Frequent Itemset Mining in Data Stream

...

Professor: Carlo Zaniolo
Members: Yuqing Wang and Jiyuan Shen

Roadmap

- **Conceptual Overview**
    - What it is
    - Applications
    - Key Challenges
    - Preliminaries and Concepts
- **Two Most Classic Algorithms**
    - Lossy Counting on Landmark
    - Lossy Counting on Sliding
- **Exploration and Conclusion**
    - Overall Analysis
    - Important Issues

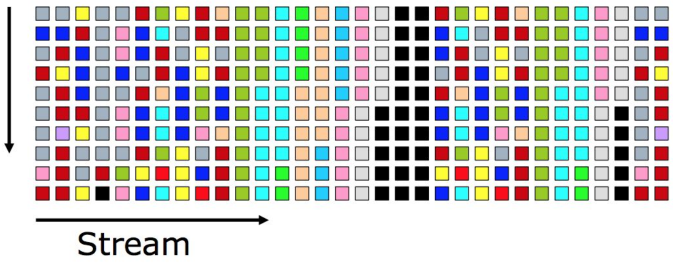
1. Conceptual Overview

1.1 What is "Frequent Itemset Mining in Data Stream"?

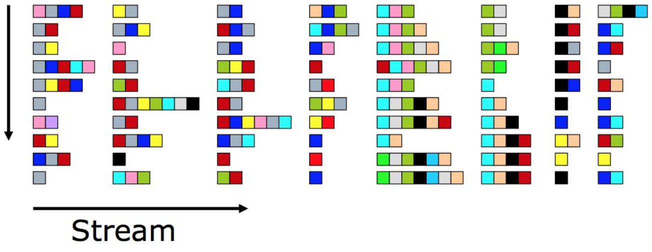
" In a given data stream, find those itemsets which appears more than the expected threshold "

❖ **Frequency of an Itemset:**

➢ freq(X) = # of transactions in window that contain X



➢ Identify all elements whose current frequency exceeds support threshold s = 0.1%.



➢ Identify all subsets of items whose current frequency exceeds s = 0.1%.

1.2 Applications

- Web Log and Click-stream Mining
- Fraud Detection in Telecommunications data
- Network Traffic Analysis
- E-business and Stock Market Analysis
- Trend Analysis
- Sensor Networks

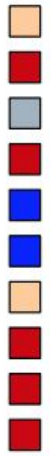
1.3 Key Challenges

- ❖ Memory Consumption: combinatorial explosion of itemsets
- ❖ Processing Efficiency: fast, real-time
- ❖ Single Pass: no multiple scans on stored data
- ❖ Data Representation: multi-dimensional

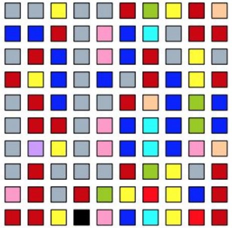
1.4 Preliminaries and Concepts

Itemset
X = {x1, x2, ..., xk}

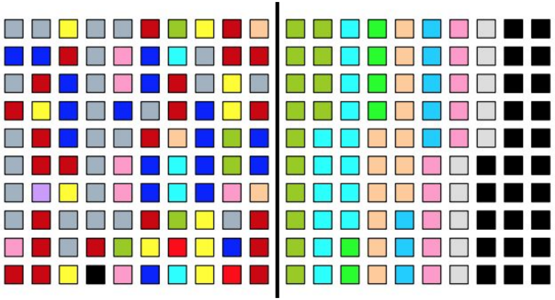
Transaction
tuple T = (transaction id, itemset X)



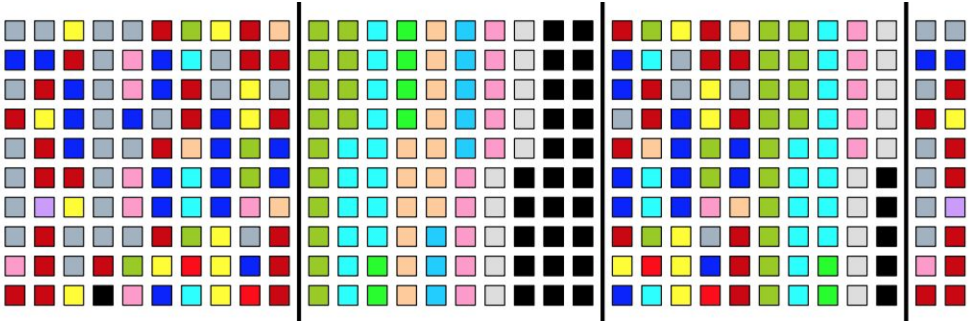
Bucket
a sequence of transactions



Batch
a sequence of buckets



Window
an except of stream



Data Stream
a sequence of incoming transactions

1.4 Preliminaries and Concepts

- Landmark Window Model
    - Data stream based on landmark windows requires handling disjoint portions of the streams, separated by landmarks
    - landmarks can be defined either in terms of time (e.g., on daily or weekly basis) or in terms of the number of elements observed since last landmark

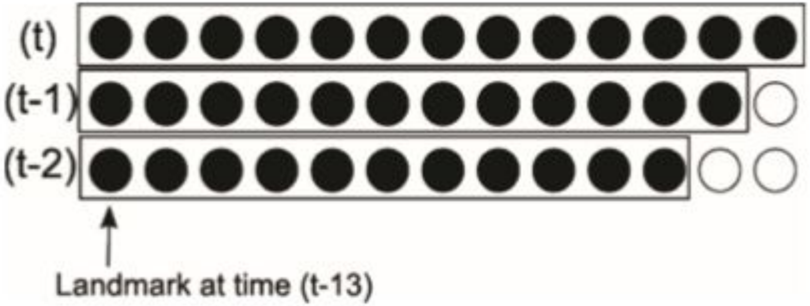


Fig. Landmark window for a time interval of size 13.

1.4 Preliminaries and Concepts

❖ Sliding Window Model

- ➢ Only the most recent information from the data stream are stored in a data structure.
- ➢ The data structure is usually a first-in first-out (FIFO) structure, which considers objects from the current period of time up to a certain period in the past.

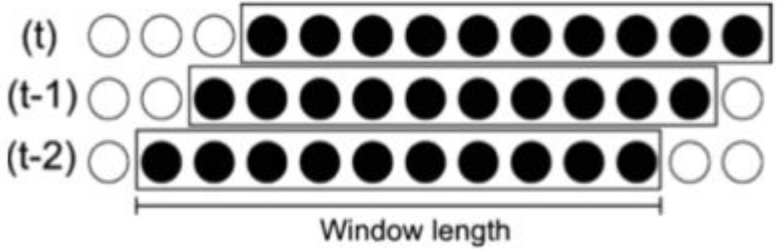


Fig. Sliding-window model.

2. Two Most Classic Algorithms

Approximate Frequency Counts over Data Streams

Gurmeet Singh Manku, Rajeev Motwani

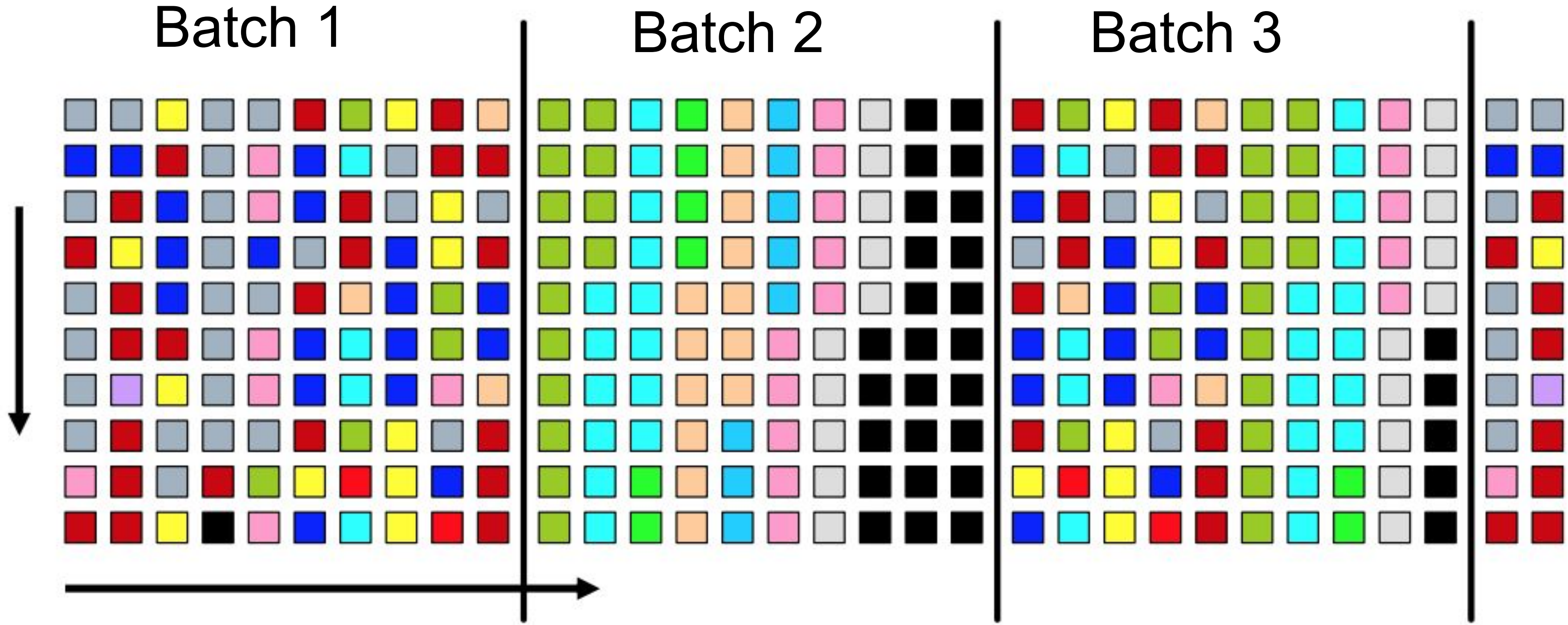
VLDB ‘02 Proceedings of the 28th International Conference on Very Large Data Bases. Pages: 346-357.

The Algorithm - Lossy Counting

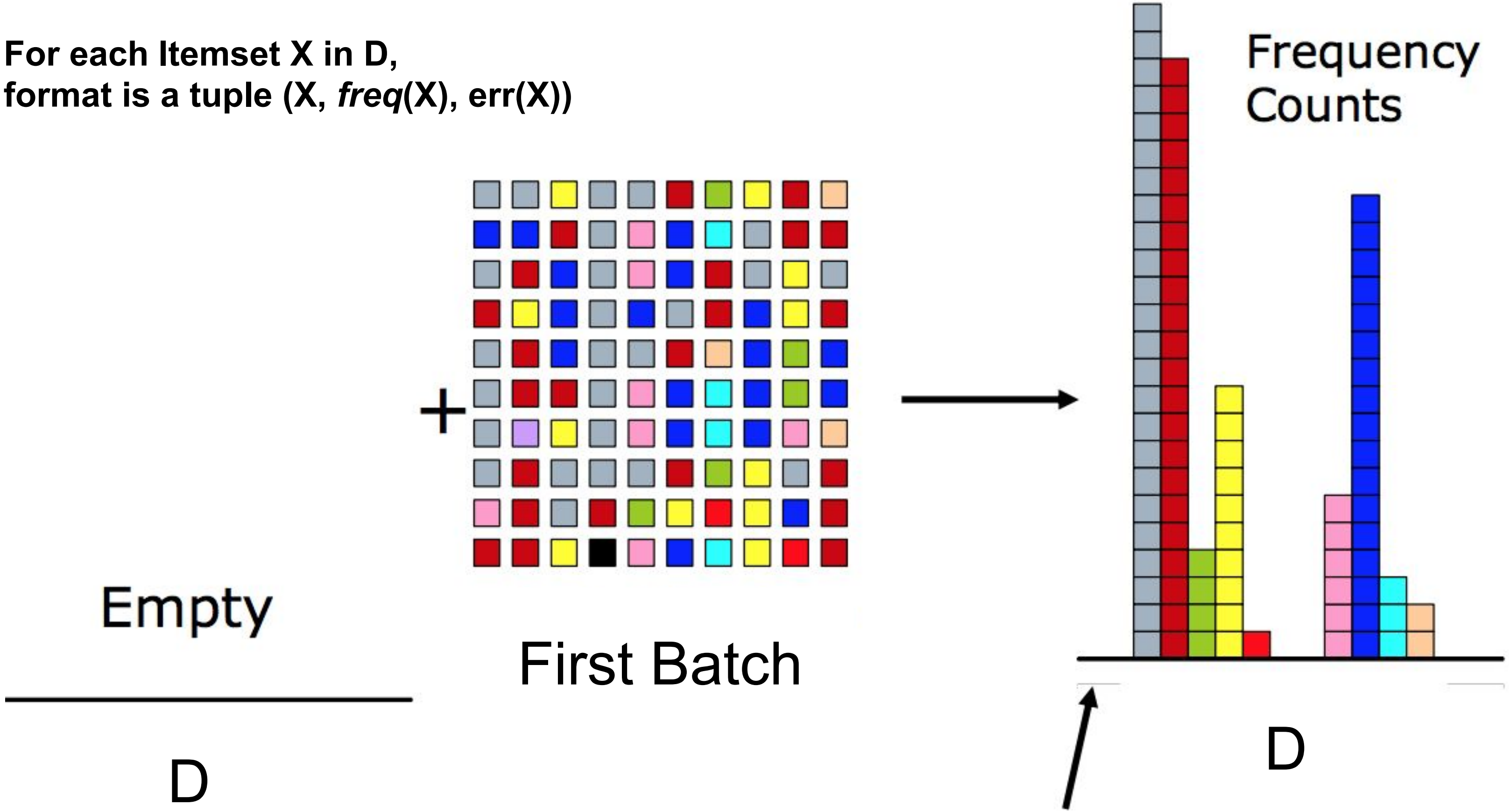
- UpdateEntry (For each itemset X in D)
  - add the frequency count of itemset X in the current batch
  - if (sum of X's frequency count and X's error para) is smaller than current batch id, then delete this itemset from D

- AddEntry
  - if the frequency count of an itemset X, in current batch, is at least the threshold, then add it into D, and assign its error para as (batch id - threshold)

Divide the stream into ‘Batches’

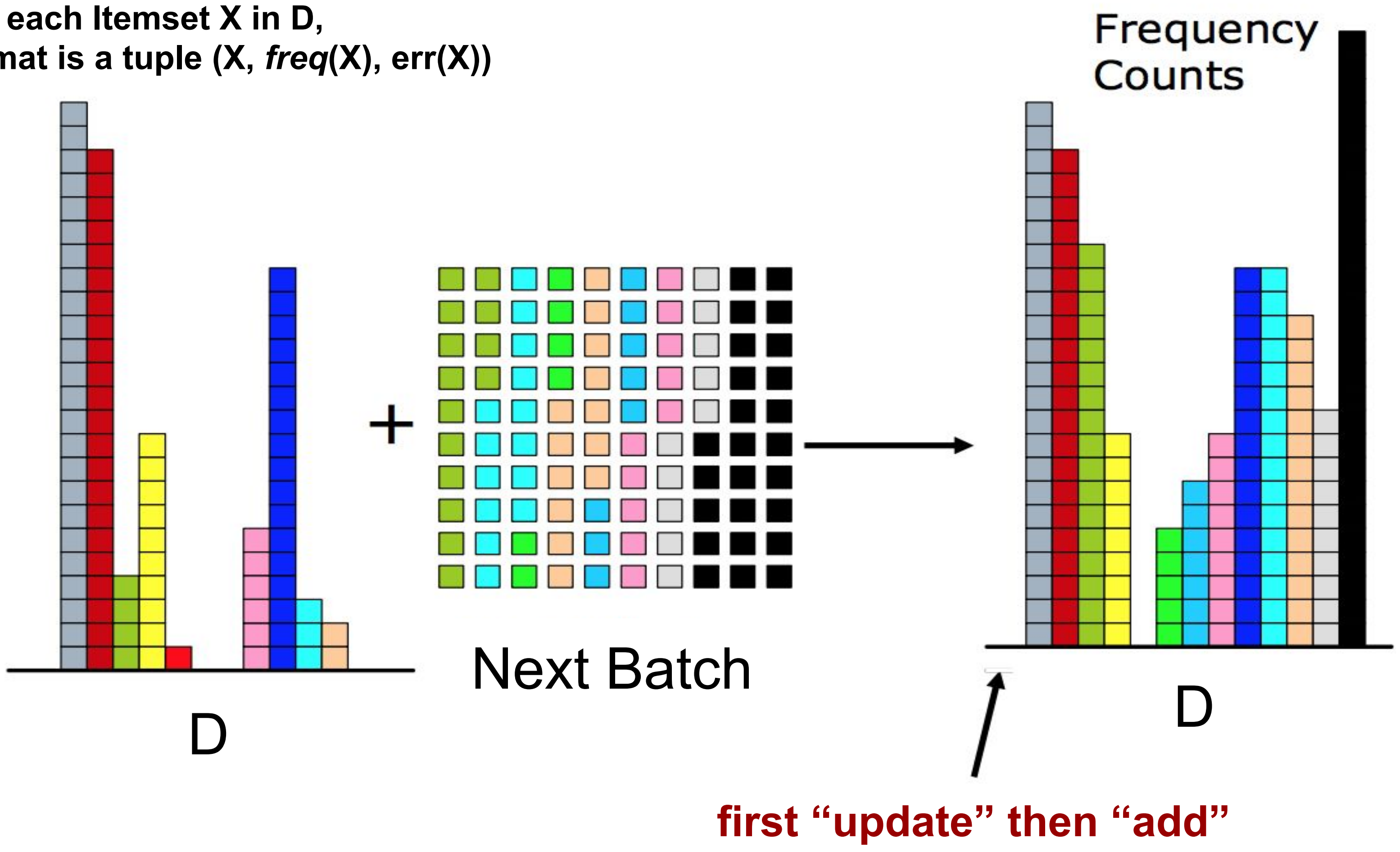


For each Itemset X in D,
format is a tuple (X, *freq*(X), err(X))



first "update" then "add"

For each Itemset X in D,
format is a tuple (X, *freq*(X), err(X))



The Guarantees ...

- All Itemsets whose true frequency exceeds σN are output. There are no false negatives.
- No Itemsets whose true frequency is less than $(\sigma - \varepsilon)N$ is output.
- Estimated frequencies are less than the true frequencies by at most εN

A Lossy-Counting-Based Algorithm over Data Streams

...

Joong Hyuk Chang, Won Suk Lee

In Journal of Information Science and Engineering,
Vol. 20, No. 4, July, 2004.

Introduction

- In a Data Stream...
  - Minimum support: Smin $\in$ (0, 1)
  - Error parameter: $\varepsilon \in$ (0, Smin)
  - Size of a sliding window: w
  - Recently Frequent Itemset (FI)
  - Significant Itemset
  - Maximum possible error count for the itemset = w * ε
    - Also called Pruning Threshold

Introduction

- In Main Memory...
  - Monitoring Lattice: each node contains an enrty (e, f, t)
    - e: Corresponding itemset
    - f: Count of the itemset
    - t: Transaction where the itemset was newly inserted
  - Current Transaction List (CTL)
    - Maintains all transactions of the current window

Theorem

- Given an error parameter ε, when w_{first} denotes the TID of the first transaction of the current window, the maximum possible count $C_{max}^{k}(e)$ of an itemset with its entry (e, f, t) is found as follows:

$$C_k^{max}(e) = \begin{cases} f & \text{if } t \le w_{first} \\ f + \lfloor (t - w_{first}) \times \varepsilon \rfloor & \text{otherwise} \end{cases}$$

The Algorithm ...

- Two Different Phases
  - Window Initialization Phase
    - Happens when the number of transactions so far is smaller or equal to window size
    - A new transaction is appended to a CTL
    - No extracted transaction
  - Window Sliding Phase
    - Happens when CTL is full
    - A new transaction is in
    - Oldest transaction is out

The Algorithm …

- Five Steps
  - Step 1: Appending a Transaction
  - Step 2: Count Updating and Insertion of New Itemsets
  - Step 3: Extracting a Transaction
    - Only in Window Sliding Phase
  - Step 4: Pruning Itemsets
    - Only periodically if needed
  - Step 5: Frequent Itemset Selection
    - Only when Up-to-date set of Recently FI is Requested

3. Exploration and Conclusion

An Overall Analysis

Representative Work	**Window Model**	**Update Interval**	**Approximation Type**	**False Results**
Manku & Motwani	Landmark Count-Based	Per Batch	False-Positive	$\{X \mid (\sigma - \varepsilon) \leq sup(X) < \sigma\}$
Chang & Lee	Sliding Count-Based	Per Transaction	False-Positive	$\{X \mid (\sigma - \varepsilon) \leq sup(X) < \sigma\}$

Exact Vs Approximate Mining

- ❖ Exact Mining
  - ➢ Keeps all itemsets' records
  - ➢ Number of itemset is large
- ❖ Approximate Mining
  - ➢ Widely adopted
  - ➢ Goal: general identification rather than exact result

Load Shedding

- ❖ Approximate the Processing Rate
  - ➢ Number of transaction per unit of time machine can handle
- ❖ Characteristic of Stream
  - ➢ Average size of a transaction
  - ➢ Average size of an FI
  - ➢ Memory requirement at a particular processing rate
- ❖ How to do Load Shedding
  - ➢ Random sampling
  - ➢ Semantic drop
  - ➢ Window reduction

Reference

[1] G. S. Manku and R. Motwani. Approximate Frequency Counts over Data Streams. In Proc. of VLDB, 2002

[2] J. H. Chang and W. S. Lee. A Sliding Window method for Finding Recently Frequent Itemsets over Online Data Streams. In Journal of Information Science and Engineering, Vol. 20, No. 4, July, 2004.

Thanks for Listening

...