中国科学院软件研究所学术年会’2020
暨计算机科学国家重点实验室开放周

学术论文

DistStream: An Order-Aware Distributed Framework for Online-Offline Stream Clustering Algorithms

Lijie Xu[1], Xingtong Ye[1], Kai Kang[1], Tian Guo[2], Wensheng Dou[1], Wei Wang[1], Jun Wei[1]

[1] Institute of Software, Chinese Academy of Sciences [2] Worcester Polytechnic Institute

The 40th IEEE International Conference on Distributed Computing Systems (ICDCS 2020)

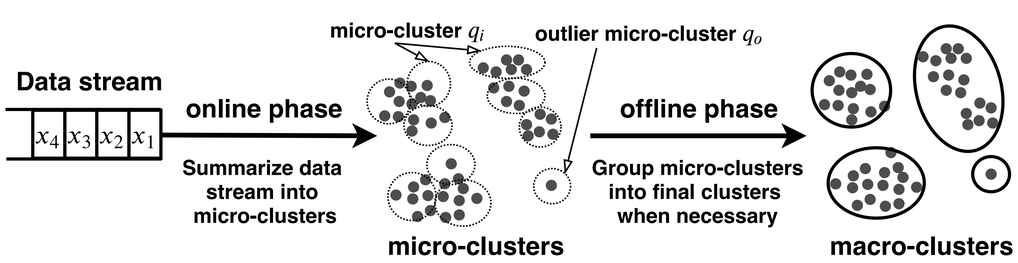
xulijie@iscas.ac.cn

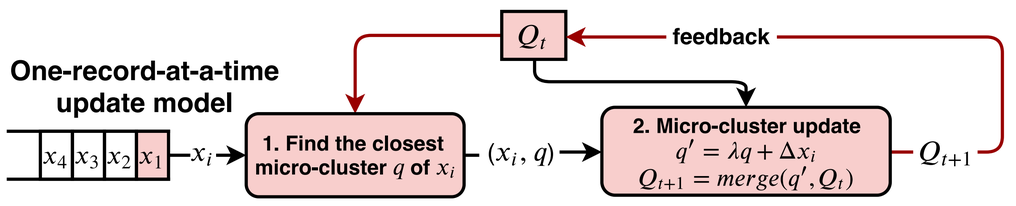
Motivation

Stream clustering algorithms are widely-used to capture the evolving patterns in real-time data streams, e.g., IoT events and Web clicks.

Problems: Existing stream clustering algorithms use a **one-record-at-a-time update** model that runs in a single machine.

- **Suffer from low throughput** (e.g., 5K records/s)
- **Cannot efficiently process high-speed** data streams (e.g., 256K transactions/s at Alibaba).





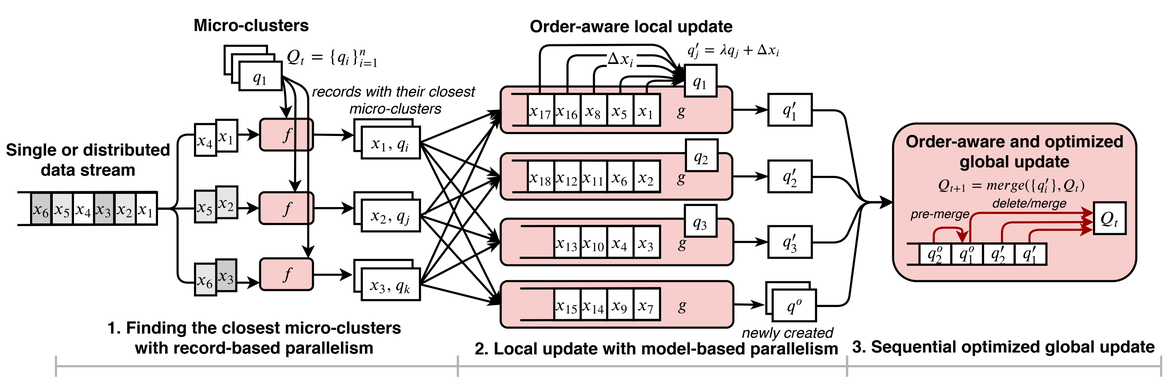
Goal and Challenges

Goal: Design a general distributed framework to parallelize stream clustering algorithms.

- How to **parallelize** stream clustering algorithms?
- How to **guarantee the clustering quality** of the parallelized stream clustering algorithms?

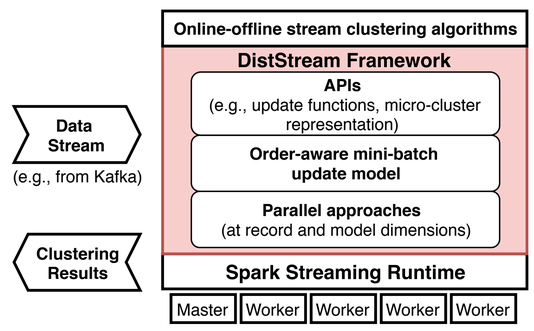
Our Approach (DistStream)

1. For parallelization, we design a new **mini-batch update model** with efficient (both record-based and model-based) parallelization approaches.
2. To maintain the algorithms' clustering quality, we design an **order-aware update mechanism** and theoretically demonstrate its effectiveness.



DistStream Implementation

- We implement DistStream framework atop widely-used Spark Streaming.
- DistStream exposes **four APIs**
  - Micro-cluster representation
  - Distance computation
  - Local update
  - Global update

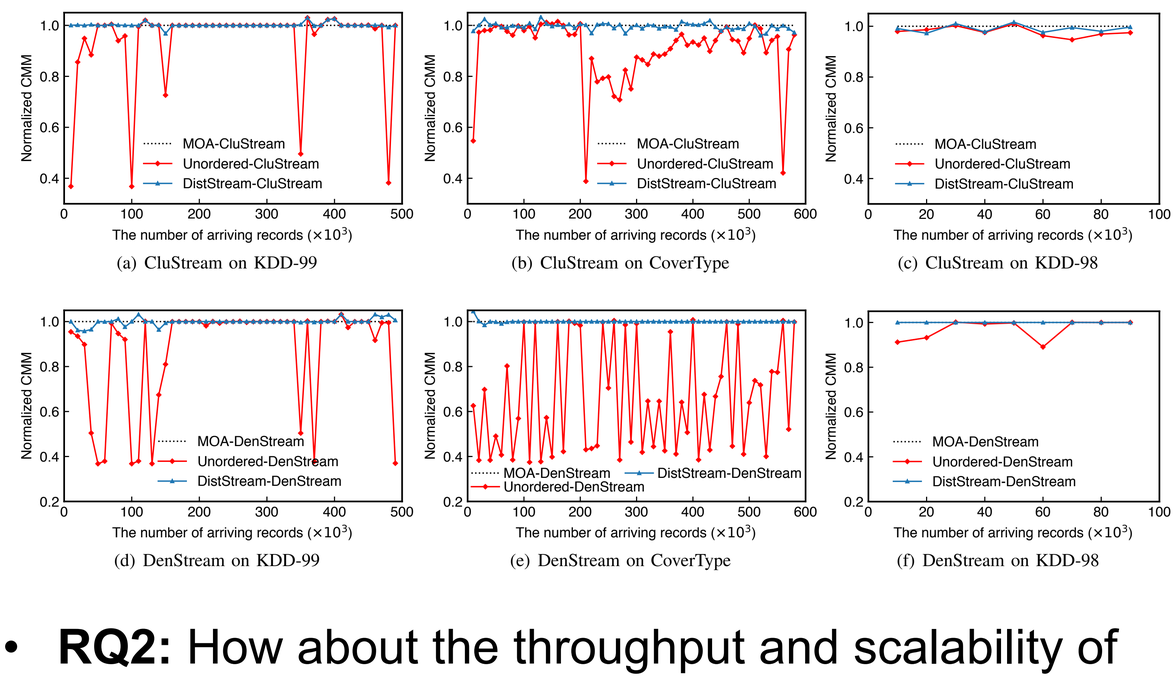


- DistStream currently includes four algorithms, including CluStream, DenStream, D-Stream, and ClusTree.

Evaluation

- **RQ1:** How about the clustering quality of our DistStream-based stream clustering algorithms?

✓ DistStream-based algorithms achieve **comparable (99%) clustering quality** with the original single-machine stream clustering algorithms.

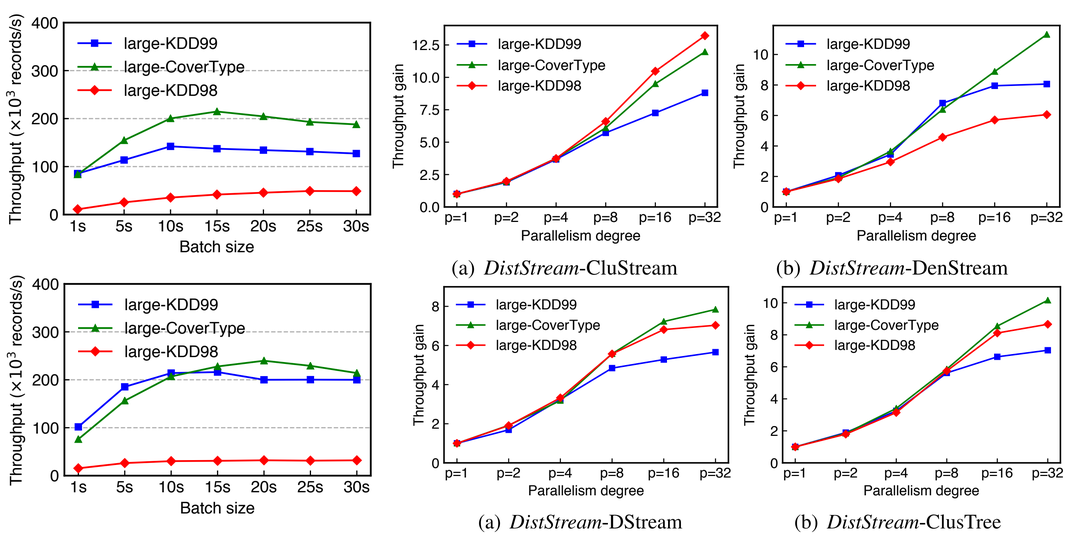


(a) CluStream on KDD-99 (b) CluStream on CoverType (c) CluStream on KDD-98

(d) DenStream on KDD-99 (e) DenStream on CoverType (f) DenStream on KDD-98

- **RQ2:** How about the throughput and scalability of DistStream-based algorithms?

✓ DistStream-based algorithms can achieve **13.2x throughput gain** (e.g., 239K records/s) on 32 cores.



(a) *DistStream*-CluStream (b) *DistStream*-DenStream

(a) *DistStream*-DStream (b) *DistStream*-ClusTree