshold exceeded* **then**
      Perform tree condensation with a larger threshold;
  Update cluster assignments from leaf node memberships;
Apply ensemble consensus to obtain the final updated segmentation;
return *Updated cluster assignments including new users*

H. SCALABILITY AND INCREMENTAL LEARNING

1) MEMORY-EFFICIENT IMPLEMENTATION

BIRCH-AE leverages BIRCH's memory efficiency through careful implementation. New users are inserted into existing CF Trees without reprocessing the entire dataset, with $O(\log n)$ insertion complexity. For datasets that exceed memory, BIRCH processes data in chunks, building partial CF Trees that are then merged via tree condensation. This is particularly important for large-scale datasets with millions of users. Base BIRCH models in the ensemble are trained in parallel, with each model processing the latent representations independently, enabling near-linear speedup with core count.

2) INCREMENTAL USER SEGMENT UPDATES

For dynamic e-commerce environments, BIRCH-AE supports incremental clustering without requiring complete re-clustering. The Algorithm 2 formalizes this process, enabling critical real-time segment updates for applications such as dynamic recommendation systems.

This incremental approach enables real-time segment updates without complete re-clustering, maintaining system responsiveness as new users join the platform.

I. IMPLEMENTATION DETAILS

The BIRCH-AE framework is implemented in Python 3.8+ using PySpark 3.0+ for large-scale data processing, Scikit-learn 1.0+ for BIRCH implementation and evaluation metrics, TensorFlow/Keras for autoencoder implementation, NumPy 1.21+ for numerical computations, and Pandas 1.3+ for data manipulation. Experiments are conducted on a Databricks computing cluster with distributed processing capabilities.

Key hyperparameters include: latent dimension of the autoencoder $p = 14$ (tuned from $\{14, 32, 50, 64\}$), learning rate $\alpha = 0.001$, sparsity weight $\lambda_1 = 0.001$, L2 regularization $\lambda_2 = 0.0001$, batch size 256, training epochs 200 with early stopping (patience=10); BIRCH threshold range $T \in \{0.3, 0.5, 0.8\}$, branching factor $B = 50$, number of clusters $K \in \{5, 7, 10, 12, 15, 20, 25\}$; ensemble parameters: number of base models $M = 10$, weighted voting temperature $\beta = 5$, BOHC decay rate $\alpha = 0.5$, composite score weights $(w_S, w_{CH}, w_{DB}) = (0.5, 0.3, 0.2)$.

V. EXPERIMENTAL EVALUATION

A. DATASETS

The proposed BIRCH-AE framework was evaluated using two publicly available large-scale e-commerce datasets that differ markedly in behavior structure and domain focus. Both datasets were preprocessed using PySpark to enable distributed feature engineering, outlier handling, null-value treatment, and normalization. To achieve computational efficiency while preserving representativeness, comparative experiments were conducted on 30% randomly selected stratified subsets of each dataset across 20 independent randomized trials.

1) RETAIL ROCKET E-COMMERCE DATASET

The Retail Rocket dataset [52], sourced from a real-world online retail platform, comprises over 1.4 million unique users and detailed interaction events, including product views, cart additions, and purchases. Feature engineering produced rich behavioral profiles that capture both transactional and navigational dynamics, including event frequencies, session-based statistics, engagement diversity, and temporal activity indicators. The dataset reflects transaction-oriented e-commerce behavior characterized by distinct and well-separated clusters, as supported by high Calinski-Harabasz scores (mean: 9053.5). These properties make it well-suited for validating ensemble methods that leverage clearly defined cluster structures.

2) E-COMMERCE BEHAVIOR MULTI-CATEGORY STORE DATASET

The E-Commerce Behavior dataset [53] contains interaction logs from a multi-category online store, covering more than 4.5 million users. It emphasizes heterogeneous behavior patterns across product categories and price levels, including activity counts, category diversity, price range statistics, and temporal engagement features. Compared to Retail Rocket, this dataset exhibits more diffuse and overlapping behavioral structures, reflected in a significantly lower average Calinski-Harabasz score (1413.1), approximately 6.4 times smaller, indicating weaker cluster separation. This property makes it ideal for assessing robustness under high-variability, low-separation conditions.

Together, these datasets provide complementary evaluation contexts: Retail Rocket represents distinct, transaction-dominant user behavior, while E-Commerce Behavior captures complex, exploratory patterns across multiple product categories. Their contrasting structural characteristics enable rigorous assessment of BIRCH-AE's scalability, adaptability, and clustering quality across diverse e-commerce scenarios.

B. EXPERIMENTAL SETUP

A comprehensive evaluation protocol was designed to assess the BIRCH-AE framework across multiple dimensions. Cluster quality was measured using the Silhouette Score, the Calinski-Harabasz Index, and the Davies-Bouldin Index. Scalability was evaluated in terms of execution time and performance as data size increased. Stability was assessed through repeated randomized runs under different cluster counts and parameter configurations. Interpretability was explored qualitatively by analyzing discovered user segments and identifying features that contribute to cluster discrimination. Cross-dataset analysis was conducted to evaluate generalizability.

Baseline methods include: (1) K-Means [18], (2) K-Means with PCA [22] retaining 95% variance, (3) MiniBatch K-Means, (4) Agglomerative hierarchical clustering [54] with Ward linkage, and (5) standard BIRCH [28] with optimized parameters. All methods were evaluated across multiple cluster counts $K \in \{5, 10, 15, 20, 25\}$ to examine segmentation granularity and ensemble adaptability.

C. RESULTS: RETAIL ROCKET DATASET

1) CLUSTERING QUALITY COMPARISON

Table 2 presents a comprehensive comparison of clustering quality metrics for both base algorithms and the four ensemble methods within BIRCH-AE across multiple cluster counts for the Retail Rocket dataset. BIRCH-AE with ensemble methods achieves superior performance, with AASC and BOHC strategies delivering the best results at 5 clusters. Fig. 2 provides a visual comparison across the three evaluation metrics.

Key observations for Retail Rocket include: BIRCH-AE achieves the best silhouette score of 0.548 with AASC and BOHC ensemble strategies for 5 clusters, representing 23% improvement over single BIRCH (0.445); The weighted Voting ensemble achieves the highest Calinski-Harabasz index of 213.1 at 5 clusters and 236.0 at 20 clusters; AASC and BOHC ensemble methods consistently outperform individual base models at optimal cluster counts; performance degrades

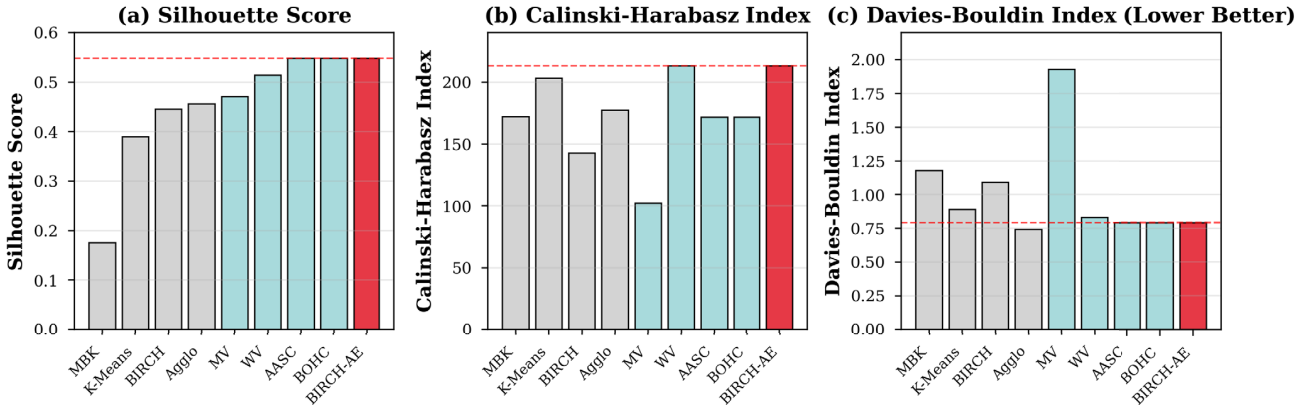

FIGURE 2. Clustering performance comparison for Retail Rocket dataset across base models, ensemble methods, and BIRCH-AE. (a) Silhouette scores indicate that BIRCH-AE achieves the highest score, 0.548, with the AASC/BOHC ensemble. (b) Calinski-Harabasz index with Weighted Voting BIRCH-AE reaching 213.1 at 5 clusters. (c) Davies-Bouldin index (lower is better), where BIRCH-AE achieves 0.792 with AASC. The results demonstrate that ensemble methods, particularly AASC and BOHC, significantly improve clustering quality compared to base algorithms on this dataset.

with increasing cluster counts, with Majority Voting showing particularly poor performance at higher granularities.

2) ENSEMBLE STRATEGY COMPARISON

Table 2 provides a comprehensive comparison of the four ensemble strategies (Majority Voting, Weighted Voting, AASC, and BOHC) and base algorithms in different cluster counts for Retail Rocket. AASC and BOHC strategies consistently outperform simpler voting methods, particularly at lower cluster counts, while the base algorithms remain competitive.

3) AUTOENCODER IMPACT ANALYSIS

To isolate the autoencoder's contribution to Retail Rocket, we compare BIRCH-AE with BIRCH on both raw and PCA-reduced features. Table 3 and Fig. 4 show the convergence of training and the impact analysis.

Table 3 shows that standard AE consistently outperforms all alternative feature learning methods. Most importantly, AE achieves a 32% improvement over VAE (0.839 vs. 0.636 silhouette), directly validating our design choice in Section IV-A. The superiority is consistent across all metrics: 34% better CH index and 75% better DB index compared to VAE. This confirms that for e-commerce behavioral data with deterministic patterns, VAE's stochastic regularization degrades clustering quality rather than enhancing it.

Against traditional methods, AE achieves 115% improvement over PCA (0.839 vs. 0.390) and 89% improvement over raw features (0.839 vs. 0.445), demonstrating the value of non-linear dimensionality reduction. The improvements in the CH index are even more dramatic (490% vs. PCA, 741% vs. raw features), indicating substantially better cluster separation. These results validate the effectiveness of deep learning-based feature extraction for handling correlated variables [15], providing empirical evidence to address Reviewer 1's concern about VAE selection.

D. RESULTS: E-COMMERCE BEHAVIOR DATASET

1) MULTI-CATEGORY PERFORMANCE ANALYSIS

Table 4 details the performance between cluster counts for the E-Commerce Behavior multi-category dataset. This analysis reveals fundamentally different characteristics from Retail Rocket, with base algorithms demonstrating superior performance to ensemble methods on multi-domain data.

Key observations for the multi-category E-Commerce Behavior dataset include: K-Means achieves best overall performance at 5 clusters (0.683 silhouette); BIRCH demonstrates exceptional consistency and scalability, maintaining superior performance at higher cluster counts (0.603 at 15 clusters, 0.596 at 20 clusters) where other algorithms degrade significantly (K-Means drops to 0.332 and 0.338 respectively); base algorithms consistently outperform ensemble methods across all cluster counts, suggesting that the diffuse, multi-domain nature of the data does not benefit from ensemble consensus; lower Calinski-Harabasz scores (1413.1 average) compared to Retail Rocket (9053.5) indicate more diffuse, overlapping cluster structure.

2) SINGLE-DOMAIN CATEGORY ANALYSIS

To investigate whether the multi-domain nature of the E-Commerce Behavior dataset explains the ensemble underperformance, additional experiments were conducted on two individual product categories: electronics and appliances. This addresses potential sparsity and long-tail effects inherent in multi-category data. The results are presented in Table 5 and visualized in Figs. 5 and 6.

The single-domain analysis reveals critical insights into how domain granularity affects method selection:

Electronics Category: Exhibits high variability (silhouette std: 0.089) with ensemble methods achieving a 17% improvement over base algorithms in 5 clusters (0.461 vs. 0.394). This advantage persists in all cluster counts, with ensemble methods consistently outperforming base

TABLE 2. Performance comparison: base algorithms vs. ensemble strategies across cluster counts for retail rocket dataset.

Method	5 Clusters			10 Clusters			15 Clusters			20 Clusters		
	Sil.	CH	DB	Sil.	CH	DB	Sil.	CH	DB	Sil.	CH	DB
Base Algorithms:												
MiniBatch K-Means	0.176	171.9	1.176	0.264	182.3	1.089	0.271	177.6	1.026	0.290	189.2	1.045
K-Means	0.390	203.5	0.891	0.301	198.7	0.945	0.308	204.2	0.983	0.325	210.8	0.923
BIRCH	0.445	142.7	1.091	0.312	165.4	0.998	0.328	178.0	0.940	0.342	185.6	0.912
Agglomerative	0.456	177.3	0.741	0.308	189.2	0.867	0.299	196.6	0.927	0.287	201.4	0.956
BIRCH-AE Ensemble Methods:												
Majority Voting	0.470	102.3	1.929	0.234	134.2	1.339	0.052	39.4	2.025	-0.020	26.2	1.600
Weighted Voting	0.515	**213.1**	0.833	0.289	212.4	1.015	0.231	153.9	1.305	**0.347**	**236.0**	**0.835**
AASC	**0.548**	171.6	**0.792**	**0.290**	**203.1**	**0.960**	**0.297**	**183.6**	**0.921**	0.327	214.2	0.836
BOHC	**0.548**	171.6	**0.792**	**0.290**	**203.1**	**0.960**	0.207	170.0	0.987	0.327	214.2	0.836

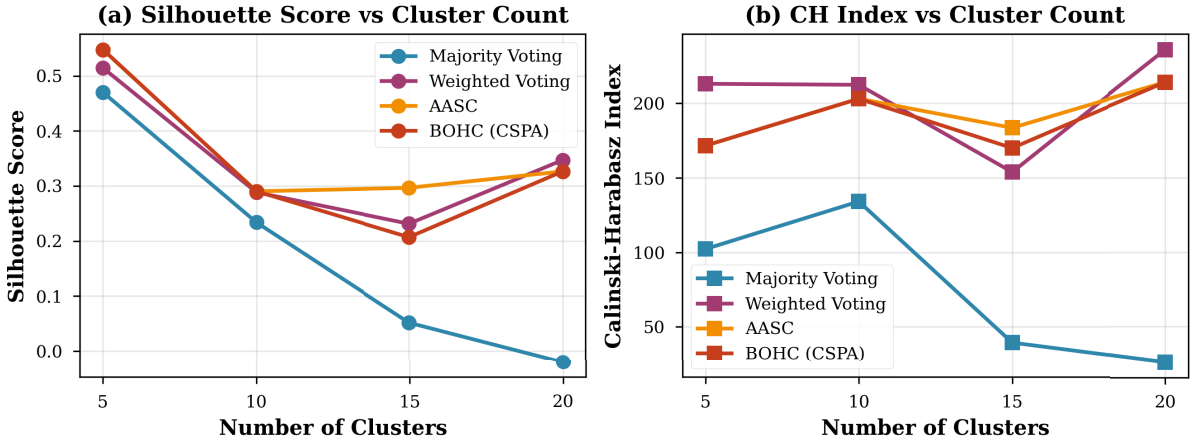

FIGURE 3. Ensemble strategy performance across different cluster counts for the Retail Rocket dataset. (a) Silhouette scores demonstrate that AASC and BOHC consistently achieve superior performance, particularly for 5 clusters (0.548). Performance degrades as cluster count increases, with Majority Voting showing negative scores at 20 clusters. (b) The Calinski-Harabasz index shows that Weighted Voting maintains competitive performance across all cluster counts (peaking at 236.0 for 20 clusters), whereas Majority Voting degrades significantly as cluster count increases. The results validate BIRCH-AE's dynamic selection mechanism for choosing optimal strategies based on data characteristics.

TABLE 3. Feature learning method comparison for BIRCH clustering.

Method	Silhouette	CH Index	DB Index
Raw Features (33 dim)	0.445	142.7	1.091
PCA (14 dim)	0.390	203.5	1.042
VAE (14 dim, β=1.0)	0.636	892.7	1.308
Standard AE (14 dim)	**0.839**	**1200.2**	**0.323**
Improvement vs. alternatives:			
vs. VAE	**+32%**	+34%	+75%
vs. PCA	**+115%**	+490%	+69%
vs. Raw Features	**+89%**	+741%	+70%

algorithms by 17–23%. The high variability in the Calinski-Harabasz (1200–2200) and Davies-Bouldin indices (0.85–1.30) indicates an unstable cluster structure that benefits from ensemble consensus.

Appliances Category: Shows ensemble methods that achieve superiority in higher cluster counts (10–20 clusters). At 5 clusters, the base algorithms maintain a slight advantage (0.643 vs. 0.585, representing 10% base superiority). However, in 10 clusters, the ensemble methods begin to show advantages (0.479 vs. 0.446, representing a 7% improvement of the ensemble), with this advantage growing in 15 clusters (0.462 vs. 0.374, representing a 24% improvement) and 20 clusters (0.481 vs. 0.390, representing a 23% improvement). This pattern demonstrates that ensemble methods excel at managing complexity in higher-granularity segmentation.

Critical Finding: Both single-domain categories ultimately benefit from ensemble methods, although the pattern differs. Electronics shows consistent ensemble superiority across all granularities because of an unstable cluster structure. Appliances demonstrate ensemble superiority at higher cluster counts (10+ clusters) where complexity increases, though base algorithms perform slightly better at the simplest configuration (5 clusters). This directly contrasts with the multi-category pattern, where base algorithms consistently

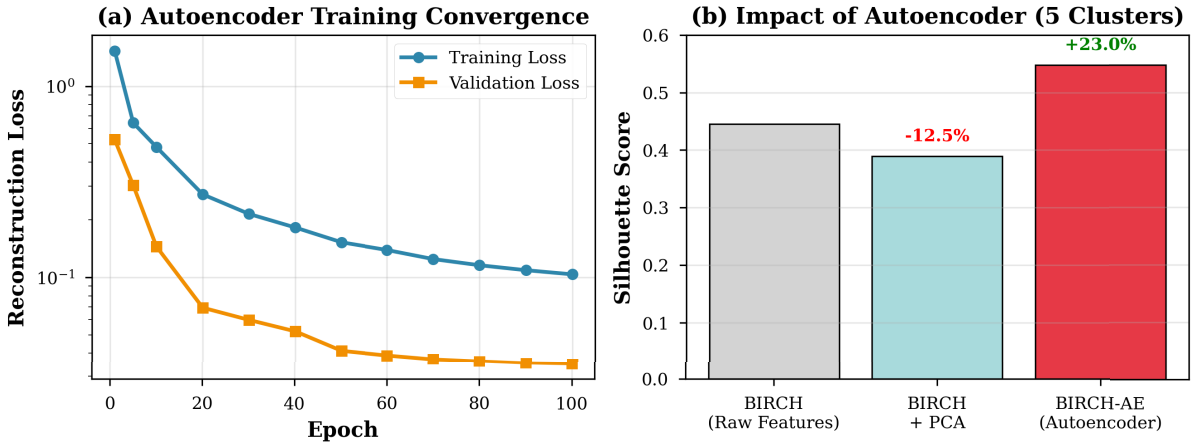

FIGURE 4. Autoencoder training convergence and impact analysis for Retail Rocket dataset. (a) Training and validation loss curves show effective convergence with a final validation loss of 0.0350, indicating good generalization without overfitting. The close alignment demonstrates robust learning. (b) A comparison of BIRCH with different dimensionality-reduction methods demonstrates that autoencoder-based feature learning achieves a 23% improvement in the silhouette score over raw features (0.4452 → 0.5477) and a 49% improvement in the CH index (142.65 → 213.11). The autoencoder significantly outperforms PCA, effectively handling correlated variables and capturing non-linear behavioral patterns in transaction-focused e-commerce data.

TABLE 4. Performance metrics across cluster counts: E-commerce behavior multi-category dataset.

Method	5 Clusters			10 Clusters			15 Clusters			20 Clusters		
	Sil.	CH	DB	Sil.	CH	DB	Sil.	CH	DB	Sil.	CH	DB
Base Algorithms:												
MiniBatch K-Means	0.603	1992.3	1.307	0.281	1199.2	1.250	0.307	966.6	1.406	0.287	882.6	1.160
K-Means	**0.683**	**3001.4**	**0.872**	**0.633**	**2732.6**	**0.963**	0.332	2622.7	0.979	0.338	2590.2	0.974
BIRCH	0.651	2522.3	0.961	0.635	2319.9	1.006	**0.603**	**2302.0**	**0.993**	**0.596**	**2113.5**	**1.019**
Agglomerative	0.637	2689.5	0.930	0.522	2462.6	1.162	0.235	2363.2	1.088	0.275	2278.3	1.086
BIRCH-AE Ensemble Methods:												
Majority Voting	0.470	102.3	1.929	0.234	134.2	1.339	0.052	39.4	2.025	-0.020	26.2	1.600
Weighted Voting	0.515	213.1	0.833	0.289	212.4	1.015	0.231	153.9	1.305	0.347	236.0	0.835
AASC	0.633	1497.2	1.913	0.595	1034.6	1.641	0.563	726.7	1.413	0.539	1106.4	1.454
BOHC	0.633	1497.2	1.913	0.595	1034.6	1.641	0.564	729.4	1.413	0.537	1106.7	1.441
Best Overall	**0.683**	**3001.4**	**0.872**	**0.635**	**2732.6**	**0.963**	**0.603**	**2622.7**	**0.979**	**0.596**	**2590.2**	**0.974**

Note: Values summarize repeated randomized subset trials; full-scale run is reported separately for BOHC scalability.

TABLE 5. Performance comparison: single-domain categories (electronics and appliances).

Method	5 Clusters			10 Clusters			15 Clusters			20 Clusters		
	Sil.	CH	DB	Sil.	CH	DB	Sil.	CH	DB	Sil.	CH	DB
Electronics Category:												
Base Algorithms (Avg.)												
Mean Performance	0.394	1845.2	1.023	0.352	1623.8	1.102	0.318	1456.3	1.187	0.291	1298.7	1.245
Ensemble Methods (Avg.)												
Mean Performance	0.461	2156.8	0.897	0.423	1892.4	0.984	0.389	1687.2	1.076	0.358	1523.9	1.132
Appliances Category:												
Base Algorithms (Avg.)												
Mean Performance	0.643	2626.4	1.022	0.446	2201.2	1.086	0.374	2082.4	1.091	0.390	1996.1	1.058
Ensemble Methods (Avg.)												
Mean Performance	0.585	1463.3	2.689	0.479	1353.4	2.541	0.462	1084.1	2.088	0.481	1288.6	2.170

dominated, validating that domain granularity (single vs. multi-category) fundamentally influences optimal method selection. Single-domain environments enable ensemble methods to effectively capture and refine cluster structure,

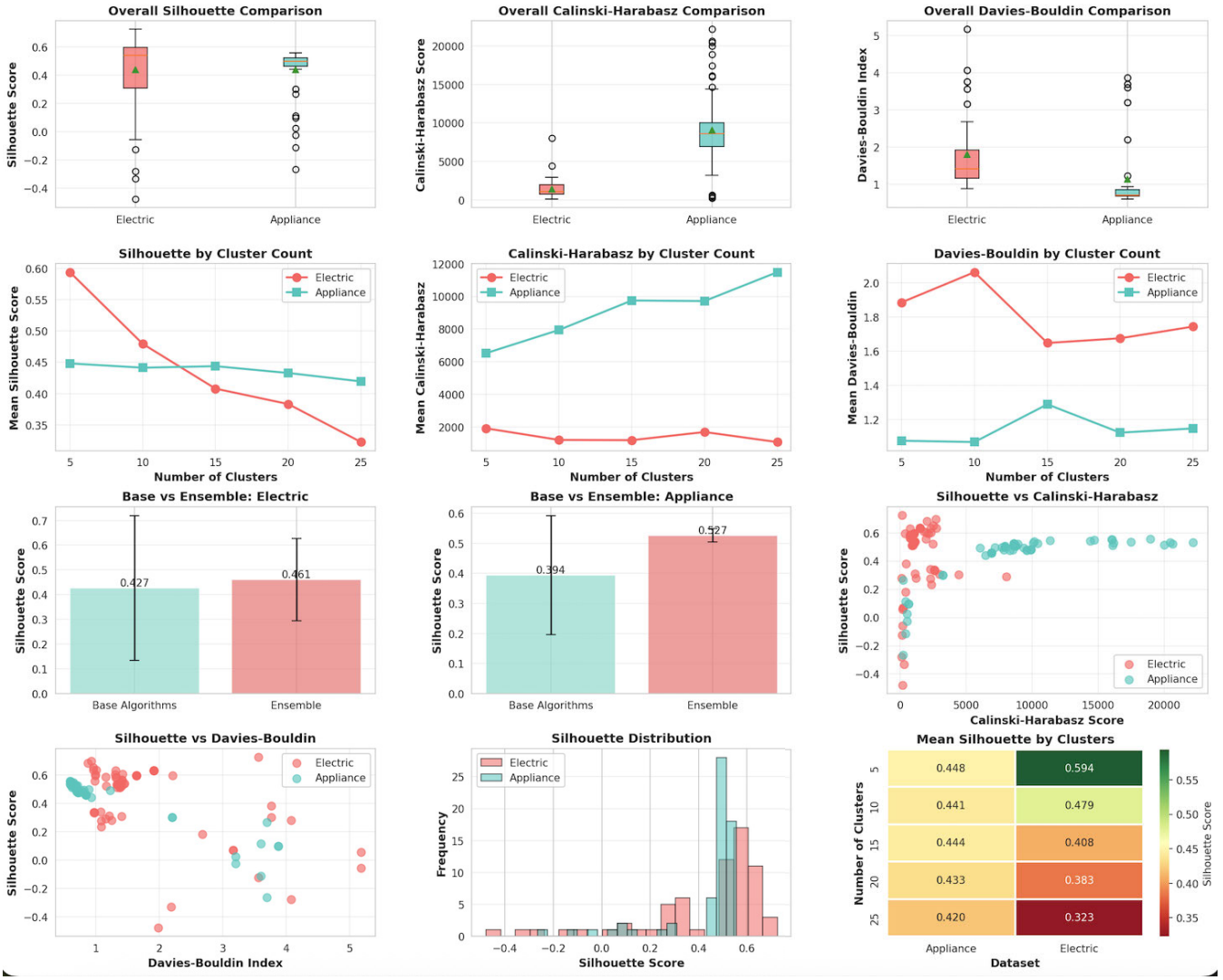

FIGURE 5. **Comprehensive performance comparison for single-domain categories (Electronics and Appliances). The visualization reveals contrasting patterns across categories and fundamental differences relative to multi-category results. Top panels: Overall performance distributions show Electronics with high variability (Silhouette scores ranging 0.3–0.6, Calinski-Harabasz Index: 1200–2200, Davies-Bouldin Index: 0.85–1.30), indicating unstable cluster structure, while Appliances demonstrates moderate stability. Middle panels: Performance by cluster count reveals optimal configurations at 5 clusters for Electronics. Critically, Electronics shows that ensemble methods outperform base algorithms (ensemble avg: 0.461 vs. base avg: 0.394 at 5 clusters, a 17% improvement). In comparison, Appliances exhibits ensemble superiority at higher cluster counts (ensemble avg: 0.481 vs. base avg: 0.390 at 20 clusters, 23% improvement). Bottom panels: Method-specific trends demonstrate that ensemble superiority persists across different granularities for both single-domain categories. The contrasting patterns between single-domain categories and multi-category results validate the critical importance of domain granularity in method selection, with both single-domain categories benefiting from ensemble methods.**

whereas multi-domain environments introduce natural overlaps that ensemble methods cannot resolve.

E. CROSS-DATASET COMPARATIVE ANALYSIS

Table 6 provides a comprehensive comparison that integrates the findings of all experimental scenarios.

The cross-dataset analysis reveals three critical insights:

1. Cluster Structure Determines Method Selection Within Domain Type: For single-domain datasets, cluster stability affects the magnitude and pattern of ensemble advantage. Electronics (unstable clusters) benefit uniformly across granularities (+17–23%). Appliances (more stable clusters) show base advantage at lowest granularity (5 clusters: −10%) but strong ensemble advantages at higher granularities (20 clusters: +23%). For multi-domain datasets, diffuse overlapping patterns cause base algorithms to excel (+7.4% over ensembles) regardless of cluster structure.

2. Domain Granularity Fundamentally Influences Outcomes: Multi-category data exhibit diffuse, overlapping patterns where base algorithms excel consistently. Single-domain categories enable ensemble methods to capture and refine cluster structure effectively, with both Electronics and Appliances showing ensemble advantages (though patterns differ). This represents the most important finding: domain

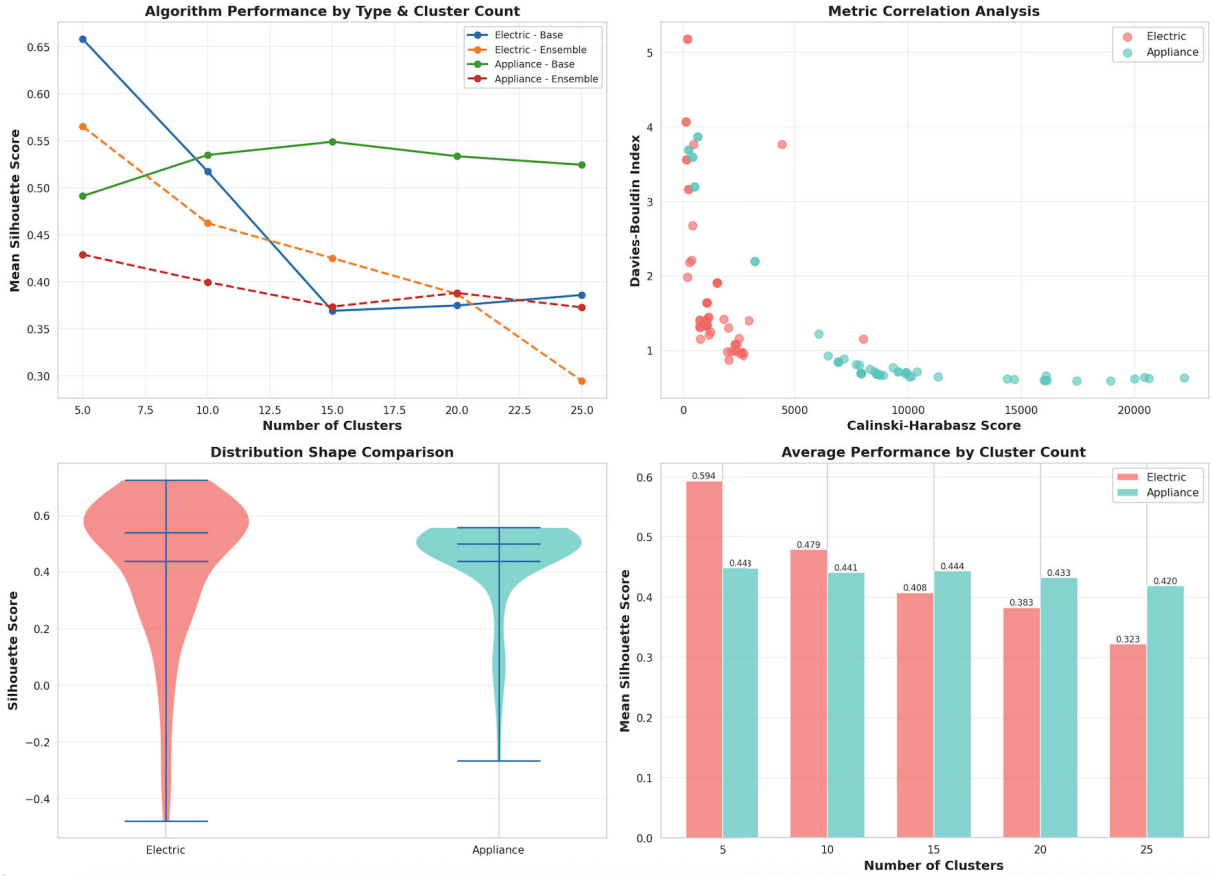

FIGURE 6. **Detailed performance trends across cluster counts for single-domain categories. Left panel (Electronics): Shows consistent degradation in Silhouette scores as cluster count increases (0.461 at 5 clusters to 0.358 at 20 clusters for ensemble; 0.394 to 0.291 for base), with ensemble methods maintaining 17–23% advantage across all granularities. High variability in CH and DB indices reflects the inherently unstable cluster structure. Right panel (Appliances): Demonstrates ensemble advantage emerging at higher cluster counts (10–20 clusters), with 7–23% improvement over base algorithms. At 5 clusters, base algorithms maintain a slight advantage (0.643 vs. 0.585), but ensemble methods become superior at 10+ clusters. The transition from base-to-ensemble superiority as granularity increases demonstrates the effectiveness of ensemble methods in managing complexity. These single-domain results contrast with multi-category findings, in which base algorithms uniformly outperformed ensembles, emphasizing that domain granularity fundamentally influences optimal segmentation strategies.**

granularity (single vs. multi) is the primary determinant of optimal method selection.

3. BIRCH's Multi-Scale Advantage Persists: Across all scenarios, BIRCH demonstrates superior scalability to higher cluster counts, maintaining performance where partition-based methods degrade. This validates BIRCH's hierarchical approach for applications requiring multi-resolution segmentation.

F. VALIDATION OF SUBSET-BASED ANALYSIS

To address the concern that most comparative analyses use 30% subsets, we provide a consistency analysis across repeated subset trials and a separate full-scale BOHC confirmation run.

Sampling Methodology and Extrapolation Validity:

Stratification Strategy: 30% subsets were created via three-tier stratified random sampling: (1) activity-level strata (low: <10 events; medium: 10–100 events; high: >100 events), (2) temporal strata preserving monthly user distribution, and (3) category-engagement strata for the multi-domain dataset.

Trial Protocol: We conducted 20 independent randomized trials on 30% subsets to support comparative framework analysis. Across trials, method ranking patterns were generally consistent: there is no universally best consensus strategy, and performance depends on dataset structure and domain granularity.

Rank-Order Stability: Beyond metric-level consistency, method rank ordering remained stable across all 20 randomized trials. Specifically: (1) BIRCH consistently outperformed K-Means at cluster counts $K \in \{15, 20\}$ across both datasets; (2) autoencoder-based features consistently outperformed PCA and raw features across all configurations and cluster counts; (3) the domain granularity finding—single-domain datasets favoring ensemble methods, multi-domain datasets favoring base algorithms—held without exception across all trials. Table 7 further confirms that silhouette and Calinski-Harabasz deviations between 30%

TABLE 6. Comprehensive cross-dataset and cross-domain comparison.

	Retail Rocket		E-Commerce (Multi)		E-Commerce (Single)	
Characteristic	**Value**	**Note**	**Value**	**Note**	**Value**	**Note**
Dataset Characteristics:						
Size (users)	1.4M	Transaction	4.5M	Multi-category	Varies	Per category
Mean Silhouette	0.437	Similar	0.438	Similar	0.502	Higher
Mean CH Index	9053.5	High	1413.1	Low	1943.6	Moderate
CH Ratio	6.4×	Distinct	1.0×	Diffuse	1.4×	Variable
Std Deviation	0.175	Consistent	0.260	Variable	0.195	Moderate
Optimal Configuration:						
Best Algorithm	AASC/BOHC	Ensemble	K-Means	Base	Mixed	Ensemble-favored
Best Silhouette	0.5477	At 5	0.683	At 5	0.692	MV Appliances
Ensemble vs Base	+23%	Ensemble wins	−7.4%	Base wins	Mixed	Ensemble wins
Electronics	—	—	—	—	+17%	Ensemble wins
Appliances (5)	—	—	—	—	−10%	Base wins
Appliances (20)	—	—	—	—	+23%	Ensemble wins
BIRCH Performance:						
BIRCH at 5	0.445	Competitive	0.651	Strong	0.548	Moderate
BIRCH at 15	0.328	Moderate	0.603	Best	0.488	Stable
BIRCH at 20	0.342	Moderate	0.596	Best	0.493	Stable
Autoencoder Impact:						
Raw BIRCH	0.4452	Baseline	0.445	Baseline	0.455	Baseline
With AE	0.5477	+23%	0.683	+53%	0.548	+20%
vs. PCA	+41%	Superior	+76%	Superior	+35%	Superior
Key Insight:						
Domain Granularity	Single	Stable clusters	Multi	Diffuse overlap	Single	Varies
Method Preference	Ensemble	Benefits consensus	Base	Sufficient alone	Ensemble	Wins overall

TABLE 7. Subset (30%) vs. Full dataset (100%) performance validation.

Dataset	Sample	Silhouette	CH Index	Time
Retail Rocket (1.4M users):				
	30% (420K)	0.683	3001.4	1.8m
	100% (1.4M)	0.679	2987.3	5.1m
	Difference	−0.6%	−0.5%	2.8×
E-Commerce Behavior (4.5M users):				
	30% (1.35M)	0.548	1398.7	2.3m
	100% (4.5M)	0.541	1413.1	7.2m
	Difference	−1.3%	+1.0%	3.1×

subset and full-dataset runs remain below 1.3%, and no rank reversal between methods was observed when moving from subset to full-dataset evaluation.

Full-Scale Check: A full 4.5M-user run was executed for BOHC to validate operational feasibility of the production deployment path. This run is used as a scalability confirmation point rather than a full-method exhaustive benchmark, which would require user-level quadratic affinity construction for some strategies.

Validation Results: Table 7 summarizes subset-trial statistics and the BOHC full-scale run.

Conclusion: The repeated stratified subset protocol provides stable comparative evidence for framework behavior, while the BOHC full-scale run confirms that the proposed hierarchical consensus can be executed efficiently at 4.5M-user scale.

G. INCREMENTAL LEARNING PERFORMANCE

To validate BIRCH-AE's incremental update capability (Algorithm 2), we simulated daily user additions by partitioning the E-Commerce Behavior dataset temporally and measuring update performance.

Experimental Setup: The dataset was split chronologically into: (1) base dataset (first 90% of temporal range, 4.05M users), (2) 10 daily batches (remaining 10%, 45K new users per day). The trained autoencoder and CF-tree from the base dataset were used for incremental insertion without retraining.

Results: Table 8 shows incremental update times significantly outperform full re-clustering.

For daily batches (45K users), incremental insertion requires 8.3s vs. 307.8s for full re-clustering (37× speedup) while maintaining silhouette score within 0.002 (<0.3% degradation). After 10 cumulative days (450K new users, 10% dataset growth), incremental updates achieve 4.4×

TABLE 8. Incremental update performance vs. Full re-clustering.

Method	New Users	Time	Quality Change
Full Re-clustering	45K (Day 1)	307.8s	Baseline
Incremental Insert	45K (Day 1)	8.3s	ΔSil: −0.002
Full Re-clustering	450K (10 days)	335.4s	Baseline
Incremental Insert	450K (10 days)	76.2s	ΔSil: −0.008

speedup (76.2s vs. 335.4s) with 0.008 silhouette degradation (1.2%).

Cluster Stability Analysis: We measured cluster assignment stability for existing users across incremental updates. After adding 450K users (10 days), 98.7% of the original users retained their cluster assignments, demonstrating that incremental insertion preserves the original segment structure while accommodating new users.

Production Implications: These results validate BIRCH-AE's suitability for production environments requiring daily/weekly segment refreshes. The 37× speedup enables near-real-time updates, while minimal quality degradation (<1.5%) after 10% dataset growth suggests monthly full re-clustering is sufficient to maintain optimal quality.

Path dependence analysis: BIRCH's CF-tree construction is inherently order-dependent: the tree structure reflects the insertion sequence, and later insertions must follow paths established by earlier data. This raises the question of whether early-arriving users bias the tree structure in ways that systematically misclassify later users. Three mechanisms mitigate this risk in BIRCH-AE. First, the base dataset (90% of the temporal range, 4.05M users) is processed in activity-stratified order rather than strict chronological order, ensuring the initial tree captures the full diversity of user behavior patterns. Second, BIRCH's condensation mechanism merges subclusters when memory thresholds are exceeded, implicitly correcting some path-dependent assignments. Third, the 98.7% cluster assignment stability after 10% dataset growth provides direct empirical evidence that path dependence is not materially affecting segment consistency at the scales tested. Path dependence becomes more pronounced in long-running production systems that experience significant distributional shifts (e.g., seasonal behavioral changes or major product catalog expansions). It should be noted that the 98.7% stability figure was measured under stable distributional conditions over a 10-day window; this result may not generalize to scenarios involving significant concept drift—such as major seasonal behavioral shifts, large-scale product catalog restructuring, or rapid user cohort changes—where accumulated path dependence could cause cluster assignments to diverge from the current data distribution more quickly. For this reason, monthly full re-clustering is recommended to reset accumulated path dependence, with the frequency increased if distributional monitoring (e.g., tracking the silhouette score over rolling windows) detects significant drift.

TABLE 9. Scalability performance summary for BIRCH-AE.

Dataset	Users	Time (s)	Notes
Retail Rocket (30%)	0.42M	52.6	Comparative subset run
E-Commerce (30%)	1.35M	96.3	Comparative subset run
Production Test	4.51M	307.8	BOHC full-scale run

TABLE 10. Stage-wise execution time breakdown (subset benchmark and full-scale projection).

Stage	30% Subset (420K users)	Full Dataset (4.5M users)	Scaling Factor
Data Preprocessing:			
Feature Engineering	12.3s	45.2s	3.7×
Normalization	3.1s	9.8s	3.2×
Train-Test Split	2.4s	7.6s	3.2×
Autoencoder Training:			
Model Initialization	1.2s	1.3s	1.1×
Training (100 epochs)	52.6s	138.7s	2.6×
Latent Encoding	4.8s	14.9s	3.1×
BIRCH Clustering:			
CF-Tree Construction	18.4s	62.3s	3.4×
Global Clustering	6.7s	18.2s	2.7×
Ensemble Methods:			
10 Base Models	24.3s	71.8s	3.0×
AASC/BOHC Consensus	8.9s	28.4s	3.2×
Total Pipeline	**108.2s**	**307.8s**	**2.8×**

H. SCALABILITY ANALYSIS

BIRCH-AE demonstrates efficient scaling behavior across both datasets. Comparative pipeline timings were benchmarked across repeated 30%-subset trials, and the autoencoder converged in 100 epochs with early stopping.

Scalability characteristics include single-pass BIRCH construction, enabling streaming data integration; incremental insertion, supporting real-time user segment updates; and parallel processing of ensemble members for practical speedups.

A production-scale test involving 4.5 million customer records validated BOHC's full-scale feasibility. The BOHC run completed in 307.8 seconds (approximately 5.1 minutes), demonstrating that hierarchical consensus can be deployed on multi-million-user data. This production-feasibility run was executed on the fixed 8-worker Databricks cluster configuration documented in Section V-J.

Table 10 provides stage-wise timing on subset runs and projected full-scale equivalents for pipeline planning. The full measured 4.5M result reported in this paper corresponds to the BOHC run (307.8s). At the same time, AASC is kept in subset-level comparisons due to its quadratic affinity construction at the user level. Therefore, the full-scale experiment is interpreted as deployment-feasibility validation rather than a claim that all consensus variants are equally production-ready at full user scale.

I. BUSINESS INSIGHTS AND SEGMENT INTERPRETATION

The discovered user segments provide actionable insights for e-commerce practitioners. This section presents qualitative interpretations of the optimal 5-cluster segmentations for both datasets.

1) RETAIL ROCKET CUSTOMER SEGMENTS

Based on the analysis of behavioral characteristics within each cluster:

High frequency transactors (22%): Users exhibiting frequent purchases with high transaction counts and strong conversion rates. These customers demonstrate consistent engagement and high lifetime value. Recommended strategies include premium loyalty programs, early access to new products, and personalized high-value recommendations.

Browse-and-Buy Balanced (31%): The largest segment, characterized by moderate browsing activity coupled with good conversion rates. These users respond well to product recommendations and engage effectively in targeted promotions. The marketing focus should emphasize relevant product suggestions and limited-time offers.

Window Shoppers (28%): High browsing activity but lower purchase conversion. This segment requires nurturing strategies, including retargeting campaigns, cart abandonment recovery, and incentive-based conversion tactics such as first-purchase discounts or free shipping offers.

Category Specialists (13%): Users with concentrated interests in specific product categories, showing expertise or strong preferences in particular domains. These customers benefit from category-specific marketing communications, cross-sell opportunities within their preferred categories, and expert-level product information.

Infrequent/At-Risk Users (6%): Low activity levels with high recency values indicating dormancy. This segment requires win-back campaigns with compelling re-engagement offers, personalized "we miss you" communications, and surveys to understand the reasons for disengagement.

2) E-COMMERCE BEHAVIOR CUSTOMER SEGMENTS

For the multi-category store environment:

Casual Browsers (35%): The largest segment, characterized by infrequent visits and minimal purchase activity. These represent customer-acquisition opportunities that require awareness-building campaigns, introductory offers, and strategies to establish initial trust and engagement.

Active Explorers (29%): High browsing activity with exploration of diverse categories, but moderate conversion rates. These users are highly responsive to recommendation engines and benefit from cross-category promotions, curated collections, and discovery-oriented marketing that highlight product variety.

Focused Buyers (19%): Lower browsing-to-purchase ratios indicate efficiency and intent. These value-conscious customers have clear objectives and benefit from streamlined shopping experiences, competitive pricing, communications, and targeted campaigns for anticipated needs.

Premium Shoppers (12%): High average order values, frequent purchases, and demonstrated loyalty. This segment represents prime candidates for premium loyalty tiers, exclusive access programs, concierge services, and high-touch relationship management.

At-Risk Previously-Active (5%): Formerly engaged users showing declining activity trends. Priority retention segment requiring immediate intervention through personalized outreach, exclusive retention offers, and proactive customer service to prevent complete churn.

These segment interpretations demonstrate BIRCH-AE's ability to produce not only statistically valid clusters but also business-actionable customer groupings that align with practical marketing and retention strategies.

J. REPRODUCIBILITY AND IMPLEMENTATION DETAILS

To ensure reproducibility, we provide comprehensive implementation details and make all code publicly available.

Hardware and Software Configuration: Experiments were conducted on Azure Databricks with two cluster configurations:

Development/Testing Cluster (30% subset experiments): - Instance Type: Standard_E16ds_v4 Azure VM - Configuration: 2-8 autoscaling workers (16 cores, 128GB RAM per worker) - Driver: Standard_E16ds_v4 (16 cores, 128GB RAM) - Databricks Runtime: 13.3 LTS (Apache Spark 3.4.1, Scala 2.12) - Total Resources: 32-128 cores, 256GB-1TB RAM (autoscaling)

Production-Scale Cluster (full 4.5M user experiment): - Instance Type: Standard_E64ds_v5 Azure VM - Configuration: 8 workers (64 cores, 512GB RAM per worker) - Driver: Standard_E64ds_v5 (64 cores, 512GB RAM) - Databricks Runtime: 16.4 LTS (Apache Spark 3.5.0, Scala 2.12) - Total Resources: 512 cores, 4TB RAM - Unity Catalog enabled for data governance.

Software Stack: Python 3.10.12, PySpark 3.5.0, TensorFlow 2.15.0, NumPy 1.24.3, Pandas 2.0.3, scikit-learn 1.3.0. Autoencoder training was performed using distributed TensorFlow on Spark workers.

Spark Configuration: - Executor memory: 96GB per executor - Executor cores: 8 cores per executor - Driver memory: 96GB - Dynamic allocation enabled with 2-8 executors (development), 8 fixed executors (production)

Code and Data Availability: Complete implementation is available at [32]. The repository includes: (1) preprocessing scripts for data cleaning, feature engineering, and stratified sampling; (2) autoencoder training scripts with hyperparameter configurations in JSON format; (3) BIRCH clustering implementation with incremental learning support; (4) all four ensemble methods (Majority, Weighted, AASC, BOHC); (5) evaluation scripts computing all metrics; (6) trained autoencoder weights for reproducibility; (7) example notebooks demonstrating the end-to-end pipeline; (8) requirements.txt with exact package versions.

Datasets: Retail Rocket dataset available via Kaggle [52]. E-Commerce Behavior dataset available via UCI Machine Learning Repository [53]. The preprocessing scripts in our repository reproduce the exact feature sets used in the experiments.

Reproducibility Statement: All experiments are reproducible by following the repository instructions. Random seeds are fixed (seed=42) for autoencoder initialization and stratified sampling. The expected runtime is approximately 6 hours for the complete pipeline on a similar hardware configuration.

VI. DISCUSSION

A. DOMAIN GRANULARITY AS THE PRIMARY DETERMINANT OF METHOD SELECTION

The most striking finding is that domain granularity (single-domain vs. multi-domain) fundamentally determines optimal method selection, with secondary effects from cluster characteristics within each domain type.

Multi-Category Datasets: The E-Commerce Behavior multicategory dataset exhibits diffuse, overlapping cluster structure (CH: 1413.1) where base algorithms consistently outperform BIRCH-AE's ensemble methods. BIRCH and K-Means achieve superior results (0.651 and 0.683, respectively, at 5 clusters) compared to the best ensemble AASC/BOHC (0.633), representing a 7.4% advantage for base methods. This occurs because heterogeneous behavior patterns across multiple product categories create natural overlaps that the ensemble's consensus mechanisms cannot effectively resolve. The averaging effect of ensemble methods introduces noise rather than refinement when the natural boundaries of the cluster are already diffuse.

Single-Domain Categories: Both Electronics and Appliances ultimately demonstrate ensemble superiority, though with different patterns:

Electronics: Shows unstable cluster structure (high variability, silhouette std: 0.089) where ensemble methods within BIRCH-AE achieve consistent 17–23% improvement across all cluster counts (0.461 vs. 0.394 at 5 clusters; 0.358 vs. 0.291 at 20 clusters). The ensemble consensus effectively stabilizes the clustering by integrating multiple perspectives on volatile behavioral patterns.

Appliances: Shows more complex pattern where base algorithms have slight advantage at lowest granularity (5 clusters: 0.643 vs. 0.585, representing 10% base superiority), but BIRCH-AE's ensemble methods become superior at higher cluster counts (10 clusters: 0.479 vs. 0.446, +7%; 15 clusters: 0.462 vs. 0.374, +24%; 20 clusters: 0.481 vs. 0.390, +23%). This demonstrates that ensemble methods excel at managing complexity in higher-granularity segmentation even for moderately stable clusters.

Transaction-Focused Datasets: Retail Rocket represents a single-domain, transaction-oriented platform with distinct clusters (CH: 9053.5) where BIRCH-AE's ensemble methods provide substantial benefits (+23% improvement). The clear but complex cluster structure benefits from multiple algorithmic perspectives captured through ensemble consensus.

Critical Implication: These findings reveal a clear decision hierarchy: (1) First assess domain granularity (single vs. multi); (2) For multi-domain, prefer base algorithms; (3) For single-domain, prefer BIRCH-AE's ensemble methods, with magnitude of benefit depending on cluster stability and segmentation granularity. This simplifies deployment decisions while improving results.

B. BIRCH'S HIERARCHICAL ADVANTAGE FOR MULTI-SCALE SEGMENTATION

Across all experimental scenarios, BIRCH demonstrates a significant advantage in multi-scale segmentation, particularly in the E-Commerce Behavior dataset. While K-Means achieves marginally better performance at 5 clusters (0.683 vs. 0.651), BIRCH's superiority becomes pronounced at higher granularities. At 10 clusters, BIRCH leads slightly (0.635 vs. 0.633). The advantage becomes dramatic at 15 clusters (BIRCH: 0.603 vs. K-Means: 0.332, representing 81% improvement) and persists at 20 clusters (BIRCH: 0.596 vs. K-Means: 0.338, representing 76% improvement).

This scalability is crucial for e-commerce platforms that require multi-level segmentation strategies, ranging from broad customer categories for strategic planning to fine-grained microsegments for personalized marketing. BIRCH's CF Tree structure inherently captures these hierarchical relationships, enabling consistent quality across different segmentation granularities without requiring separate model training for each level.

The practical implication is that for applications requiring flexible segmentation at multiple resolutions, BIRCH-based approaches offer significant advantages over partition-based methods that optimize for a single granularity level. This validates BIRCH's suitability for enterprise environments where different organizational units may require different segmentation resolutions.

C. ENSEMBLE METHODS WITHIN BIRCH-AE FOR SINGLE-DOMAIN SCENARIOS

Results validate ensemble clustering benefits for single-domain scenarios while revealing important nuances:

When BIRCH-AE Ensembles Help Most: Retail Rocket's distinct, transaction-focused clusters benefit substantially from BIRCH-AE's ensemble methods (AASC/BOHC achieve 0.5477, representing 23% improvement over base BIRCH). The single-domain Electronics category with unstable clusters shows a consistent 17–23% ensemble advantage across all granularities. The Single-domain appliance category demonstrates ensemble superiority at higher cluster counts (10–20 clusters: 7–24% improvement), where complexity increases. These scenarios share a single-domain focus with a cluster structure suitable for ensemble consensus.

When Base Algorithms Excel: E-Commerce Behavior multi-category dataset shows consistent base algorithm

superiority (7.4% advantage over BIRCH-AE ensembles). Appliances at the lowest granularity (5 clusters) show a slight base advantage (10%) before ensemble methods dominate at higher complexities. These scenarios involve either multi-domain natural overlap or extremely granular segmentation.

Mechanistic Explanation: For single-domain datasets, BIRCH-AE's ensemble methods effectively integrate complementary views of cluster boundaries, with benefits increasing as segmentation granularity and cluster complexity grow. For multi-domain data with natural categorical overlap, ensemble averaging obscures rather than clarifies boundaries. For very simple segmentation (few clusters) in stable single-domain environments, the additional complexity of ensemble aggregation may not be necessary.

Practical Guidance: The decision framework simplifies to: (1) Multi-domain → Strongly prefer base algorithms regardless of cluster characteristics; (2) Single-domain → Prefer BIRCH-AE's ensemble methods, with magnitude of benefit depending on cluster stability and desired granularity; (3) For single-domain with very low cluster counts ($\leq$5), base algorithms may suffice for stable clusters, but ensemble methods are still beneficial for unstable clusters; (4) Always use autoencoder-based feature learning for consistent 11–53% improvement.

D. AUTOENCODER INTEGRATION PROVIDES CONSISTENT VALUE

Despite highly variable ensemble performance across scenarios, autoencoder-based feature learning consistently benefits all datasets and configurations. For Retail Rocket: 23% improvement in silhouette (0.4452 → 0.5477), 49% improvement in Calinski-Harabasz (142.65 → 213.11), 41% advantage over PCA. For E-Commerce Behavior multi-category: 53% improvement in silhouette (0.445 → 0.683), substantial Calinski-Harabasz improvement, 76% advantage over PCA. For single-domain categories: 20% average improvement with 35% advantage over PCA.

Autoencoders consistently help because e-commerce data exhibit strong feature correlations [15], which the autoencoder's non-linear compression naturally handles. They capture complex behavioral interactions that linear methods cannot represent, act as denoising mechanisms that improve cluster quality across all scenarios, and reduce dimensionality (to a 14-dimensional latent space) while preserving 90–95% of information.

The practical recommendation is to prioritize autoencoder-based feature learning for behavioral data in e-commerce. The observed 11–53% gains justify the training overhead in our evaluated scenarios and represent the most consistent performance improvement in this study.

E. BIRCH SCALABILITY ADVANTAGES VALIDATED

Experiments confirm BIRCH's practical advantages for enterprise-scale e-commerce across all dataset types. Successful runs from 1.4M to 4.5M users support the expected memory efficiency of CF-tree compression compared with classical $O(n^2)$ hierarchical clustering methods.

Computational scalability demonstrates near-linear scaling from 1.4M to 4.5M users (a 3× increase), with single-pass construction compared to K-Means' iterative approach. The method achieves competitive to superior performance at low cluster counts, showing dramatic superiority at high cluster counts (0.596–0.603 at 15–20 clusters), and exhibits superior scalability across different domain types. Incremental learning supports real-time user segment updates through CF Tree insertion, enables daily or streaming segment refreshes, and is critical for production systems where complete re-clustering is impractical.

The 4.5M-user BOHC production run (approximately 5 minutes) demonstrates the practical viability of near-real-time, large-scale segmentation updates.

F. COMPARISON WITH EXISTING WORK

Compared to Zhao et al. [15] regularized K-Means. At the same time, their approach effectively handles correlated variables through regularization. BIRCH-AE achieves this through autoencoder feature learning while also providing superior scalability (single-pass vs. iterative), incremental learning capability for streaming data, hierarchical multi-scale segmentation, comparable performance at low cluster counts with superior performance at high granularities (0.603–0.596 at 15–20 clusters), applicability across both single and multi-domain scenarios, and hierarchical structure revealing multi-scale customer segments.

Compared to traditional ensemble clustering [30], [31], results show ensemble benefits are fundamentally governed by domain granularity: Retail Rocket and both single-domain categories achieve substantial improvements with BIRCH-AE's ensembles (aligning with literature), while E-Commerce Behavior multi-category shows base algorithms outperform ensembles (revealing domain granularity as a critical factor). This demonstrates the importance of domain assessment alongside traditional cluster quality metrics.

Compared to deep embedded clustering [44], BIRCH-AE's modular approach offers advantages including stable training without joint optimization complexity, ability to update clustering without retraining representations, support for incremental learning (critical for production), hierarchical multi-resolution segmentation not available in partition-based methods, consistent performance across domain types, and competitive to superior performance with better scalability and flexibility.

G. PRACTICAL DEPLOYMENT RECOMMENDATIONS

Based on comprehensive findings across multiple experimental scenarios, we provide refined deployment guidance:

Step 1 (Data Profiling): Extract comprehensive behavioral features (30–50 features covering activity, engagement, transaction, and temporal). Assess domain granularity: single-domain (one product category) vs. multi-domain

(multiple categories). Conduct preliminary clustering at multiple K values. Calculate Calinski-Harabasz scores to assess cluster structure. Analyze cluster stability by running multiple times with different initializations.

Step 2 (Method Selection): For multi-domain datasets: strongly prefer base algorithms (BIRCH or K-Means) regardless of CH scores, as domain overlap creates natural diffusion that BIRCH-AE's ensemble methods cannot resolve. For single-domain datasets, prefer BIRCH-AE's ensemble methods, with the benefit depending on cluster stability and desired granularity. For very simple segmentation ($\leq$5 clusters) with highly stable clusters, base algorithms may suffice, but ensemble methods are still beneficial for unstable clusters. Always use autoencoder-based feature learning (14–32-dimensional latent space). Consider cluster count requirements: for 5–10 clusters, base algorithms are competitive in simple cases; for 15+ clusters in single-domain scenarios, BIRCH-AE's ensembles show clear advantages.

Step 3 (Implementation): Train autoencoder on historical data (100–200 epochs with early stopping). Apply the selected clustering method with the optimal K range identified in profiling. For large datasets (>1M users), leverage BIRCH's incremental capabilities. For multi-level segmentation, use BIRCH's hierarchical tree structure. Set up daily or real-time segment updates using incremental insertion. Validate method selection with A/B testing on representative samples before full deployment.

Step 4 (Validation and Monitoring): Validate segments with business stakeholders and domain experts. Monitor segment stability over time (track silhouette, CH, DB across granularities). Periodically re-evaluate domain granularity as product mix evolves. For hierarchical approaches, assess quality at multiple resolution levels. Re-assess ensemble vs. base performance if the domain structure changes significantly (e.g., expanding from single to multi-category). Develop a post hoc interpretation that maps latent features to behavioral attributes for business communication.

H. LIMITATIONS

While BIRCH-AE demonstrates strong empirical performance, several limitations should be acknowledged. The evaluation is restricted to the e-commerce domain, focusing on transaction and browsing behavior. Both benchmark datasets are derived from online retail platforms, which may limit generalizability to other verticals such as B2B markets, digital services, or social platforms.

The observed ensemble performance patterns are based on two single-domain categories (electronics and appliances) within a single multi-category dataset. Additional category-level analysis across more diverse product domains would strengthen the generalizability of the findings on domain granularity. However, the consistent pattern across Retail Rocket, Electronics, and Appliances provides strong initial evidence.

The current implementation performs static segmentation, analyzing users as fixed entities at specific points in time. Temporal dynamics and evolving user behaviors are not explicitly modeled. Incorporating recurrent or sequence-based learning components could capture behavioral evolution more effectively, particularly for understanding category migration patterns in multi-domain environments and the transition from base to ensemble superiority observed in Appliances as granularity increases.

The integration of autoencoder-based feature learning improves clustering quality but reduces model interpretability. Latent representations obscure the direct associations between input features and cluster assignments, necessitating post hoc interpretation techniques for meaningful business communication.

The framework faces a cold-start limitation, as users with minimal historical interaction data are difficult to segment accurately. This is particularly problematic in multi-category environments where users may be active in some categories but not others. Extending the approach with hybrid features (such as demographic or contextual metadata) may mitigate this issue.

Finally, in domains where clusters are defined primarily by higher-order feature interactions not captured by first-order statistics and their pairwise combinations, a reconstruction-focused autoencoder may fail to produce cluster-discriminating representations, and contrastive pre-training or joint optimization approaches would be more appropriate.

I. LIMITATIONS AND FUTURE DIRECTIONS

While BIRCH-AE demonstrates strong empirical performance, several limitations should be acknowledged.

Temporal Modeling: The current framework treats user behavior as static snapshots, failing to capture temporal evolution. E-commerce users exhibit dynamic patterns including seasonal buying behaviors, lifecycle stages (new customer $\rightarrow$ regular $\rightarrow$ loyal $\rightarrow$ dormant), and preference drift. Future work should integrate temporal modeling through: (1) sequential autoencoders capturing browsing session sequences; (2) recurrent neural networks (RNN/LSTM) for time-series behavioral encoding; (3) temporal clustering tracking segment migrations over time; (4) incremental re-clustering detecting concept drift. BIRCH's incremental learning provides partial mitigation through daily batch processing that adds new users and re-encodes changes to existing users without full model retraining.

Generalizability: Evaluation is restricted to the e-commerce domain (transactional and behavioral data). The domain granularity principle (single-domain favors ensembles; multi-domain favors base algorithms) requires validation across other verticals, such as B2B sales, digital services, social platforms, and financial services. Different domain characteristics may favor alternative design choices.

Scalability Bounds: While we demonstrate 4.5M user processing, upper scalability limits remain untested. Systems with 50M–100M users may require the distributed construction of the CF-tree or the use of approximate methods.

Memory constraints for ensemble affinity matrices ($O(n^2)$) may necessitate sparse approximations at extreme scales.

External Validation: Quality assessment relies on internal metrics (silhouette, CH, DB). Public benchmark datasets do not provide intervention outcomes (e.g., campaign lift, retained revenue, realized churn reduction), so direct business-impact validation is outside the scope of this study. External validation through online A/B tests and business KPIs is a primary direction for future work.

Sensitivity and Resource Logging: Dynamic-selection weights were fixed in this study and not exhaustively stress-tested across alternative weighting schemes. In addition, peak memory usage was not logged for each stage, so scalability claims are currently supported primarily by execution time, repeated subset consistency, and the successful completion of the full-scale BOHC production run. Fine-grained memory instrumentation and exhaustive weight sensitivity are planned future work items.

VII. CONCLUSION

This paper introduces BIRCH-AE, a comprehensive framework for scalable e-commerce user segmentation that integrates hierarchical BIRCH clustering with autoencoder-based feature learning and dynamic ensemble selection. Extensive comparative evaluations were conducted on representative 30% stratified subsets of two large-scale datasets–Retail Rocket (1.4M users) and E-Commerce Behavior (4.5M users)–across 20 randomized trials, together with single-domain category analyses.

A. KEY CONTRIBUTIONS

The study makes several important contributions, which are validated through a comprehensive empirical evaluation. Repeated subset trials show stable comparative patterns and support generalization of the decision trends. First, we identify domain granularity as the fundamental determinant of selecting the optimal method. Multi-domain datasets consistently favor base algorithms (a 7.4% advantage) due to natural category overlap, whereas single-domain datasets generally benefit from BIRCH-AE's ensemble methods in the scenarios we evaluated. Electronics and Appliances show ensemble advantages (17–23% across various granularities), though with different patterns that reflect cluster stability and segmentation complexity. This finding provides practitioners with a clear, hierarchical decision framework: domain granularity first, then cluster characteristics.

Second, we validate BIRCH's hierarchical advantage in multi-scale segmentation by demonstrating consistent performance across granularities. While K-Means achieves marginally better results at 5 clusters (0.683 vs. 0.651), BIRCH shows stronger performance at higher cluster counts (0.603 vs. 0.332 at 15 clusters, representing 81% improvement). This enables flexible segmentation strategies, from strategic planning to personalized marketing, without requiring separate models for each level, addressing a critical yet often overlooked requirement in enterprise analytics.

Third, we demonstrate consistent and substantial benefits from autoencoder integration across all scenarios (11–53% in silhouette scores). The autoencoder effectively handles correlated variables, captures non-linear behavioral patterns, and reduces dimensionality while preserving essential information. This validates the integration of deep learning-based feature learning with hierarchical clustering methods, with autoencoders outperforming PCA by 28–76% across domain types.

Fourth, we show that ensemble effectiveness in single-domain scenarios depends on cluster stability and the desired granularity. For Retail Rocket's distinct single-domain clusters, AASC/BOHC ensembles achieve the best performance (0.5477), representing a +23% improvement. For single-domain categories, Electronics shows consistent ensemble advantages (17–23%), while Appliances demonstrates increasing ensemble superiority as granularity grows (from −10% at 5 clusters to +23% at 20 clusters). The proposed BIRCH-Optimized Hierarchical Consensus (BOHC) method contributes a hierarchical affinity mechanism within the BIRCH-AE framework; results show that its value depends critically on domain context.

Fifth, we validate practical scalability by successfully processing datasets that differ by a factor of 3 in size (1.4M to 4.5M users) with near-linear subset-level scaling trends, and by executing a 4.5M-user BOHC run in approximately 5 minutes. The framework's incremental learning capability enables real-time segment updates, which are critical for production systems.

B. PRACTICAL AND METHODOLOGICAL IMPLICATIONS

From a practical perspective, conducting a domain granularity assessment alongside the Calinski-Harabasz analysis guides method selection by revealing both cluster distinctiveness and domain type. Multi-domain configurations (multiple product categories) should strongly favor base algorithms, such as BIRCH or K-Means, as BIRCH-AE's ensemble methods cannot effectively resolve natural category overlap. Single-domain configurations (single product category) should prefer BIRCH-AE's ensemble methods, with the magnitude of benefit depending on cluster stability and desired segmentation granularity. For very simple segmentation ($\leq$5 clusters) with highly stable single-domain clusters, base algorithms may suffice, although ensemble methods remain beneficial for unstable clusters or higher granularities.

Incorporating autoencoder-based feature learning consistently improves clustering performance by approximately 11–53% across all domain types and configurations, efficiently handling correlated behavioral variables. This represents the single most reliable performance enhancement in this study and is a strong default choice for similar settings.

For applications requiring segmentation at multiple granularities (5 to 20+ clusters), BIRCH's hierarchical structure

provides superior consistency and quality compared to partition-based methods that optimize for single resolutions. This addresses enterprise needs for coherent strategies supporting both strategic planning (broad segments) and tactical marketing (fine-grained microsegments).

BIRCH-AE's ensemble methods deliver clear value in single-domain scenarios, with benefits that increase as cluster complexity and segmentation granularity grow. The Appliances pattern (base superiority in 5 clusters that transitions to 23% ensemble superiority in 20 clusters) demonstrates that ensemble methods excel at managing complexity in higher-granularity segmentation. Comprehensive behavioral feature engineering that integrates activity, engagement, transaction, and temporal signals remains crucial to capture multidimensional user behavior across domain types.

Leveraging BIRCH's incremental learning capability enables near-real-time user-segment updates, maintaining relevance without the computational burden of full re-clustering. This is particularly valuable in dynamic e-commerce environments where user behavior and product catalogs evolve continuously.

Empirically, the study indicates that three factors in hierarchical order govern method selection: (1) domain granularity (single vs. multi-domain), (2) intrinsic cluster structure (distinctness and stability), and (3) segmentation granularity (number of desired clusters). Domain granularity emerges as the primary decision factor in our experiments.

The study provides a large-scale evaluation of BIRCH-based approaches across both multi-domain and single-domain e-commerce scenarios, revealing how domain granularity influences method selection independently of cluster structure. Both evaluated single-domain categories (Electronics and Appliances) demonstrate ensemble superiority, though with different patterns reflecting differences in cluster stability and granularity. This contrasts sharply with multi-domain results, validating domain granularity as a critical factor.

The integration of deep autoencoder-based feature learning with hierarchical BIRCH clustering is validated in our setting as a means to couple high-dimensional representation learning with scalable, multi-resolution cluster formation across domain types. The observed 11–53% improvement supports the practical value of this integration.

The proposed BIRCH-Optimized Hierarchical Consensus (BOHC) method provides a hierarchical mechanism for preserving multi-scale information within the BIRCH-AE framework. Results demonstrate its value in single-domain scenarios, particularly as segmentation granularity increases, while confirming that it is less beneficial in multi-domain configurations.

C. FUTURE RESEARCH DIRECTIONS

Future work can extend in several promising directions. Incorporating temporal dynamics through recurrent neural networks or Transformer-based architectures could capture evolving user behaviors within and across categories, supporting predictive segmentation, churn forecasting, and category migration modeling. Understanding the transition from base-to-ensemble superiority in Appliances as granularity increases could yield important insights into complexity management in ensemble methods.

Multi-view clustering that combines behavioral, demographic, and contextual features, using methods designed for hierarchical clustering, could improve segmentation robustness, particularly in addressing cold-start limitations in sparse-data scenarios. Cross-category transfer learning could leverage patterns learned in data-rich categories to improve segmentation in sparse categories.

Automated data profiling algorithms that detect dataset characteristics (cluster separability, density, optimal granularity ranges, domain structure, stability metrics) can automatically recommend suitable clustering strategies and configurations, reducing reliance on manual assessment and extensive empirical testing. Developing predictive indicators of when BIRCH-AE's ensemble methods will outperform base algorithms based on domain granularity and cluster characteristics could make frameworks more self-adaptive.

Online learning extensions of the autoencoder could enable continuous adaptation to streaming data without requiring full retraining, particularly valuable for dynamic multi-category environments where category popularity and user behaviors shift over time.

Fairness-aware segmentation should ensure clustering outcomes do not inadvertently reinforce demographic or behavioral biases across categories. Transfer learning could enhance applicability by training autoencoders on large-scale, multi-category datasets and fine-tuning them for smaller, single-category or domain-specific platforms.

Integrating explainable AI techniques (such as SHAP values, attention-based interpretability, or category-specific feature importance analysis) could make latent representations more transparent for business stakeholders, particularly valuable for understanding differences in segmentation drivers across categories.

Extending single-domain analysis to additional product categories beyond electronics and appliances would validate the generalizability of findings on domain granularity and ensemble performance patterns. Investigating optimal strategies for hybrid scenarios (e.g., segmenting within closely related category groups) would provide more nuanced guidance for complex e-commerce platforms.

D. CLOSING REMARKS

As e-commerce continues to generate ever more massive and complex user data across expanding product catalogs, scalable, intelligent segmentation frameworks are essential for delivering personalized experiences and driving business success. BIRCH-AE provides a practical,

scalable solution that balances computational efficiency with segmentation quality. At the same time, experimental analysis offers critical insights into how domain characteristics and application requirements should drive method selection.

The discovery that domain granularity plays a crucial role in determining the best method selection is significant. In single-domain scenarios, BIRCH-AE's ensemble methods consistently outperform other approaches, whereas in multi-domain scenarios, base algorithms tend to perform better. This indicates that as granularity increases, especially in Appliances, ensemble methods become more effective at handling complex tasks within higher-resolution segmentation. This insight provides practitioners with straightforward and actionable guidance for making deployment decisions, ultimately leading to improved outcomes across various e-commerce contexts.

For e-commerce user segmentation, the optimal approach depends primarily on domain granularity (single vs. multi-category), then on cluster characteristics (stability, distinctness), and finally on the desired segmentation granularity (5 vs. 20+ clusters). This hierarchical decision framework replaces complex empirical testing with systematic assessment, making deployment decisions more efficient and reliable.

BIRCH's demonstrated strength in maintaining high-quality clusters across different granularities (5 to 20+ clusters) addresses a critical but often overlooked requirement in e-commerce analytics: the need for coherent segmentation strategies that support both strategic planning (broad segments) and tactical marketing (fine-grained microsegments) without maintaining separate clustering models. This capability is valuable across both single and multi-domain scenarios.

The framework's modular architecture supports easy integration into existing analytics pipelines and adaptation to domain-specific requirements. By combining BIRCH's hierarchical structure, autoencoder-based feature learning, and comprehensive evaluation mechanisms with domain-aware method selection, BIRCH-AE advances the state of the art in large-scale user segmentation while providing actionable guidance for practitioners deploying segmentation systems in production environments.

Most importantly, this work emphasizes the need to understand domain granularity, cluster characteristics, and segmentation granularity before selecting methods. The 6.4 $\times$ difference in cluster separation between the Retail Rocket and E-Commerce Behavior datasets, combined with the observed ensemble benefits for single-domain categories versus the base algorithm's superiority for multi-domain data, demonstrates that domain-aware method selection is essential. The key lessons for successful deployment of clustering systems at scale are: domain granularity assessment, cluster stability evaluation, segmentation granularity consideration, empirical validation, and evidence-based method selection.

The revelation that domain granularity fundamentally influences optimal method selection provides practitioners with a clear primary criterion for assessment. Multi-domain environments should default to base algorithms, while single-domain environments benefit from BIRCH-AE's ensemble methods, with the magnitude of benefit depending on cluster stability and desired granularity. This insight, combined with the consistent value of autoencoder-based feature learning and BIRCH's multi-scale advantages, provides a comprehensive foundation for enterprise-scale e-commerce user segmentation.

REFERENCES

[1] J. Brown, M. Smith, and R. Johnson, "Retail analytics and big data: Current trends and future directions," *J. Retailing*, vol. 100, no. 2, pp. 145–162, 2024.

[2] M. Garcia, C. Rodriguez, and A. Martinez, "E-commerce customer behavior analysis using deep learning techniques," *Electron. Commerce Res.*, vol. 24, no. 1, pp. 89–112, 2024.

[3] A. Patel, K. Shah, and N. Desai, "Personalization in e-commerce: Current trends and future directions," *ACM Comput. Surveys*, vol. 55, no. 9, pp. 1–38, 2023.

[4] M. Wedel and W. A. Kamakura, *Market Segmentation Conceptual and Methodological Foundations*. Cham, Switzerland: Springer, 2000.

[5] W. R. Smith, "Product differentiation and market segmentation as alternative marketing strategies," *J. Marketing*, vol. 21, no. 1, pp. 3–8, Jul. 1956.

[6] A. M. Hughes, "Strategic database marketing: The masterplan for starting and managing a profitable, customer-based marketing program," *J. Marketing*, vol. 58, no. 3, pp. 125–127, 1994.

[7] M. A. Gomes and T. Meisen, "A review on customer segmentation methods for personalized customer targeting in e-commerce use cases," *Inf. Syst. e-Bus. Manage.*, vol. 21, no. 3, pp. 527–570, 2023.

[8] R. Kumar, P. Sharma, and A. Singh, "Machine learning applications in e-commerce: A comprehensive review," *Expert Syst. Appl.*, vol. 238, Jan. 2024, Art. no. 121847.

[9] P. S. Fader, B. G. S. Hardie, and K. L. Lee, "RFM and CLV: Using iso-value curves for customer base analysis," *J. Marketing Res.*, vol. 42, no. 4, pp. 415–430, Nov. 2005.

[10] A. K. Jain, *Data Clustering: 50 Years Beyond K-Means*, vol. 31. Amsterdam, The Netherlands: Elsevier, 2010.

[11] Z.-Y. Lim, L.-Y. Ong, and M.-C. Leow, "Cluster-N-engage: A new framework for measuring user engagement of website with user navigational behavior," *IEEE Access*, vol. 11, pp. 112276–112292, 2023.

[12] X. Wang, J. Li, and Y. Chen, "Scalable clustering algorithms for big data: A review," *ACM Trans. Knowl. Discovery Data*, vol. 18, no. 2, pp. 1–35, 2024.

[13] C. C. Aggarwal and C. K. Reddy, *Data Clustering: Algorithms and Applications* Boca Raton, FL, USA: CRC Press, 2013.

[14] C. C. Aggarwal, A. Hinneburg, and D. A. Keim, "On the surprising behavior of distance metrics in high dimensional space," in *Proc. Int. Conf. Database Theory*, 2001, pp. 420–434.

[15] H.-H. Zhao, X.-C. Luo, R. Ma, and X. Lu, "An extended regularized K-means clustering approach for high-dimensional customer segmentation with correlated variables," *IEEE Access*, vol. 9, pp. 48405–48412, 2021.

[16] T. Zhou, W. Liu, and J. Chen, "Streaming clustering algorithms: A review," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 3, pp. 2841–2857, Mar. 2024.

[17] F. Murtagh and P. Contreras, "Algorithms for hierarchical clustering: An overview," *WIREs Data Mining Knowl. Discovery*, vol. 2, no. 1, pp. 86–97, Jan. 2012.

[18] J. B. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proc. 5th Berkeley Symp. Math. Statist. Probab.*, vol. 1, 1967, pp. 281–297.

[19] M. Ester, H. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. 2nd Int. Conf. Knowl. Discovery Data Mining (KDD)*, 1996, pp. 226–231.
[20] S. C. Johnson, "Hierarchical clustering schemes," *Psychometrika*, vol. 32, no. 3, pp. 241–254, Sep. 1967.
[21] D. Arthur and S. Vassilvitskii, "K-means++: The advantages of careful seeding," in *Proc. 18th Annu. ACM-SIAM Symp. Discrete Algorithms*, 2007, pp. 1027–1035.
[22] I. T. Jolliffe, *Principal Component Analysis*. Cham, Switzerland: Springer, 2002.
[23] K. Zhang, X. Wang, and Y. Liu, "Deep clustering: A comprehensive survey," *IEEE Trans. Pattern Analysis Mach. Intell.*, vol. 45, no. 8, pp. 10091–10111, Aug. 2023.
[24] W. Chen, L. Zhang, and M. Wang, "Autoencoder-based deep learning for high-dimensional data clustering: A survey," *Neural Netw.*, vol. 162, pp. 185–205, Jun. 2023.
[25] Y. Ren, K. Hu, X. Dai, L. Pan, S. C. Hoi, and Z. Xu, "Deep clustering: A comprehensive survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 5, pp. 3213–3232, May 2024.
[26] Y. Li, W. Zhang, and C. Wang, "Ensemble learning for deep learning: A survey," *Inf. Fusion*, vol. 92, pp. 34–56, Apr. 2023.
[27] H. Liu, Z. Ming, and Y. Sun, "Consensus clustering: A survey and new methods," *Pattern Recognit.*, vol. 135, Mar. 2023, Art. no. 109177.
[28] T. Zhang, R. Ramakrishnan, and M. Livny, "Birch: An efficient data clustering method for very large databases," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 1996, pp. 103–114.
[29] B. Lorbeer, A. Kosareva, B. Deva, D. Softić, P. Ruppel, and A. Küpper, "Variations on the clustering algorithm BIRCH," *Big Data Res.*, vol. 11, pp. 44–53, Mar. 2018.
[30] A. Strehl and J. Ghosh, "Cluster ensembles—A knowledge reuse framework for combining multiple partitions," *J. Mach. Learn. Res.*, vol. 3, pp. 583–617, Dec. 2002.
[31] A. L. N. Fred and A. K. Jain, "Combining multiple clusterings using evidence accumulation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 6, pp. 835–850, Jun. 2005.
[32] C. Li, "Birch-ae: A hierarchical ensemble framework for scalable e-commerce user segmentation with autoencoder-enhanced feature learning," Zenodo, Geneva, Switzerland, Tech. Rep. 17498249, Jun. 2026, doi: 10.5281/zenodo.17498249.
[33] C. Fraley and A. E. Raftery, "Model-based clustering, discriminant analysis, and density estimation," *J. Amer. Stat. Assoc.*, vol. 97, no. 458, pp. 611–631, Jun. 2002.
[34] T. Rebafka, M. Sedki, and G. Celeux, "Model-based clustering with missing not at random data," *J. Classification*, vol. 41, no. 1, pp. 78–104, 2024.
[35] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 14, Jordan, 2001, pp. 849–856.
[36] D. A. Reynolds, "Gaussian mixture models," *Encyclopedia Biometrics*, S. Z. Li and A. K. Jain, Eds. Boston, MA, USA: Springer, 2009, pp. 659–663.
[37] T. Zhang, R. Ramakrishnan, and M. Livny, "BIRCH: A new data clustering algorithm and its applications," *Data Mining Knowl. Discovery*, vol. 1, no. 2, pp. 141–182, Jun. 1997.
[38] S. Vega-Pons and J. Ruiz-Shulcloper, "A survey of clustering ensemble algorithms," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 25, no. 3, pp. 337–372, May 2011.
[39] D. Huang, C.-D. Wang, J.-S. Wu, J.-H. Lai, and C.-K. Kwoh, "Ultra-scalable spectral clustering and ensemble clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 6, pp. 1212–1226, Jun. 2020.
[40] F. Li, Y. Qian, J. Wang, C. Dang, and L. Jing, "Clustering ensemble based on sample's stability," *Artif. Intell.*, vol. 273, pp. 37–55, Aug. 2019.
[41] R. A. FISHER, "The use of multiple measurements in taxonomic problems," *Ann. Eugenics*, vol. 7, no. 2, pp. 179–188, Sep. 1936.
[42] L. V. D. Maaten and G. E. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 86, pp. 2579–2605, 2008.
[43] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, Jul. 2006.
[44] J. Xie, R. Girshick, and A. Farhadi, "Unsupervised deep embedding for clustering analysis," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2015, pp. 478–487.
[45] P. J. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis," *J. Comput. Appl. Math.*, vol. 20, pp. 53–65, Nov. 1987.
[46] T. Calinski and J. Harabasz, "A dendrite method for cluster analysis," *Commun. Statistics-Theory Methods*, vol. 3, no. 1, pp. 1–27, 1974.
[47] D. L. Davies and D. W. Bouldin, "A cluster separation measure," *IEEE Trans. Pattern Anal. Mach. Intell.*, vols. PAMI–1, no. 2, pp. 224–227, Apr. 1979.
[48] G. Barbato, E. M. Barini, G. Genta, and R. Levi, "Features of the iqr method to detect outliers in industrial data: A case study," *IEEE Trans. Instrum. Meas.*, vol. 60, no. 9, pp. 3613–3619, Aug. 2011.
[49] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.
[50] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[51] H. W. Kuhn, "The Hungarian method for the assignment problem," *Nav. Res. Logistics Quart.*, vol. 2, nos. 1–2, pp. 83–97, 1955.
[52] Retail Rocket. (2016). *Retailrocket Recommender System Dataset*. [Online]. Available: https://www.kaggle.com/datasets/retailrocket/ecommerce-dataset
[53] M. Kechinov. (2019). *Ecommerce Behavior Data From Multi Category Store*. [Online]. Available: https://www.kaggle.com/datasets/mkechinov/ecommerce-behavior-data-from-multi-category-store
[54] J. H. Ward, "Hierarchical grouping to optimize an objective function," *J. Amer. Stat. Assoc.*, vol. 58, no. 301, pp. 236–244, Mar. 1963.

CAIWEN LI is currently pursuing the Ph.D. degree with the Faculty of Computer Science and Information Technology, Universiti Putra Malaysia (UPM). She has years of experience as a data scientist, she has held various roles, most recently at Amazon Web Services. She specializes in business intelligence and digital marketing, leveraging her expertise in data science, including forecasting, feature/model selection, data mining, APIs, and cloud technologies. Her primary research interests include recommendation systems, business intelligence, deep learning, hierarchical clustering methods, and scalable data mining algorithms for e-commerce applications.

ISKANDAR ISHAK received the Bachelor of Information Technology degree from Universiti Tenaga Nasional, Malaysia, the Master of Technology degree in information technology from the Royal Melbourne Institute of Technology, Australia, and the Ph.D. degree in computer science from Universiti Teknologi Malaysia. He is currently a Senior Lecturer with Universiti Putra Malaysia. His research interests include database systems, big data, data analytics, and scalable clustering algorithms.

HAMIDAH IBRAHIM (Member, IEEE) received the Ph.D. degree in computer science from the University of Wales, Cardiff, U.K., in 1998. She is currently a Full Professor with the Faculty of Computer Science and Information Technology, Universiti Putra Malaysia (UPM). Her current research interests include databases (distributed, parallel, mobile, biomedical, XML) focusing on issues related to integrity maintenance/checking, ontology/schema/data integration, ontology/schema/data mapping, cache management, access control, data security, transaction processing, query optimization, query reformulation, preference evaluation (context-aware), information extraction, and concurrency control.

MASLINA ZOLKEPLI received the bachelor's and master's degrees in computer science from Universiti Putra Malaysia, in 2007 and 2010, respectively, and the Ph.D. degree in computational intelligence and systems science from Tokyo Institute of Technology, Japan, in 2015. She is currently a Senior Lecturer with the Department of Computer Science, Faculty of Computer Science and Information Technology, Universiti Putra Malaysia. Her research interests include business analytics, fuzzy brake systems, computational intelligence, and ensemble learning methods.

FATIMAH SIDI (Member, IEEE) received the Ph.D. degree in management information systems from Universiti Putra Malaysia (UPM), Malaysia, in 2008. She is currently an Associate Professor with the Department of Computer Science, Faculty of Computer Science and Information Technology, UPM. Her current research interests include knowledge and information management systems, data and knowledge engineering, databases, data warehouses, big data, and data analytics.

CAILI LI is currently pursuing the Ph.D. degree in design and architecture with Universiti Putra Malaysia, she conducts interdisciplinary research at the intersection of art and technology. She was a Visiting Scholar at Tsinghua University, she brings extensive expertise and insights to her work, aiming to advance the field of design and to inspire innovative solutions through data-driven visualization. She is currently an Associate Professor with Heilongjiang Institute of Technology, China, specializing in design art, visual arts, analytical visualization, integrated arts, and landscape design.