Bachelor Thesis Topic: Non-redundant Stream Clustering

Winter Semester 2026/27

1 Introduction and Background

While conventional clustering seeks the best grouping of similar data for a given dataset, clusters are not always that straightforward. In practice, there isn't a single best clustering, but rather a collection of valid but distinct clustering solutions. Different viewpoints of the same dataset can form equally suitable clusterings. As such, the discipline of non-redundant clustering [6] has emerged, with the goal of producing alternative yet interesting clusterings. Typically, this is done by treating different alternatives as results from different orthogonal subspaces.

Data streams are a challenging data management setting, where incoming data is potentially infinite, rapid, and subject to a data drift that alters distributions over time [7]. As such, special handling is required to cluster such data. A common framework is the Online-Offline framework [1], which splits the stream processing into two phases: the online phase summarizes the stream, and the offline phase intermittently processes these summaries into actual clusterings. The offline phase typically corresponds to a traditional clustering approach.

So far, there has been little research into combining non-redundant clustering with stream clustering. However, the offline phase is often easy to swap out and strongly affects the clustering result [3, 4]. As such, a possible strategy would be to replace the standard offline processing with an approach that supports non-redundant clustering.

For this thesis, the candidates for combination are CluStream [1], a foundational yet powerful stream clustering algorithm, and NR-Kmeans [6], a non-redundant clustering approach based on k-Means [5]. This combination seems most suitable, as CluStream already uses k-Means for offline processing, which is likely aligned with the properties of NR-Kmeans.

The thesis should primarily investigate how well this offline processing copes with the different limitations of the stream setting. In particular, data drift and a strong reduction in granularity are important. Comparison should be performed against both standard NR-Kmeans without the data stream limitations as well as standard k-Means as a post-processing technique. To facilitate this,

the investigation should span a significant set of datasets (e.g., from the Extreme Classification Repository [2] `http://manikvarma.org/downloads/XC/XMLRepository.html`, which offers many multi-label datasets) and different hyperparameter configurations.

2 Research Question

How suitable is non-redundant clustering with NR-Kmeans for the stream setting when combined with a standard stream processing framework?

3 Tasks & Goals

- **Literature Review:** Understanding the primary literature regarding NR-Kmeans and stream clustering
- **Implementation:** Combine NR-Kmeans with CluStream
- **Experiment Design:** Set up a suitable experimental environment to systematically examine the different constraints
- **Analysis & Discussion:** Investigation of the effects and difficulties that occur from the combination (and potentially investigate ways to resolve them)

4 Expected Outcomes

- An in-depth analysis of the suitability of post-processing with NR-Kmeans for non-redundant stream clustering
- A proper comparison of the CluStream combination with the constituting components
- A well-documented thesis with reproducible code.

5 Resources

Python implementations for NR-KMeans (`https://github.com/collinleiber/ClustPy`) and CluStream (`https://github.com/online-ml/river/`) are available. Multi-label datasets can be found in the Extreme Classification Repository [2] (`http://manikvarma.org/downloads/XC/XMLRepository.html`) and in ClustPy (`https://github.com/collinleiber/ClustPy`).

6 Requirements

- Study in the field of computer science
- Prior programming experience in Python
- Beneficial: understanding of clustering, Data Mining Algorithms 1/2

7 Contact

If you are interested, please send your CV and transcripts to jahn@dbs.ifi.lmu.de

References

[1] Charu C Aggarwal, S Yu Philip, Jiawei Han, and Jianyong Wang. A framework for clustering evolving data streams. In *Proceedings 2003 VLDB conference*, pages 81–92. Elsevier, 2003.

[2] K. Bhatia, K. Dahiya, H. Jain, P. Kar, A. Mittal, Y. Prabhu, and M. Varma. The extreme classification repository: Multi-label datasets and code, 2016.

[3] Matthias Carnein and Heike Trautmann. Optimizing data stream representation: An extensive survey on stream clustering algorithms. *Bus. Inf. Syst. Eng.*, 61(3):277–297, 2019.

[4] Philipp Jahn, Walid Durani, Collin Leiber, Anna Beer, and Thomas Seidl. Going offline: An evaluation of the offline phase in stream clustering. In Rita P. Ribeiro, Bernhard Pfahringer, Nathalie Japkowicz, Pedro Larrañaga, Alípio M. Jorge, Carlos Soares, Pedro H. Abreu, and João Gama, editors, *Machine Learning and Knowledge Discovery in Databases. Research Track - European Conference, ECML PKDD 2025, Porto, Portugal, September 15-19, 2025, Proceedings, Part VII*, volume 16019 of *Lecture Notes in Computer Science*, pages 308–326. Springer, 2025.

[5] Stuart P. Lloyd. Least squares quantization in PCM. *IEEE Trans. Inf. Theory*, 28(2):129–136, 1982.

[6] Dominik Mautz, Wei Ye, Claudia Plant, and Christian Böhm. Discovering non-redundant k-means clusterings in optimal subspaces. In Yike Guo and Faisal Farooq, editors, *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018, London, UK, August 19-23, 2018*, pages 1973–1982. ACM, 2018.

[7] Jonathan A Silva, Elaine R Faria, Rodrigo C Barros, Eduardo R Hruschka, André CPLF de Carvalho, and João Gama. Data stream clustering: A survey. *ACM Computing Surveys (CSUR)*, 46(1):1–31, 2013.