Bachelor Thesis Topic: Feasibility of Alternate Dimensionality-Reduction Techniques in EMCStream

Summer Semester 2026

1 Introduction and Background

Data streams are an important challenge in data mining, where data arrives continuously and potentially infinitely, while the methods processing them are limited to a fixed memory size [3]. The incoming data may be subject to data drift that alters the underlying distributions over time [5].

Stream clustering applies clustering, i.e., the unsupervised grouping of similar data, in this setting [11]. Here, a common setting is to summarize the data as it arrives and later use those summaries to obtain the actual clustering. Usually, only a subset of the data stream is summarized at once, with older data being aged out of the summaries. While the typical approach is to summarize multiple instances into a single summary structure [1, 4], EMCStream [12] is a unique approach as it reduces the dimensionality of the stream data rather than compressing multiple instances. EMCStream works by applying UMAP [8] on a batch of the data stream. The fitted projection function is reapplied to incoming data batches until a data drift is detected. At that point, the data stream is reinitialized on a limited set of preceding batches, which produces a new UMAP projection function that is then used until the next drift is detected. Drift detection occurs when a k-Means clustering [7] is performed on the embedding of the current batch using the current main UMAP projection function does not share enough similarity (based on the Adjusted Rand Index (ARI) [6]) with a k-Means clustering of the same data with a UMAP trained only on that set of data. The interval during which drift checks are performed, and the ARI threshold, are both dynamically adjusted based on how often drifts are detected and the past drift differences. If no drift is detected, these values decay. The final k-Means clustering for a batch is reported as the clustering of that batch.

However, while UMAP is currently the most popular technique for extracting projections for visualization purposes, there are multiple other techniques that share similar properties, but may be more suitable for different purposes or provide better guarantees. DensMap [9] extends UMAP with better density preservation, TriMAp [2] provides a better global view of the data, and PaCMAP [10]

attempts to capture both local and global structure. This raises the question of how suitable these alternative projection techniques are for stream clustering with EMCStream.

This question should be investigated in this thesis by examining several dimensionality-reduction techniques regarding their suitability when combined with the general EMCStream framework. There are several criteria. The robustness of the projections (given different and the same data) and the speed of the methods are both important criteria, but the final clustering performance (e.g., using ARI) also needs to be considered. However, it should be noted that EMCStream's formulation deviates from standard stream clustering, as it requires storing a significant amount of data in memory to recompute its projections. As such, this aspect should also be considered in this thesis to check how small this overhead can be made without negatively affecting the capabilities of the algorithm.

2 Research Question

Is it advantageous to replace the UMAP projection in EMCStream with an alternate dimensionality-reduction technique?

3 Tasks & Goals

- **Literature Review:** Understanding of literature regarding dimensionality-reduction
- **EMCStream Implementation:** Reimplement a simplified version of EMCStream (fix windows for initialization and drift detection to the batch size)
- **Dimensionality Reduction:** Integrate different dimensionality reduction methods into your EMCStream implementation (at least, densMAP, PaCMAP and TriMAP)
- **Examination:** Compare the EMCStream variants based on:
  - Batch-size requirements (size of batches at which the dimensionality-reduction fails to produce consistent results)
  - Runtime (time requirements for the embedding process)
  - Drift Robustness (robustness of the embedding function to drift, given otherwise consistent parametrization)
  - Batch Robustness (robustness of the embedding function for identical data, given otherwise consistent parametrization)
  - Clustering Quality (ARI scores of the resulting k-Means clustering)
- **Discussion:** Analysis of experiments, particularly regarding trade-offs and design decisions made

4 Expected Outcomes

- Integrating different dimensionality-reduction techniques into EMCStream;
- A clear examination of the differences of different dimensionality-reduction techniques regarding batch-size requirements, runtime, robustness, and clustering quality;
- A well-documented thesis with reproducible code.;

5 Requirements

- Study in the field of computer science
- Prior programming experience in Python
- Beneficial: understanding of data stream mining and clustering
- Willingness to be included in a research paper on this topic later down the line

6 Notes

While an implementation of EMCStream is already available[1], using your own implementation is advisable for behavior consistency and clarity.
Code for densMAP and UMAP is available at `https://github.com/lmcinnes/umap`.
Code for PaCMAP is available at `https://github.com/YingfanWang/PaCMAP`.
Code for TriMAP is available at `https://github.com/eamid/trimap`.

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] Ehsan Amid and Manfred K. Warmuth. Trimap: Large-scale dimensionality reduction using triplets. *CoRR*, abs/1910.00204, 2019.

[1] `https://gitlab.com/alaettinzubaroglu/emcstream`

[3] Brian Babcock, Shivnath Babu, Mayur Datar, Rajeev Motwani, and Jennifer Widom. Models and issues in data stream systems. In Lucian Popa, Serge Abiteboul, and Phokion G. Kolaitis, editors, *Proceedings of the Twenty-first ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, June 3-5, Madison, Wisconsin, USA*, pages 1–16. ACM, 2002.

[4] Feng Cao, Martin Ester, Weining Qian, and Aoying Zhou. Density-based clustering over an evolving data stream with noise. In Joydeep Ghosh, Diane Lambert, David B. Skillicorn, and Jaideep Srivastava, editors, *Proceedings of the Sixth SIAM International Conference on Data Mining, April 20-22, 2006, Bethesda, MD, USA*, pages 328–339. SIAM, 2006.

[5] João Gama, Indre Zliobaite, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. A survey on concept drift adaptation. *ACM Comput. Surv.*, 46(4):44:1–44:37, 2014.

[6] Lawrence Hubert and Phipps Arabie. Comparing partitions. *J. Classif.*, 2:193–218, 1985.

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

[8] Leland McInnes and John Healy. UMAP: uniform manifold approximation and projection for dimension reduction. *CoRR*, abs/1802.03426, 2018.

[9] Ashwin Narayan, Bonnie Berger, and Hyunghoon Cho. Assessing single-cell transcriptomic variability through density-preserving data visualization. *Nature biotechnology*, 39(6):765–774, 2021.

[10] Yingfan Wang, Haiyang Huang, Cynthia Rudin, and Yaron Shaposhnik. Understanding how dimension reduction tools work: An empirical approach to deciphering t-sne, umap, trimap, and pacmap for data visualization. *J. Mach. Learn. Res.*, 22:201:1–201:73, 2021.

[11] Alaettin Zubaroğlu and Volkan Atalay. Data stream clustering: a review. *Artificial Intelligence Review*, 54(2):1201–1236, 2021.

[12] Alaettin Zubaroglu and Volkan Atalay. Online embedding and clustering of evolving data streams. *Stat. Anal. Data Min.*, 16(1):29–44, 2023.