INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD)
© 2026 IJNRD | Volume 11, Issue 4, April 2026 | ISSN: 2456-4184 | IJNRD.ORG

A COMPREHENSIVE SURVEY ON DOCUMENT CLUSTERING TECHNIQUES

Priyadharshini V, Dr. R. Nagarajan
Research Scholar, Assistant Professor/Programmer
Annamalai University

ABSTRACT

Document clustering plays a crucial role in text mining and natural language processing by organizing large volumes of unstructured textual data into meaningful groups without requiring labeled instances. With the rapid growth of digital information, efficient clustering techniques have become essential for tasks such as information retrieval, topic discovery, and knowledge organization. This survey provides a structured analysis of document clustering approaches by categorizing them into three major groups: classical methods, probabilistic models, and nature-inspired optimization techniques. Classical approaches, including K-Means, hierarchical clustering, DBSCAN, and graph-based methods, offer computational efficiency but often struggle to capture deeper semantic relationships. In contrast, probabilistic models such as Latent Dirichlet Allocation (LDA), Gaussian Mixture Models (GMM), and Hidden Markov Models (HMM) aim to uncover latent structures within text data, albeit with increased computational complexity. Furthermore, nature-inspired algorithms-including Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), Artificial Bee Colony (ABC), and Grey Wolf Optimization (GWO)-enhance clustering performance through global optimization strategies. The study critically examines the strengths and limitations of these techniques and emphasizes the importance of hybrid and adaptive frameworks to improve clustering accuracy, scalability, and semantic representation.

Keywords:

Document Clustering, Text Mining, NLP, K-Means, Hierarchical Clustering, DBSCAN, Graph-based, LDA, GMM, HMM, Nature-Inspired Algorithms, GA, PSO, ACO, ABC, GWO

1. INTRODUCTION

The exponential growth of textual data generated from sources such as web platforms, social media, and scientific publications has created significant challenges in information organization and retrieval. Document clustering has emerged as an effective unsupervised learning technique that automatically groups similar documents based on their content, enabling applications such as topic detection, recommendation systems, and data summarization.

Conventional clustering techniques primarily rely on statistical representations and distance-based similarity measures. While these approaches are computationally efficient, they often fail to capture the semantic and contextual relationships inherent in textual data. To address these limitations, more advanced methods have been developed. Probabilistic models provide a framework for identifying hidden thematic structures, whereas nature-inspired optimization algorithms improve clustering quality by exploring complex solution spaces.

This survey aims to present a comprehensive and structured overview of document clustering techniques by organizing them into three major categories: classical methods, probabilistic approaches, and nature-inspired algorithms. In addition, the study highlights key challenges and identifies potential research directions for developing more robust and scalable clustering frameworks.

2. LITERATURE REVIEW

Document clustering has been extensively explored as a fundamental approach for managing large-scale unstructured text data. Existing studies can be broadly grouped into classical clustering techniques, probabilistic models, and optimization-based approaches, each contributing unique advantages and limitations.

Classical clustering methods, such as K-Means, hierarchical clustering, and DBSCAN, continue to be widely adopted due to their simplicity and efficiency [1], [2]. K-Means is particularly suitable for large datasets; however, its dependence on initial centroid selection and the requirement to predefine the number of clusters limit its robustness [3]. Hierarchical clustering offers improved interpretability through tree-like structures but suffers from high computational complexity, making it less suitable for large-scale applications [4]. DBSCAN addresses some of these limitations by identifying arbitrarily shaped clusters and effectively handling noise, although its performance is sensitive to parameter selection [5]. Graph-based clustering methods have gained attention for their ability to model documents as interconnected nodes, enabling the capture of complex relationships that traditional distance-based approaches often overlook [6]. Nevertheless, these methods are computationally intensive and rely heavily on similarity measures [7], [8].

To overcome the limitations of classical approaches, probabilistic models have been introduced to capture latent semantic structures within documents. Latent Dirichlet Allocation (LDA) is widely used for topic modeling, representing documents as mixtures of latent topics [9], [10]. Despite its effectiveness, the assumption of word independence restricts its ability to model contextual dependencies [11]. Gaussian Mixture Models (GMM) provide a flexible framework by allowing soft clustering, where documents can belong to multiple clusters with varying probabilities [12]. Hidden Markov Models (HMM) extend this capability to sequential data by modeling temporal dependencies, although their application in general document clustering remains limited [13]. These probabilistic approaches, while powerful, often involve complex parameter estimation and increased computational cost [14], [15].

In recent years, nature-inspired algorithms have been increasingly applied to enhance clustering performance through optimization techniques. Algorithms such as Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), Artificial Bee Colony (ABC), and Grey Wolf Optimization (GWO) aim to overcome the limitations of traditional methods by providing global search capabilities [16]-[22]. These approaches are effective in avoiding local optima and improving clustering quality; however, they typically require careful parameter tuning and may incur higher computational overhead [23], [24].

Recent research trends emphasize the integration of multiple approaches to leverage their complementary strengths. Hybrid clustering frameworks that combine classical, probabilistic, and optimization-based methods have demonstrated improved performance in terms of accuracy and scalability [25], [26]. Despite these advancements, challenges such as high-dimensional data representation, parameter sensitivity, and semantic understanding continue to persist, highlighting the need for more adaptive and intelligent clustering models [27]-[34].

3. DOCUMENT CLUSTERING TECHNIQUES

3.1 Classical methods

Classical methods, also referred to as classical clustering techniques, are among the earliest and most widely used approaches for document clustering. These methods primarily rely on geometric distance measures and statistical properties of the data to partition documents into meaningful groups. In document clustering, each document is typically represented as a high-dimensional feature vector (e.g., TF-IDF), and clustering is performed by minimizing intra-cluster dissimilarity while maximizing inter-cluster separation.

The general objective of standard clustering methods can be expressed as an optimization problem:

$$\min J = \sum i = 1K \sum xj \in Cid(xj, \mu i)$$

where J represents the clustering objective function, Cidenotes the ith cluster, xj is a document vector,mui is the centroid of cluster Ci, and $d(cdot)$ is a distance metric such as Euclidean distance or cosine similarity. These methods are computationally efficient and easy to implement, making them suitable for large-scale applications. However, they often struggle to capture semantic relationships in textual data due to their reliance on surface-level features.

Classical methods can be broadly categorized into partition-based, hierarchical, density-based and graph-based approaches.

3.1.1 K-Means

K-Means is a partition-based algorithm that divides documents into K clusters by minimizing intra-cluster variance. It iteratively updates cluster centroids and assigns documents based on proximity.

Advantages

- Simple and computationally efficient
- Scalable to large datasets
- Easy to implement

Limitations

- Requires predefined number of clusters
- Sensitive to initialization
- Assumes spherical cluster structures

3.1.2 Hierarchical Clustering

Hierarchical clustering is a widely used unsupervised learning technique that constructs a hierarchy of clusters either through a bottom-up (agglomerative) or top-down (divisive) approach. In document clustering, each document is initially treated as an individual cluster, and pairs of clusters are iteratively merged based on a defined similarity or distance measure until a single cluster is formed. Alternatively, in divisive clustering, the process begins with all documents in one cluster and recursively splits them into smaller clusters.

Advantages

- No need to predefine cluster number
- Produces interpretable dendrograms
- Flexible distance metrics

Limitations

- High computational complexity
- Not scalable for large datasets
- Sensitive to noise

3.1.3 DBSCAN

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a density-based clustering algorithm that groups documents based on the notion of density connectivity. Unlike partition-based methods, DBSCAN does not require the number of clusters to be predefined and is capable of identifying clusters of arbitrary shapes while effectively handling noise and outliers. In document clustering, each document is represented as a feature vector (e.g., TF-IDF or embeddings), and clusters are formed by identifying dense regions in the feature space.

Advantages

- Detects arbitrary-shaped clusters
- Handles noise effectively
- No need to specify number of clusters

Limitations

- Sensitive to parameter selection
- Poor performance in high-dimensional spaces
- Difficulty with varying densities

3.1.4 Graph-Based Clustering

Graph-based clustering represents documents as nodes in a graph, where edges indicate similarity between documents. In document clustering, a similarity matrix (e.g., cosine similarity) is used to construct the graph, and clusters are formed by partitioning the graph into groups with strong intra-cluster connections and weak inter-cluster connections.

The clustering objective is often defined using the cut function:

$$\mathrm{Cut}(A, B) = i \in A, j \in B \sum wij$$

where wij represents similarity between documents.

Advantages

- Captures complex document relationships
- Suitable for similarity-based clustering

Limitations

- Computationally expensive
- Sensitive to similarity measure

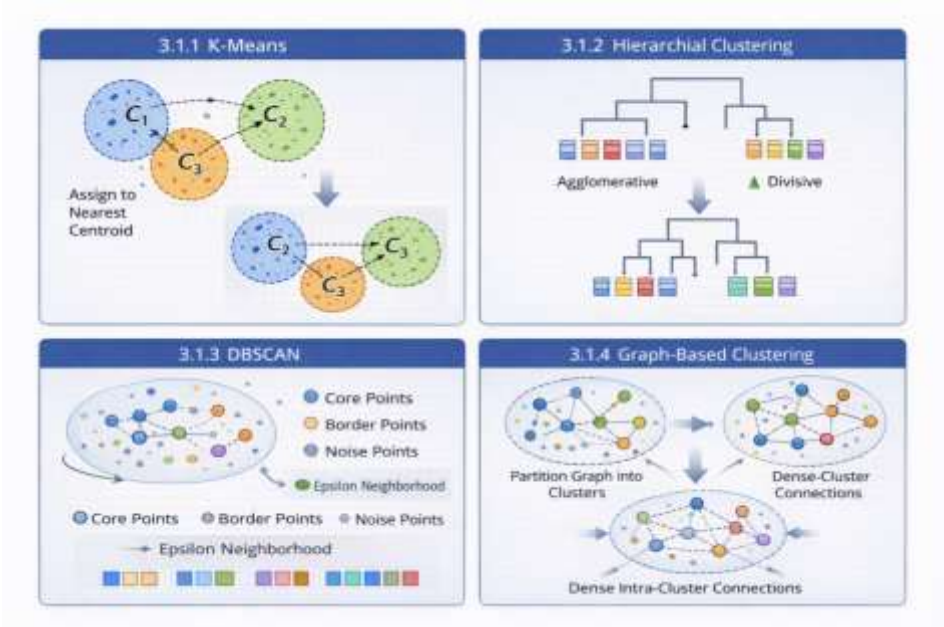

Fig. 3.1 Classical methods

3.2 Probabilistic Methods

Probabilistic methods in document clustering are based on the assumption that documents are generated from underlying statistical distributions. Unlike traditional clustering approaches that rely on geometric distance measures, probabilistic models aim to capture the latent structure of textual data by modeling the probability of word occurrences and document-topic relationships. These methods provide a principled framework for handling uncertainty, variability, and hidden semantic structures within large text corpora.

In probabilistic clustering, each document is typically represented as a mixture of latent variables (such as topics), and clustering is achieved by estimating the probability distribution that best explains the observed data. The general objective is to maximize the likelihood of the dataset under the assumed probabilistic model, which can be expressed as:

$$L = \prod i = 1NP(xi \mid \theta)$$

where xi represents theith document, theta denotes the model parameters, and N is the total number of documents. The goal is to estimate theta such that the likelihood of the observed data is maximized.

3.2.1 Latent Dirichlet Allocation (LDA)

Latent Dirichlet Allocation (LDA) is one of the most widely used probabilistic models for document clustering and topic modeling. It assumes that each document is a mixture of multiple topics, and each topic is a probability distribution over words.
The generative process of LDA can be mathematically represented as:

$$P(w \mid d) = \sum k = 1KP(w \mid zk)P(zk \mid d)$$

where $\boldsymbol{P(wmidd)}$ is the probability of word w in document d, zk represents the kth topic, and K is the total number of topics.

Advantages

- Produces interpretable topic distributions
- Effective for large-scale text corpora
- Captures latent semantic structure

Limitations

- Assumes independence between words (bag-of-words assumption)
- Requires careful tuning of hyperparameters
- Limited contextual understanding

3.2.2 Gaussian Mixture Models (GMM)

Gaussian Mixture Models represent data as a mixture of multiple Gaussian distributions, where each component corresponds to a cluster. Unlike K-Means, GMM provides soft clustering by assigning probabilities to each document belonging to different clusters.

The probability density function of GMM is defined as:

$$P(x) = \sum k = 1K\pi kN(x \mid \mu k, \Sigma k)$$

Where pik is the mixing coefficient, muk is the mean, and Sigma k is the covariance matrix of the hkth Gaussian component.

Advantages

- Supports soft clustering
- Models complex data distributions
- More flexible than K-Means

Limitations

- Computationally expensive
- Sensitive to initialization
- Struggles with high-dimensional sparse text data

3.2.3 Hidden Markov Models (HMM)

Hidden Markov Models are probabilistic models used to represent sequential data, where the system is assumed to follow a Markov process with hidden states. In document clustering, HMMs can model sequences of words or topics, making them suitable for structured or temporal text data.

The joint probability of observed and hidden states is given by:

$$P(X, Z) = \prod t = 1TP(zt \mid zt - 1)P(xt \mid zt)$$

where X represents the observed sequence of words and Z represents the hidden states.

Advantages

- Captures sequential dependencies
- Suitable for temporal and structured text

Limitations

- Limited applicability for general document clustering
- High computational complexity
- Requires large training data

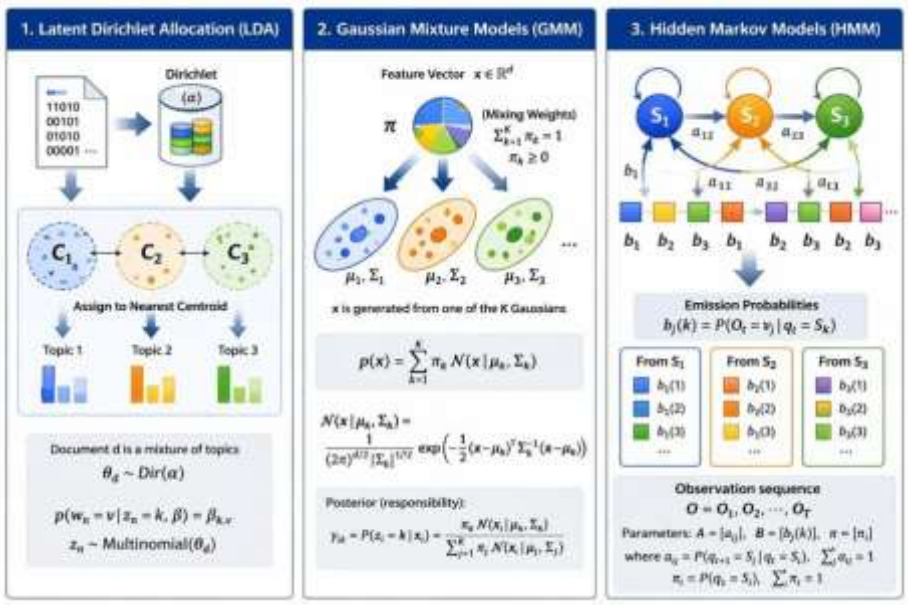

Fig. 3.2 Probabilistic Methods

3.3 Nature-Inspired Algorithms

Nature-inspired algorithms are a class of optimization techniques derived from biological, physical, and social behaviors observed in nature. In the context of document clustering**,** these algorithms are primarily employed to optimize clustering performance by improving intra-cluster similarity and maximizing inter-cluster separation. Unlike traditional clustering methods, which often suffer from local optima, nature-inspired approaches provide a global search capability that enables better exploration of the high-dimensional document space.

In document clustering, each document is represented as a feature vector (e.g., TF-IDF or embeddings), and clustering aims to group similar documents based on an objective function. Nature-inspired algorithms optimize this clustering objective by iteratively refining candidate solutions. The general objective function is:

$$\min J = \sum i = 1K \sum xj \in Cid(xj, \mu i)$$

Where J represents clustering error, Ci denotes clusters, xj is a document vector, and μi is the centroid. These algorithms search for optimal cluster assignments or centroids to minimize this objective.

3.3.1 Genetic Algorithm (GA)

In document clustering, Genetic Algorithm (GA) is used to optimize cluster assignments by encoding documents or centroids as chromosomes. Through selection, crossover, and mutation operations, GA evolves solutions that minimize clustering error.

The fitness function used in clustering can be defined as:

$$F = 1J$$

where J is the clustering objective function. The goal is to maximize the fitness value by minimizing clustering error.

Advantages

- Capable of global optimization
- Avoids local minima
- Flexible and adaptable

Limitations

- High computational cost
- Slow convergence rate
- Requires careful parameter tuning

3.3.2 Particle Swarm Optimization (PSO)

Particle Swarm Optimization is inspired by the social behavior of birds flocking or fish schooling. Each particle represents a potential solution and adjusts its position based on its own experience and that of neighboring particles.

The velocity and position updates are given by:

$$\mathrm{vi(t+1)} = \mathrm{wvi(t)} + \mathrm{c1r1(pi - xi)} + \mathrm{c2r2(g - xi)}$$

$$\mathrm{xi(t+1)} = \mathrm{xi(t)} + \mathrm{vi(t+1)}$$

where xi and vi represent the position and velocity of particle i, pi is the personal best, and g is the global best.

Advantages

- Fast convergence
- Simple implementation
- Effective for continuous optimization

Limitations

- May converge prematurely
- Sensitive to parameter settings

3.3.3 Ant Colony Optimization (ACO)

In document clustering, ACO is used to discover optimal grouping structures by simulating the behavior of ants searching for food. Documents are clustered based on probabilistic decisions influenced by pheromone trails and similarity measures.

The probability of selecting a path is given by:

$$\mathrm{Pij} = \tau\mathrm{ij}\alpha \cdot \eta\mathrm{ij}\beta \sum \mathrm{k}\tau\mathrm{ik}\alpha \cdot \eta\mathrm{ik}\beta$$

ACO is particularly useful for clustering problems that can be modeled as graph-based structures, where documents are nodes and similarities form edges.

Advantages

- Good for combinatorial optimization
- Finds optimal paths effectively

Limitations

- High computational overhead
- Slow convergence for large datasets

3.3.4 Artificial Bee Colony (ABC)

Artificial Bee Colony algorithm is inspired by the foraging behavior of honey bees. It consists of employed bees, onlooker bees, and scout bees, each contributing to the search for optimal solutions.

The probability of selecting a solution is defined as:

$$Pi = fi\sum j = 1Nfj$$

where fi represents the fitness of solution i.

Advantages

- Strong exploration capability
- Avoids local optima

Limitations

- Slower convergence in some cases
- Performance depends on parameter tuning

3.3.5 Grey Wolf Optimization (GWO)

In document clustering, ABC algorithm searches for optimal clustering solutions by simulating the foraging behavior of bees. Each solution represents a clustering configuration, and bees explore and exploit the search space to improve clustering quality.

$$D^{\rightarrow} =| C^{\rightarrow} \cdot X^{\rightarrow}p - X^{\rightarrow} |$$

$$X^{\rightarrow}(t+1) = X^{\rightarrow}p - A^{\rightarrow} \cdot D^{\rightarrow}$$

In document clustering, GWO is applied to optimize cluster centroids by mimicking the leadership hierarchy and hunting behavior of grey wolves. The best solutions (alpha, beta, delta) guide the search process toward optimal clustering configurations.

Advantages

- Strong global search capability
- Balances exploration and exploitation effectively
- Avoids local optima
- Suitable for optimizing clustering parameters

Limitations

- May converge slowly in some cases
- Performance depends on parameter tuning
- Computational cost increases with dataset size

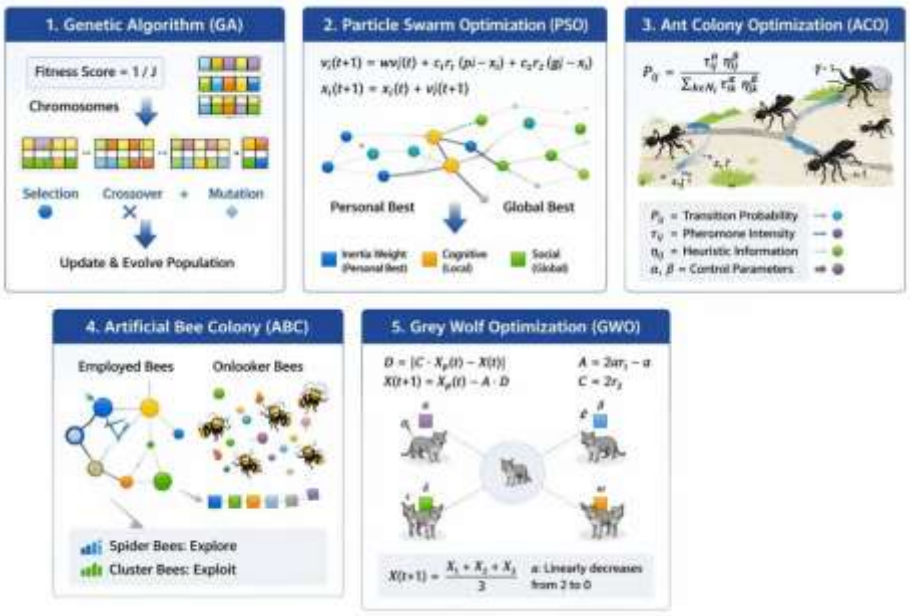

Fig. 3.3 Nature-Inspired Algorithms

4. COMPARATIVE ANALYSIS

Techniques	Category	Key Idea	Advantages	Limitations
K-Means	Classical	Centroid-based clustering	Fast, scalable, simple	Requires K, sensitive to initialization
Hierarchical	Classical	Tree-based clustering	No need for K, interpretable	High complexity, not scalable
DBSCAN	Classical	Density-based clustering	Detects noise, arbitrary shapes	Parameter sensitive, poor in high dimensions
Graph-Based Clustering	Classical	Graph representation using nodes and edges similarities	Captures complex relations, flexible	High complexity, graph construction cost, parameter sensitive
LDA	Probabilistic	Topic distribution model	Semantic understanding, interpretable	Assumes independence, tuning required
GMM	Probabilistic	Gaussian distribution	Soft clustering, flexible	Computationally expensive
HMM	Probabilistic	Sequential probabilistic model	Captures sequence patterns	Limited for general clustering

GA	Nature-Inspired	Evolution-based optimization	Global search, avoids local minima	Slow convergence
PSO	Nature-Inspired	Swarm-based optimization	Fast convergence, simple	Premature convergence
ACO	Nature-Inspired	Pheromone-based search	Good for optimization	High computational cost
ABC	Nature-Inspired	Bee foraging behavior	Strong exploration	Slow convergence
GWO	Nature-Inspired	Wolf hunting strategy	Balanced exploration & exploitation	Parameter sensitivity

Table 4.1 Comparative Analysis

5. CHALLENGES

Document clustering faces several important challenges, including high dimensionality and sparsity of text data, which reduce the effectiveness of similarity measures. classical methods lack semantic understanding, while probabilistic models involve complex parameter estimation and high computational cost. Nature-inspired algorithms, although powerful in optimization, suffer from parameter sensitivity, slow convergence in some cases, and scalability issues when applied to large datasets. Additionally, the presence of noise and outliers further degrades clustering performance.

6. FUTURE RESEARCH DIRECTIONS

Future research should focus on developing hybrid and adaptive clustering frameworks that combine the strengths of classical, probabilistic, and nature-inspired approaches. The integration of advanced semantic representations, such as contextual embeddings, can improve clustering accuracy. Moreover, designing self-tuning algorithms to reduce parameter dependency and improving scalability for large-scale and real-time data are key directions. Enhancing optimization efficiency and robustness in dynamic environments also remains an important area for further investigation.

7. CONCLUSION

This survey reviewed document clustering techniques across classical, probabilistic, and nature-inspired approaches. classical methods provide simplicity and efficiency but lack semantic understanding, while probabilistic models capture latent structures at the cost of increased complexity. Nature-inspired algorithms enhance clustering performance through effective optimization but require higher computational resources.

Overall, each technique has distinct strengths and limitations, indicating the need for hybrid and adaptive models. Future research should focus on improving scalability, semantic representation, and optimization strategies to achieve more accurate and efficient document clustering.

REFERENCES

[1]J. Han, M. Kamber, and J. Pei, Data Mining: Concepts and Techniques, 3rd ed. Waltham, MA, USA: Morgan Kaufmann, 2011.

[2] J. MacQueen, “Some methods for classification,” 1967.

[3] D. Arthur and S. Vassilvitskii, “K-means++,” 2007.DOI:https://doi.org/10.1145/1283383.1283494

[4] F. Murtagh and P. Contreras, "Hierarchical clustering," 2012. DOI: https://doi.org/10.1002/widm.53

[5] M. Ester et al., "DBSCAN," 1996. DOI: https://doi.org/10.5555/3001460.3001507

[6] Zakariyaa Ait El Mouden, A Survey of Graph-based Clustering Techniques. DOI: 10.1109/ADACIS59737.2023.10424063

[7] A. K. Jain, "Data clustering review," 2010. DOI:https://doi.org/10.1016/j.patrec.2009.09.011

[8] C. Aggarwal and C. Zhai, Mining Text Data, 2012. DOI: https://doi.org/10.1007/978-1-4614-3223-4

[9] D. Blei et al., "LDA," 2003. DOI: https://doi.org/10.5555/944919.944937

[10] T. Griffiths and M. Steyvers, "Topic modeling," 2004.

[11] H. Wallach, "Topic modeling beyond bag-of-words," 2006.

[12] C. Bishop, Pattern Recognition, 2006.

[13] L. Rabiner, "HMM tutorial," 1989. DOI: https://doi.org/10.1109/5.18626

[14] Z. Ghahramani, "Probabilistic models," 2004.

[15] K. Murphy, Machine Learning, 2012.

[16] D. E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning. Boston, MA, USA: Addison-Wesley, 1989.

[17] D. Goldberg, Genetic Algorithms, 1989.

[18] J. Kennedy and R. Eberhart, "PSO," 1995. DOI: https://doi.org/10.1109/ICNN.1995.488968

[19] M. Dorigo and T. Stützle, Ant Colony Optimization. Cambridge, MA, USA: MIT Press, 2004.

[20] D. Karaboga, "ABC algorithm," 2005.

[21] S. Mirjalili, "Grey Wolf Optimizer," 2014. DOI:https://doi.org/10.1016/j.advengsoft.2013.12.007

[22] X. Yang, Nature-Inspired Optimization, 2014.

[23] X. Li et al., "Deep clustering survey," 2023. DOI: https://doi.org/10.1016/j.neucom.2023.126123

[24] P. Kumar et al., "Hybrid clustering," 2024. DOI: https://doi.org/10.1016/j.eswa.2023.121234

[25] R. Gupta et al., "Optimization clustering," 2024.DOI:https://doi.org/10.1016/j.asoc.2024.110123

[26]N. Ahmed et al., "Clustering optimization," 2023. DOI: https:// doi.org/10.1109/ACCESS.2023.3298765

[27] X. Sun et al., "Deep clustering," 2024. DOI: https://doi.org/10.1016/j.ins.2024.119876

[28]Y. Zhao et al., "Multimodal clustering," 2025.DOI:https://doi.org/10.1016/j.neucom.2024.127654

[29] A. Khan et al., "Clustering challenges," 2023. DOI: https://doi.org/10.1111/exsy.13245

[30] M. Ali et al., "Big data clustering," 2023. DOI:https://doi.org/10.1016/j.future.2023.02.012

[31] L. Chen et al., "Scalable clustering," 2022. DOI:https://doi.org/10.1016/j.patcog.2022.108765

[32]H. Zhang et al., "Future clustering trends," 2025.DOI:https://doi.org/10.1109/A CCESS.2025.3456789

[33] Hybrid topic modeling for short text clustering (2024). DOI: https://doi.org/10.1186/s40537-024-00930-9

[34] Hybrid feature selection + clustering (2025). https://doi.org/10.1016/j.eswa.2025.128762