**Summary**

- **Goal:** to detect and update dense subtensors in streaming tensors
- **Previous Work**
  - showed that dense subtensors signal anomalies or fraud
  - batch algorithms for fast and accurate dense subtensor detection
- **Algorithm:** incremental algorithm for detecting the densest subtensor
- **Result:**
  - **fast:** up to 320x faster than best streaming methods,
  - **robust:** splicing theory to do incremental splices for dense block,
  - **accurate:** successfully detect anomalies in real-world tensors, including App rank boosting fraud, and rating manipulations.

**Motivation**

- Synchronized behavior in App data: rank boosting fraud results in dense subtensors

**Basic Concepts**

- **3-mode tensor**
  - density: \( g() = \frac{M()}{S()} \) the sum of entries / number of attributes

**Proposed Algorithm: AugSplicing**

- **Goal:** to incrementally update dense subtensors while the input tensor changes
- **Overall algorithm:** iteratively choose two blocks from candidate blocks and splice these two blocks until reaches the splicing threshold, and output top \( k \) dense blocks.

**Procedure of splicing two blocks:**

1. **splice** \( B \) into \( B' \) if \( g(B') \geq g(B) \) to make \( g(B') \) increase
2. randomly choose a mode to splice until no large-mass blocks

**Splicing Theorem**

**Experimental Results**

- **Q1 Effectiveness:** what does AugSplicing detect in real-world tensors?
  1. App installation and uninstallation data
  2. Wi-Fi router connection and disconnection data

- **Q2 Speed:** How fast is AugSplicing compared to baselines?

- **Q3 Accuracy:** How accurately does AugSplicing update a dense subtensor?