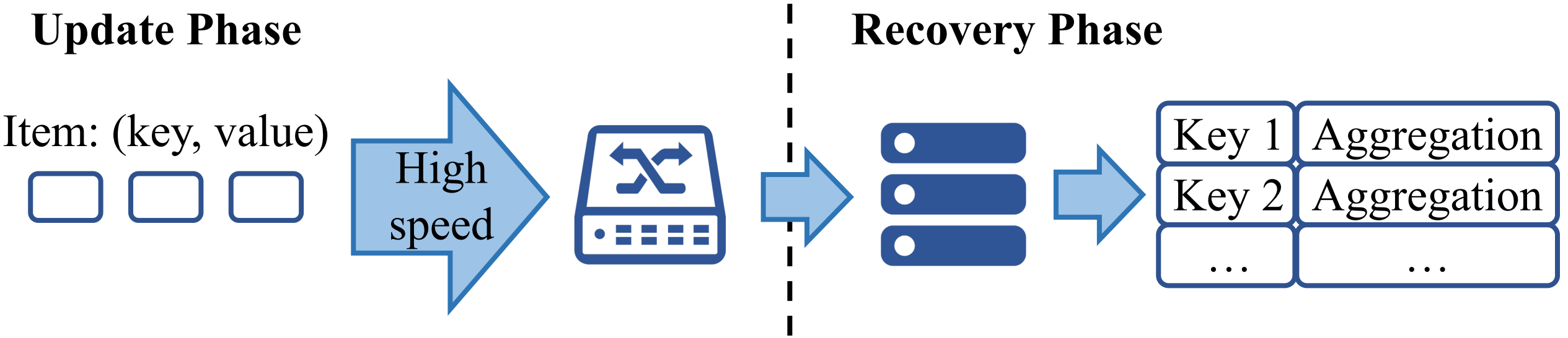
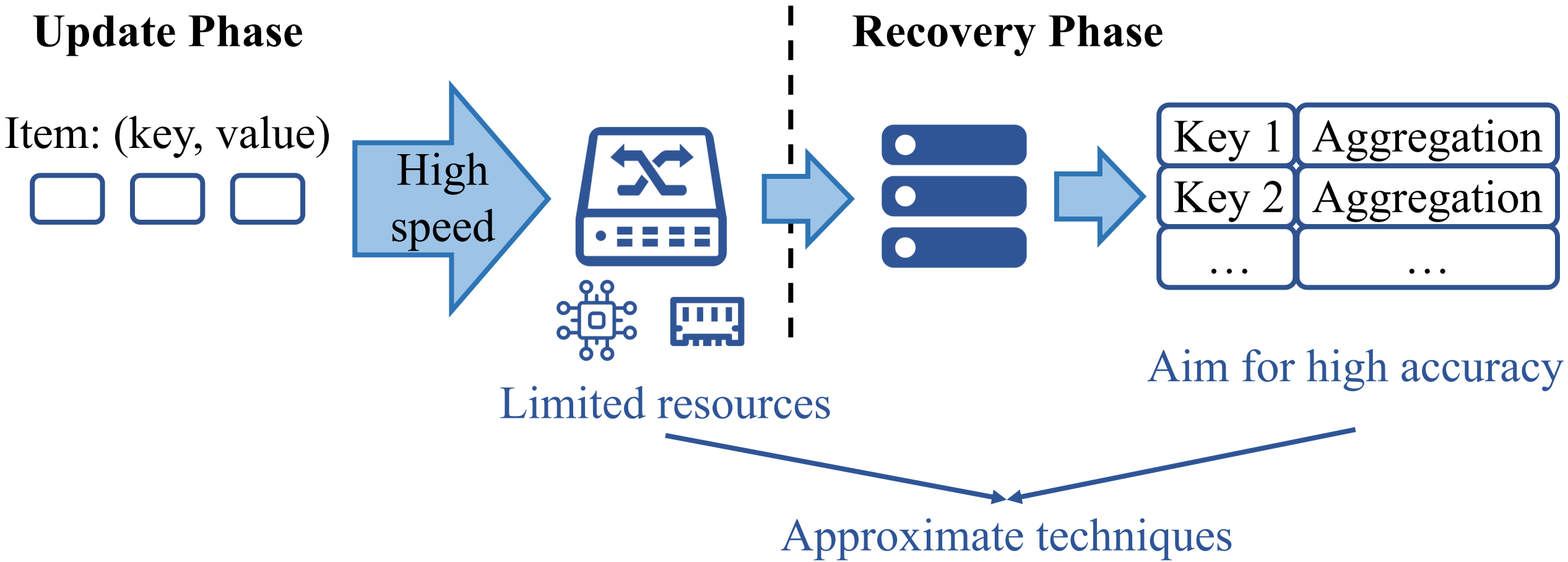
PR-Sketch: Monitoring Per-key Aggregation of Streaming Data with Nearly Full Accuracy

Siyuan Sheng, Qun Huang, Sa Wang, Yungang Bao

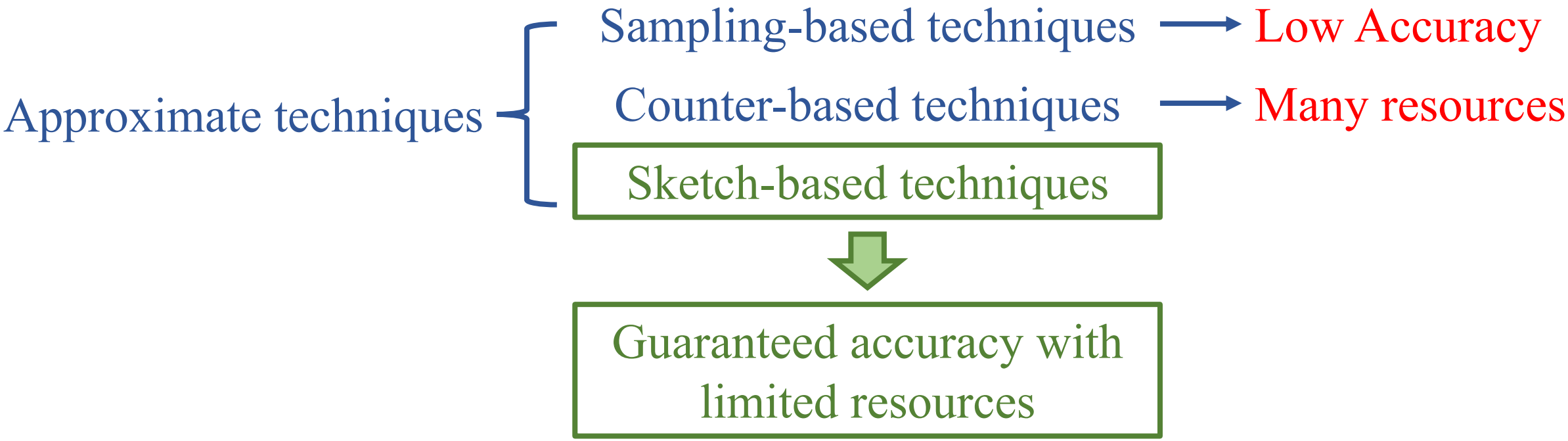
Stream Processing



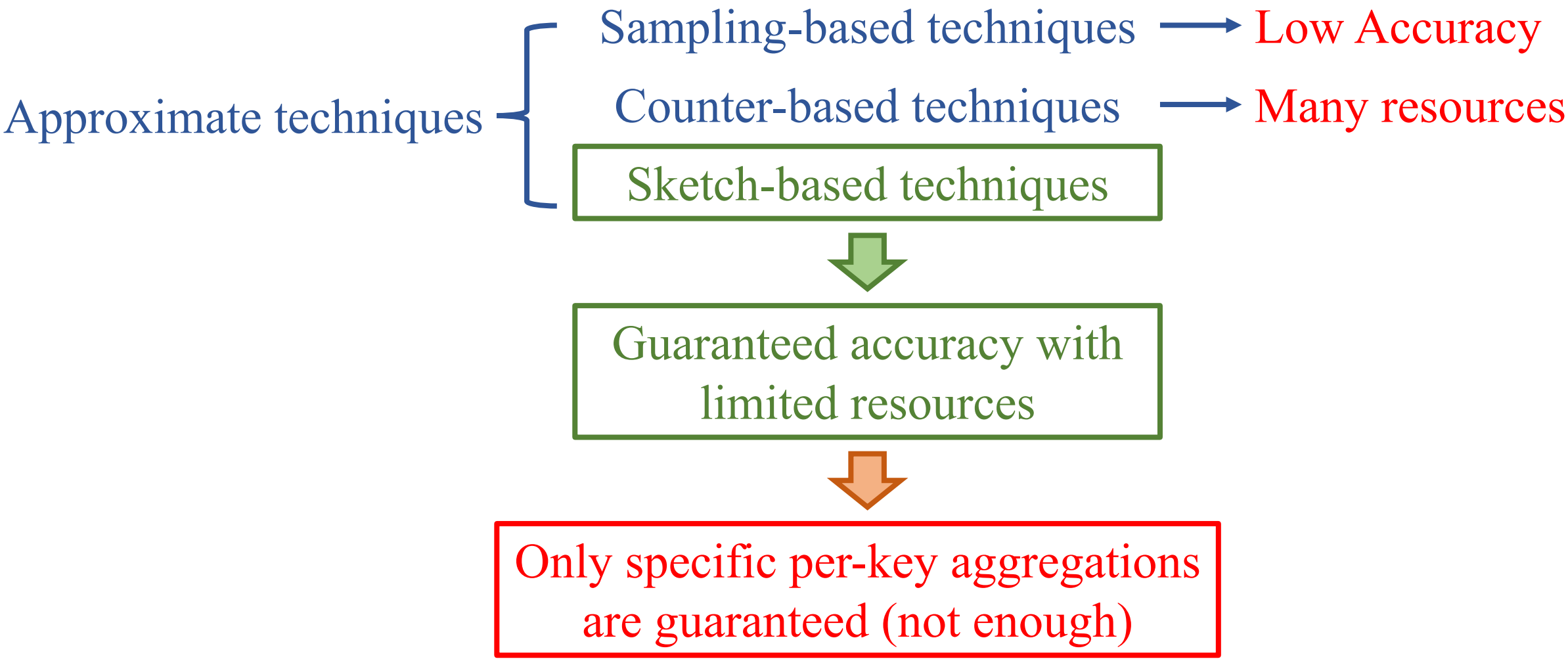
Stream Processing



Existing Approximation



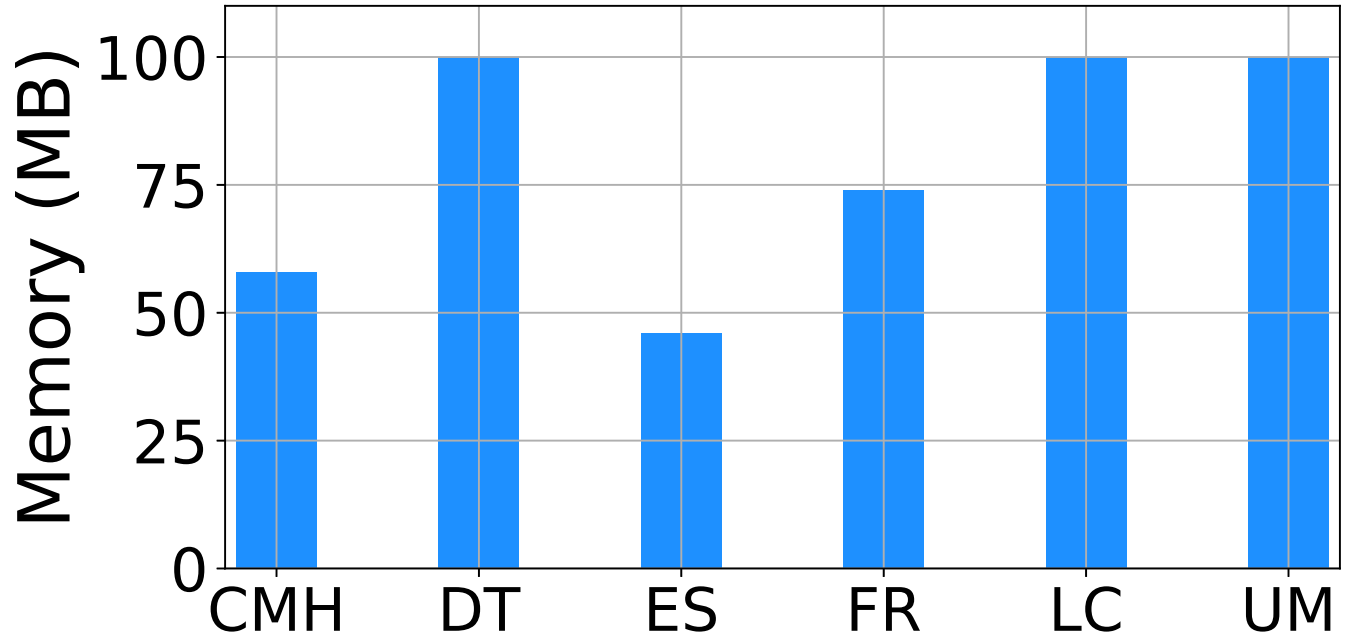
Existing Approximation



Preliminary Experiment

Target: accurately recover at least **95%** per-key aggregations

Result: existing sketching needs at least around **50 MB** memory

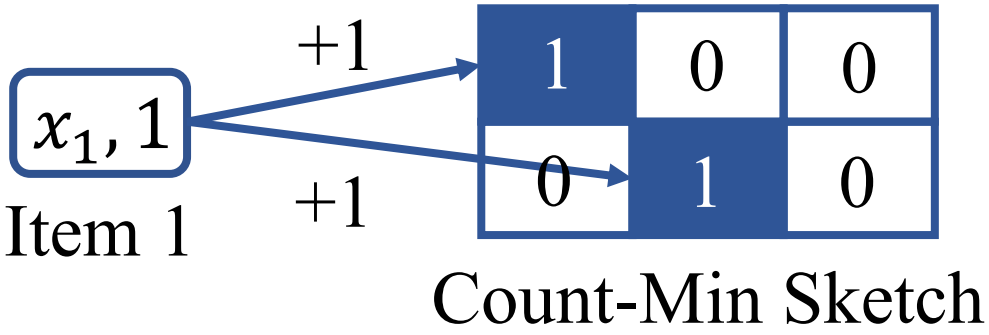


Root Causes

- Simple estimation
  - Estimate per-key aggregations by simply calculating counter values
- Complicated key tracking
  - Heavy-weight mechanisms to track the keys that have appeared

Example of Simple Estimation

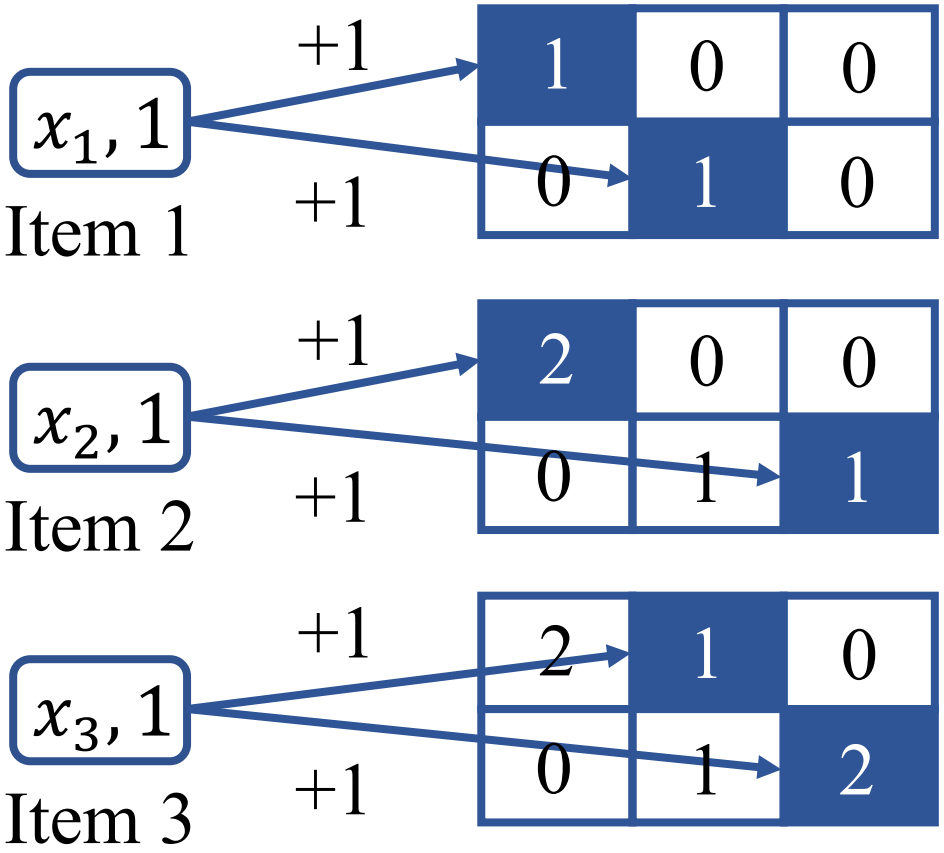
Update Phase



Recovery Phase

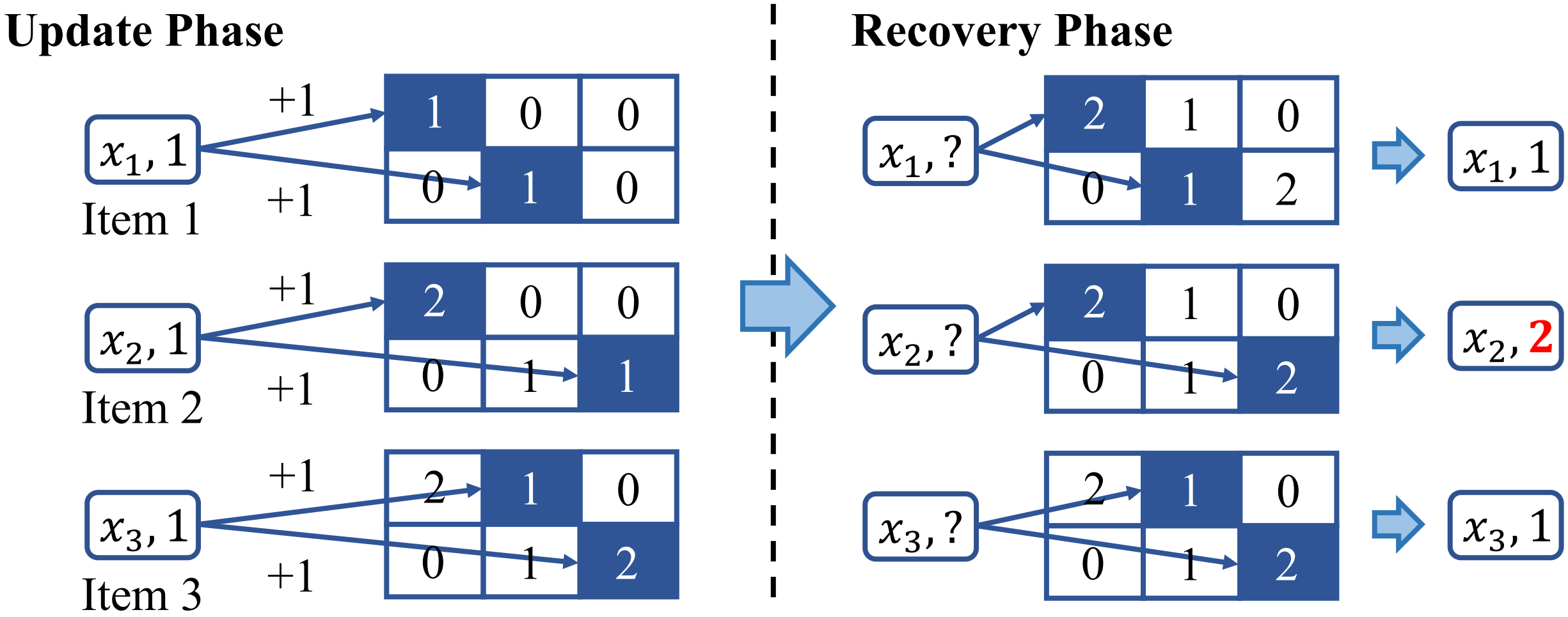
Example of Simple Estimation

Update Phase



Recovery Phase

Example of Simple Estimation

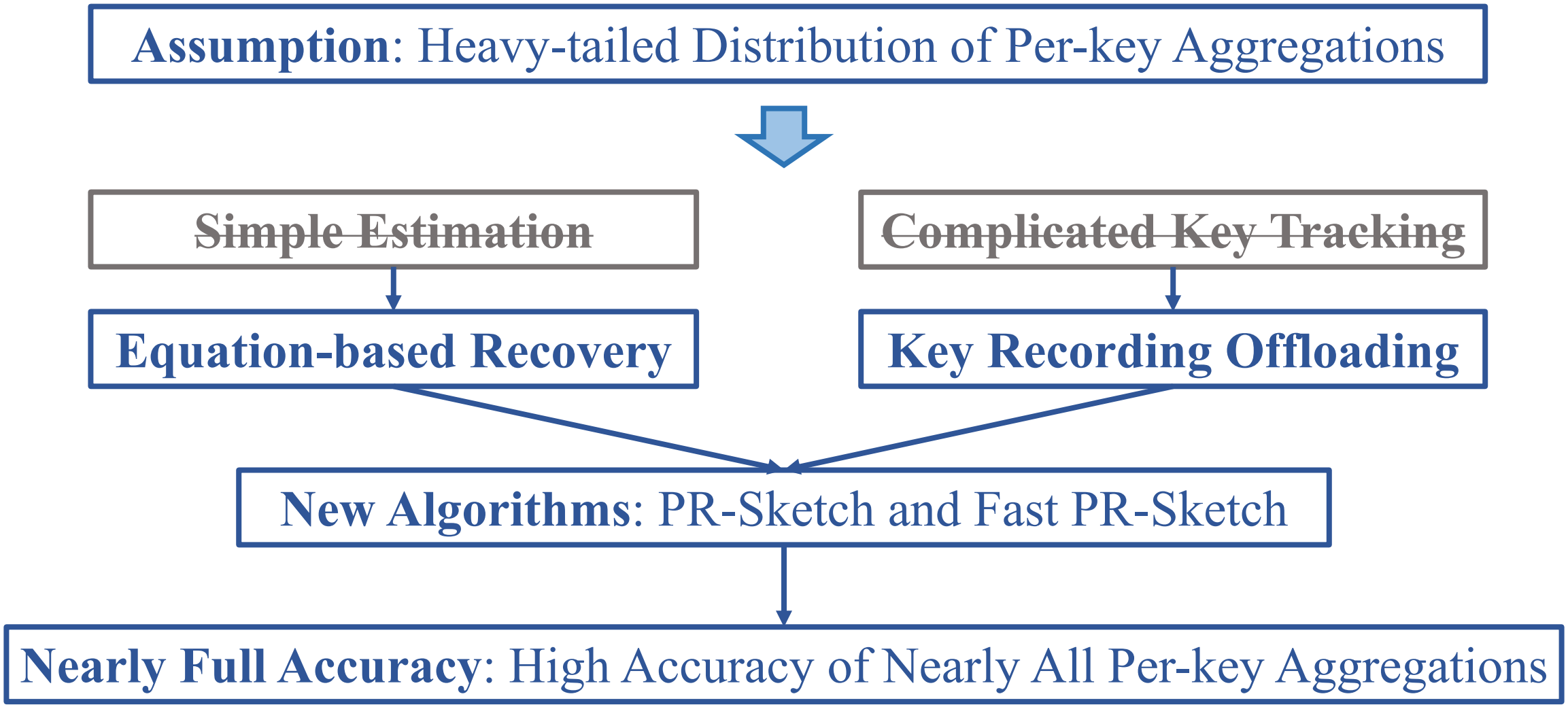


Complicated Key Tracking

- Existing key tracking mechanisms
  - Time-consuming: XOR coding, multi-level hashing, and group testing
  - Memory-consuming: dedicated buckets

- Considerable memory and computation are required
  - More hash collisions in the update phase ☹

Our Contributions

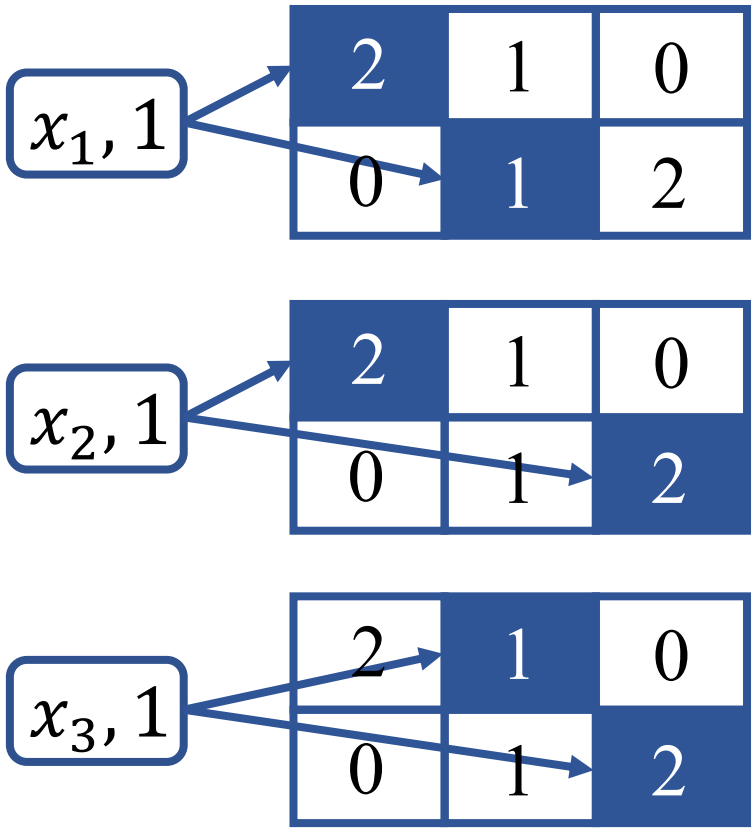


Heavy-tailed Distribution

- Characteristics
  - Most per-key aggregations are small
  - The majority of stream volume is contributed by a few large aggregations
- Validation
  - Our workloads: network traffic, click stream, and market basket data
- Two design features based on it
  - Equation-based recovery
  - Key recording offloading

Example of Equation-based Recovery

Recovery Phase



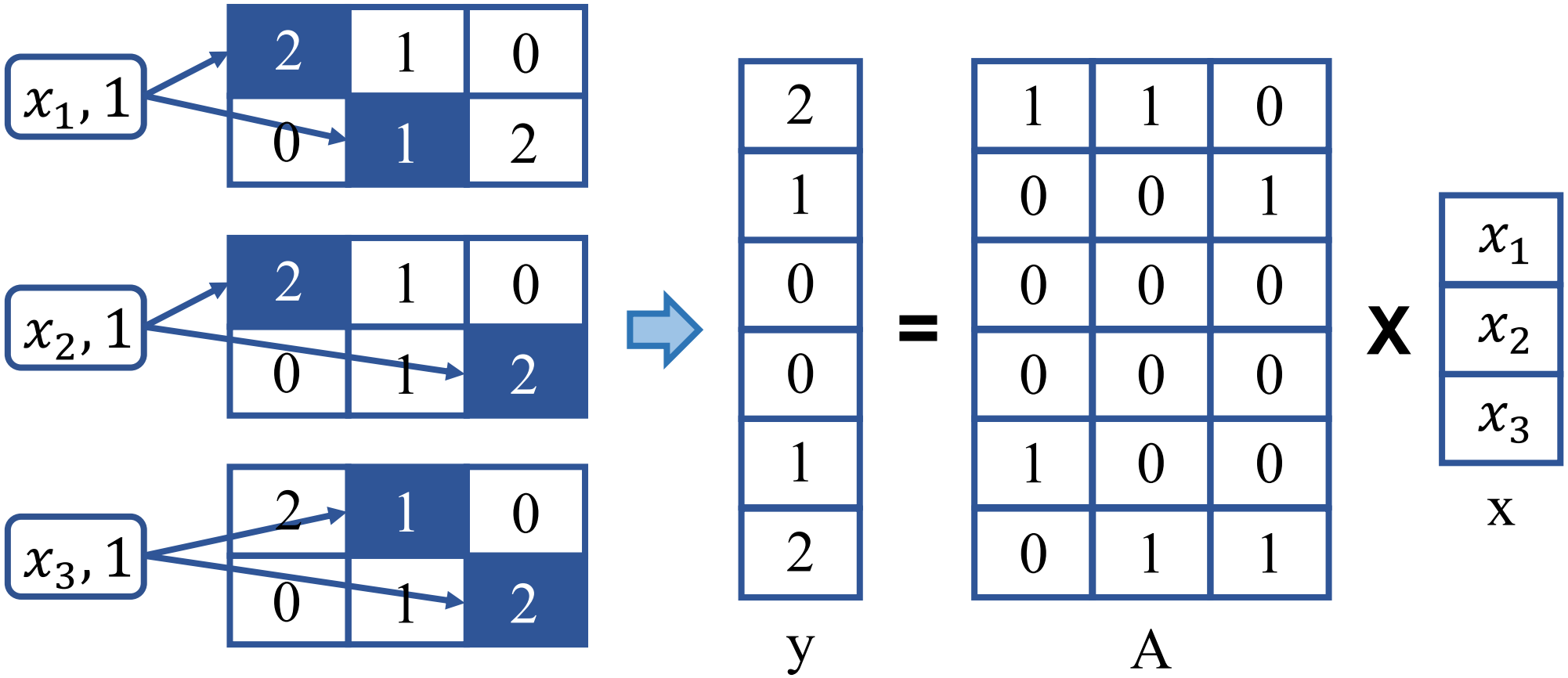
$$x_1 + x_2 = 2$$
$$x_1 = 1$$

$$x_1 + x_2 = 2$$
$$x_2 + x_3 = 2$$

$$x_3 = 1$$
$$x_2 + x_3 = 2$$

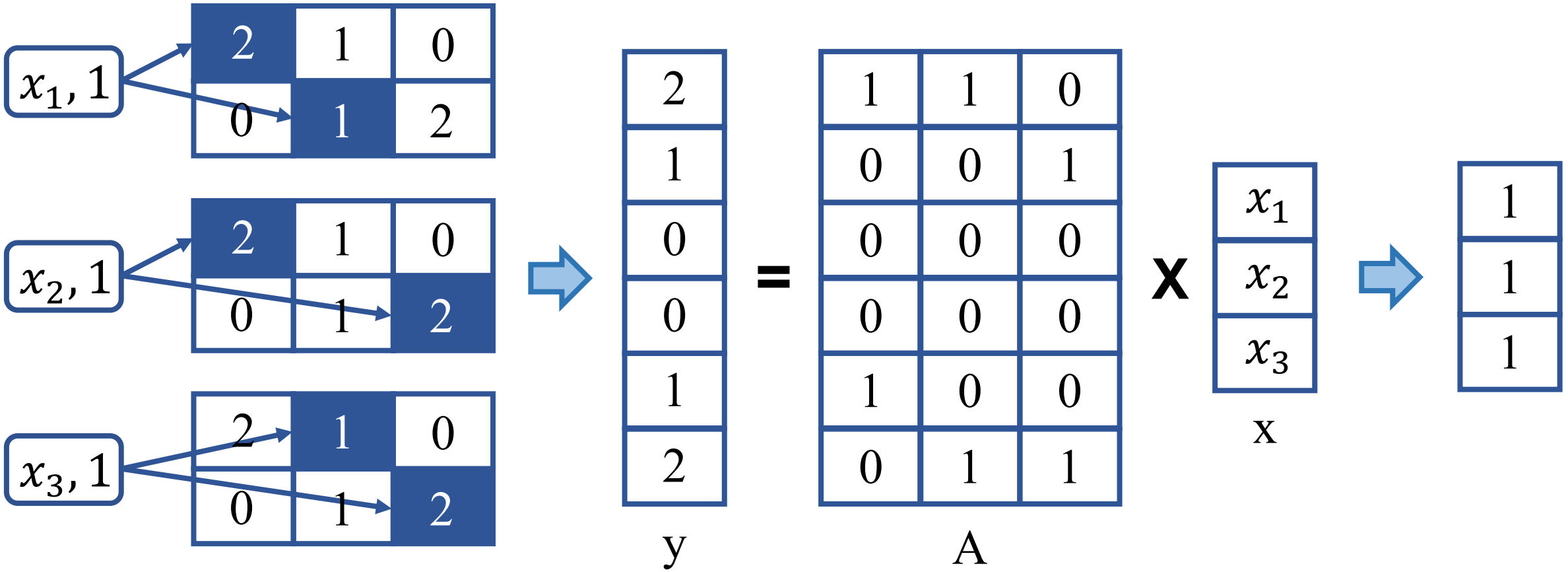
Example of Equation-based Recovery

Recovery Phase



Example of Equation-based Recovery

Recovery Phase



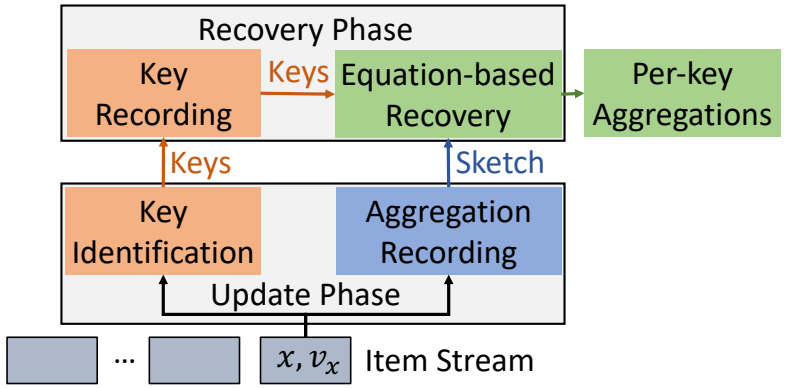
Equation-based Recovery

- Under-constrained case
  - Infinite feasible yet irrelevant solutions → Undermine recovery accuracy
- Heavy-tailed distribution
  - Most hash collisions are caused by keys with similar small aggregations
- ℓ_2 norm minimization
  - Penalization on large aggregations and well suited by small “noises”

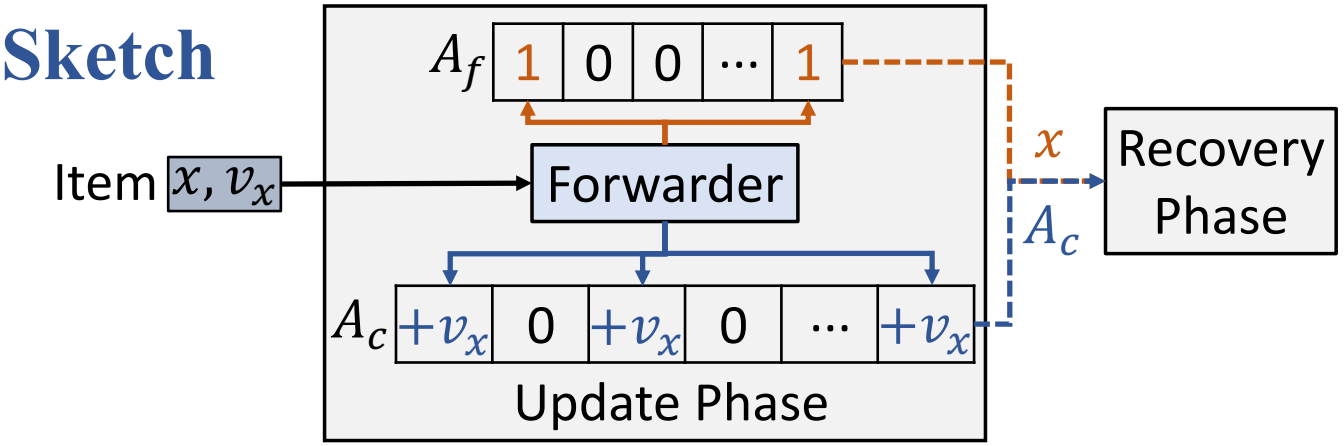
Key Recording Offloading

- Update phase
  - A lightweight bloom filter to identify new keys
  - Report newly identified keys to the recovery phase
- Heavy-tailed distribution
  - A few keys contributing major stream volume are reported only once
- Limited bandwidth usage for reporting keys

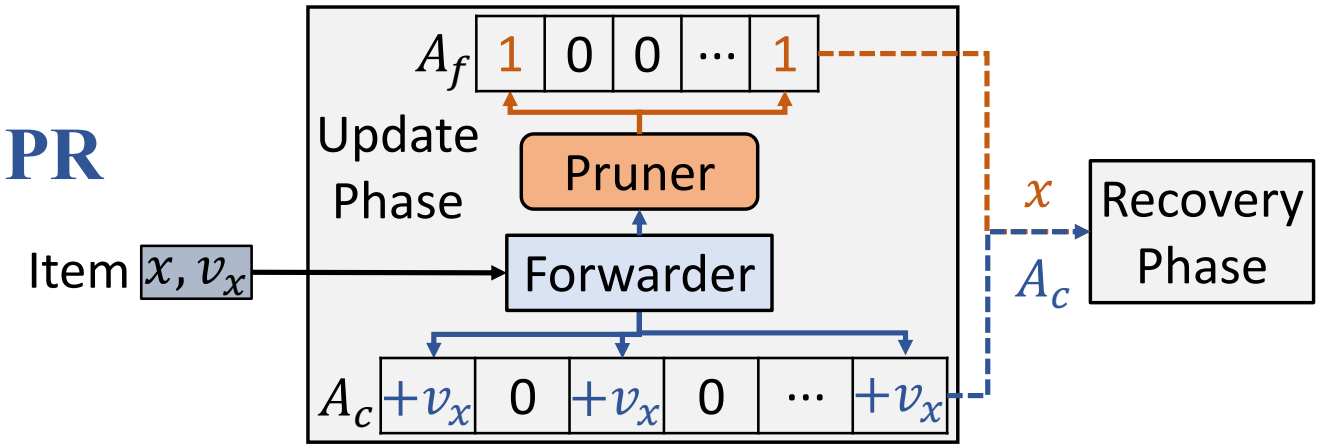
New Algorithms



PR-Sketch



Fast PR



Our Results

- Accuracy
  - 100% precision, 100% recall, and 100% F1 score
  - Accurately recover (<0.1% relative error) >95% per-key aggregations
- Resources
  - Throughput: >30 Mips
  - Limited bandwidth usage
  - Limited recovery time
- Generality on both real-world and synthetic workloads
- Robustness on different parameter configuration
- Use cases

Thank you