Early Clustering Techniques

Foundations for Modern Algorithms in Euclidean and Non-Euclidean Spaces

Project Goals

- Understand the **CURE** algorithm and **GRGPF** Framework
- Identify the **motivations** behind their key innovations
- **Highlight modern parallels** to develop and intuition for modern clustering design strategies

Focus of this Presentation

- Explain the **core problems** that CURE and GRGPF were designed to address
- Identify **where the techniques** introduced in these algorithms **persist** today

CURE (1998)

S. Guha, R. Rastogi, and K. Shim

An Efficient Clustering

Algorithm for Large Databases

Core Problems (1/2)

Previous algorithms suffered from **two major flaws** which deteriorated the quality of their clusters:

- High **sensitivity** to outliers
- **Bias** toward form spherical clusters with similar sizes

Solution

- Biases were a **consequence of using single points** to represent clusters
- CURE represents each cluster with multiple **well-scattered representative points**
- Representative points are **shrunk towards the mean** to reduce impact of outliers

Modern Parallels

- **Prototypes** – originally single data points representing clusters
- CURE popularized the use of **multiple prototypes** per cluster
- Modern prototypes can be complex vectors or learned parameters, but their role still **mirrors representative points**

Core Problems (2/2)

- Previous algorithms could not be effectively scaled to large data sets

- A critical weakness as data sets often consist of hundreds of thousands of points and beyond

Solution

- Generate a **uniformly random subset** of the data
- Split data into $\boldsymbol{p}$ **partitions**
- Generate $\boldsymbol{q}$ **pre-clusters** in each partition
- **Merge pre-clusters** with CURE to form final clusters

Modern Parallels

- **Coresets** – small subsets used to approximate data
- Uses **non-uniform sampling** based on a various weighting schemes
- Achieves **accurate approximations with fewer points**

GRGPF Framework (1999)

V. Ganti, R. Ramakrishnan, J. Gherke, A.L. Powell, and J.C. French

Clustering Large Datasets in Arbitrary Metric Spaces

Core Problems (1/2)

- **Limited operations** available in distance spaces
- With limited operations, **distances spaces have difficulties creating cluster summaries** as seen commonly in coordinate spaces
- Without summaries, **computing distances may require examining entire clusters**

Solution (1/2)

- **Generalized cluster features** - first of two core abstractions in the GRGPF framework

- Stores summary data for a cluster

- Must satisfy two properties:
  - Can be **incrementally updated** as new points are introduced
  - Can be used to **compute distances and other metrics**

Solution (2/2)

- **Generalized cluster features tree** – second core abstraction
- Guides objects to their best cluster
- Composed of non-leaf and leaf nodes:
  - Non-leaf nodes – **direct objects** toward the appropriate leaf
  - Leaf nodes – contain **candidate clusters**

Use Case Example (BUBBLE-FM)

- Cluster features: cluster size, a clustroid, and clustroid radius

- Cluster feature tree:
  - Interior nodes contain a constant number of **sample objects**
  - Leaf nodes contains a set of **candidate clusters**

- Guide new objects through closest interior nodes to a leaf containing their optimal cluster

Modern Parallels

- **Micro-clusters** used in data stream clustering are similar to generalized cluster features
- Used to summarize dense regions of points
- **Tree structures** built from micro-clusters guide the clustering process, mirroring cluster feature trees

Core Problems (2/2)

- Distance computations in arbitrary metric spaces can be **costly**
  - Example: Hamming distance between bitstrings

- This can make the guidance stage of the GRGPF framework expensive

Solution

- Create **image vectors** for sample objects in interior nodes
- Generate an **image space** to compute approximated centroids for interior nodes
- Compare new objects' image vectors to these centroids
- Leaf nodes remain unchanged to maintain quality

Modern Parallels

- Mapping complex data to simpler vector spaces persists:
  - Text clustering: documents → vector spaces
  - LLMS: semantic/syntactic patterns → vector spaces
  - Deep clustering: high-dimensional → low-dimensional spaces

- Motivation remains unchanged:
  - Transform data to facilitate new operations
  - Allow for cheaper distance computations

Conclusion

Presentation Review

- Highlighted the **core problems** CURE and GRGPF were introduced to solve
- Showed **modern parallels** to new innovations from these algorithms
- For a deeper dive into the algorithms, their design, and modern connections, see my **full paper**

References

Petukhova, A., Matos-Carvalho, J.P., & Fachada, N. (2024). *Text Clustering with Large Language Model Embeddings*. International Journal of Cognitive Computing in Engineering.

Wang, H., & Lu, N. (2020). *Deep Embedded Clustering with Asymmetric Residual Autoencoder*. In Proceedings of the 2020 Chinese Automation Congress (CAC).

Leskovec, J., Rajaraman, A., & Ullman, J.D. (2014). *Mining of Massive Datasets*. Cambridge University Press.

Sun, J., Du, M., & Dong, Y. (2025). *Efficient Online Stream Clustering Based on Fast Peeling of Boundary Micro-Cluster*. IEEE Transactions on Neural Networks and Learning Systems.

Liu, M., Jiang, X., & Kot, A.C. (2009). *A multi-prototype clustering algorithm*. In Proceedings of the 2009 Chinese Pattern Recognition Conference (CCPR).

Guha, S., Rastogi, R., & Shim, K. (1998). *CURE: An efficient clustering algorithm for large databases*. In Proceedings of the ACM SIGMOD International Conference on Management of Data (pp. 73–84).

Cohen-Addad, V., Saulpic, D., & Schwiegelshohn, C. (2021). *A new coreset framework for clustering*. In Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing (STOC) (pp. 169–182).

Ganti, V., Ramakrishnan, R., Gehrke, J., Powell, A.L., & French, J.C. (1999). *Clustering large datasets in arbitrary metric spaces*. In Proceedings of the International Conference on Data Engineering (pp. 502–511).

Ping, Y., Li, H., Guo, C., & Hao, B. (2025). *kProtoClust: Towards Adaptive k-Prototype Clustering without Known k*. Computers, Materials & Continua, 82(3), 4949–4976.